

Una guía para el

**CUERPO DE CONOCIMIENTO DE SCRUM  
(GUÍA SBOK™)**

Edición 2016

A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™ Guide)



Una guía para el

**CUERPO DE CONOCIMIENTO DE  
SCRUM**

*(Guía SBOK™)*

---

**Edición 2016**

Una guía completa para la entrega de proyectos utilizando Scrum

© 2016 SCRUMstudy™, una marca de VMEdU, Inc. Todos los derechos reservados.

Número de control de la Biblioteca del Congreso

*Una guía para el cuerpo de conocimiento de Scrum (Guía SBOK™) - 2016 Edición*

Título original en inglés: *A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™ GUIDE) 2016 Edition*

Incluye referencia bibliográfica e índice.

ISBN: 978-0-9899252-0-4

1. Marco de Scrum.I. SCRUMstudy™. II. *Guía SBOK™*

2013950625

ISBN: 978-0-9899252-0-4

Publicado por:

SCRUMstudy™, una marca de VMEdU, Inc.

410 N 44<sup>th</sup> Street, Suite 240

Phoenix, Arizona 85008 USA

Teléfono: +1-480-882-0706

Fax: +1-240-238-2987

Correo Electrónico: [sbok@scrumstudy.com](mailto:sbok@scrumstudy.com)

Sitio Web: [www.scrumstudy.com](http://www.scrumstudy.com)

"SBOK", el logotipo de SCRUMstudy, "SDC", "SMC", "AEC", "SPOC", y "ESM" son marcas registradas de SCRUMstudy™ (una marca de VMEdU, Inc.) Para obtener una lista completa de las marcas SCRUMstudy™, póngase en contacto con el departamento jurídico de SCRUMstudy™.

La *Guía para el cuerpo de conocimiento de Scrum (Guía SBOK™)* se ofrece para propósitos educativos. SCRUMstudy™ o VMEdU, Inc.

no garantizan que sea adecuada para cualquier otro propósito; no brinda una garantía expresa o implícita de ningún tipo, ni asumen responsabilidad por errores u omisiones. No se asume responsabilidad por daños incidentales o consecuentes en conexión que deriven de su uso o que surjan del uso de la información contenida en la guía.

SCRUMstudy™ acepta correcciones y comentarios sobre sus libros. Envíe por favor sus comentarios sobre errores tipográficos, de formato o de otro tipo. Puede fotocopiar la página indicada, señalar el error y enviarla al domicilio antes mencionado o por correo electrónico [sbok@scrumstudy.com](mailto:sbok@scrumstudy.com).

Se prohíbe la reproducción o transmisión parcial de cualquier tipo o por cualquier medio, ya sea electrónico, manual, de fotocopiado, grabación o cualquier sistema de almacenamiento y recuperación, sin la autorización previa por escrito de la editorial.

10 9 8 7 6 5 4 3 2

---

## PREFACIO

*Una guía para el cuerpo de conocimiento de Scrum (Guía SBOK™)* proporciona directrices para la aplicación con éxito de Scrum—el desarrollo de productos Ágil y metodología de entrega de proyectos más popular. Scrum, tal como se define en la *Guía SBOK™*, es un marco que se aplica a carteras, programas o proyectos de cualquier tamaño y complejidad; y se puede aplicar de manera efectiva en *cualquier* industria para crear un producto, servicio, o cualquier otro resultado.

La *Guía SBOK™* se desarrolló como una guía estándar para profesionales y organizaciones de negocios que deseen implementar Scrum, así como para los que ya lo estén utilizando y quieran aun mejorar el proceso. Está diseñada para utilizarse como referencia y guía de conocimiento tanto por profesionales con experiencia en Scrum, para profesionales de productos o servicios de desarrollo, así como para personas que no tengan experiencia previa o conocimiento de Scrum o cualquier otra metodología de entrega del proyecto.

La *Guía SBOK™* se basa en el conocimiento y la visión combinada obtenida de miles de proyectos a través de una variedad de organizaciones e industrias. Además, las contribuciones han sido hechas por expertos que han enseñado Scrum y cursos de entrega de proyectos a más de 500,000 profesionales en 150 países. Su desarrollo ha sido verdaderamente un esfuerzo de colaboración de un gran número de expertos en una variedad de disciplinas. En particular, me gustaría dar las gracias a los diecisiete coautores y expertos en la materia y los veintiocho correctores que contribuyeron en gran medida a la creación de la *Guía SBOK™*.

Una amplia adopción del marco *Guía SBOK™* ayudará a estandarizar la aplicación de Scrum a todo tipo de proyectos a través de las organizaciones a nivel mundial, al igual que ayudará a mejorar significativamente su retorno sobre la inversión. Además, promoverá una mayor reflexión y deliberación sobre la aplicación de Scrum para muchos tipos de proyectos, que a su vez contribuirá a ampliar y enriquecer el acervo de conocimientos y consecuentemente actualizaciones futuras de esta guía.

Aunque la *Guía SBOK™* es una guía completa y marco para la entrega de proyectos que utilizan Scrum, su contenido está organizado para una referencia fácil, independientemente de los conocimientos previos del lector sobre el tema. Espero que cada lector pueda aprender y disfrutar tanto como los autores y correctores aprendieron y disfrutaron del proceso de cotejo de los conocimientos y la sabiduría colectiva aquí contenida.



Tridibesh Satpathy

Autor principal, *Guía SBOK™*

---



**TABLA DE CONTENIDO**

1.	INTRODUCCIÓN .....	1
1.1	Información general de Scrum.....	2
1.1.1	Breve historia de Scrum .....	3
1.2	¿Por qué utilizar Scrum? .....	4
1.2.1	Escalabilidad de Scrum .....	5
1.3	Propósito de la <i>Guía SBOK™</i> .....	6
1.4	Marco de la <i>Guía SBOK™</i> .....	7
1.4.1	¿Cómo utilizar la <i>Guía SBOK™</i> ? .....	8
1.4.2	Principios de Scrum .....	8
1.4.3	Aspectos de Scrum .....	10
1.4.4	Procesos de Scrum .....	15
1.5	Scrum vs. Gestión de proyectos tradicional.....	19
2.	PRINCIPIOS .....	20
2.1	Introducción .....	20
2.2	Guía de roles .....	21
2.3	Control del proceso empírico.....	21
2.3.1	Transparencia .....	21
2.3.2	Inspección.....	22
2.3.3	Adaptación.....	23
2.4	Auto-organización .....	25
2.4.1	Beneficios de la auto-organización .....	25
2.5	Colaboración .....	27
2.5.1	Beneficios de la colaboración en los proyectos Scrum .....	28
2.5.2	La importancia de la co-ubicación en la colaboración .....	29
2.6	Priorización basada en valor .....	30
2.7	Bloque de tiempo asignado .....	32
2.7.1	Bloques de tiempo de Scrum .....	32
2.8	Desarrollo iterativo .....	34

2.9	Scrum vs. Gestión tradicional de proyectos.....	36
3.	ORGANIZACIÓN.....	37
3.1	Introducción .....	37
3.2	Guía de los roles.....	38
3.3	Roles de un proyecto Scrum .....	38
3.3.1	Roles centrales .....	38
3.3.2	Roles no centrales .....	40
3.4	Propietario del producto.....	42
3.4.1	Voz del cliente .....	43
3.4.2	Jefe propietario del producto.....	43
3.5	Scrum Master.....	44
3.5.1	Jefe Scrum Master .....	45
3.6	Equipo Scrum .....	46
3.6.1	Selección de personal.....	48
3.6.2	Tamaño del equipo Scrum.....	48
3.7	Scrum en proyectos, programas, y carteras .....	49
3.7.1	Definición de proyecto, programa, y cartera .....	49
3.7.2	Scrum en Proyectos .....	50
3.7.3	Scrum en carteras y programas.....	51
3.7.4	Mantenimiento de la participación de los socios.....	53
3.8	Resumen de responsabilidades .....	54
3.9	Scrum vs. Gestión de proyectos tradicional.....	55
3.10	Teorías de recursos humanos populares y su relevancia para Scrum .....	56
3.10.1	Modelo de dinámica de grupo de Tuckman.....	56
3.10.2	Gestión de conflictos .....	57
3.10.3	Técnicas de gestión de conflictos .....	57
3.10.4	Estilos de liderazgo .....	59
3.10.5	Teoría de jerarquía de necesidades de Maslow.....	61
3.10.6	Teoría X y Teoría Y .....	62
4.	JUSTIFICACIÓN DEL NEGOCIO .....	63

4.1	Introducción .....	63
4.2	Guía de roles .....	64
4.3	Entrega basada en valor.....	64
4.3.1	Responsabilidades del propietario del producto en la justificación del negocio....	66
4.3.2	Responsabilidades de otros roles de Scrum en la justificación del negocio .....	66
4.4	Importancia de la justificación del negocio .....	67
4.4.1	Factores que se utilizan para determinar la justificación del negocio .....	68
4.4.2	Justificación del negocio y el ciclo de vida del proyecto .....	68
4.5	Técnicas de justificación del negocio .....	70
4.5.1	Estimación del valor del proyecto .....	70
4.5.2	Planificar para el valor .....	72
4.5.3	Clasificación relativa de priorización .....	74
4.5.4	Mapeo de historias.....	74
4.6	Justificación continua de valor.....	74
4.6.1	Análisis del valor ganado .....	75
4.6.2	Diagrama de flujo acumulativo (DFA) .....	77
4.7	Confirmar la realización de beneficios.....	78
4.7.1	Prototipos, simulaciones y demostraciones.....	78
4.8	Resumen de responsabilidades .....	79
4.9	Scrum vs. Gestión tradicional de proyectos.....	80
5.	CALIDAD.....	81
5.1	Introducción .....	81
5.2	Guía de roles .....	82
5.3	Definición de calidad.....	82
5.3.1	Calidad y alcance .....	82
5.3.2	Calidad y valor del negocio.....	83
5.4	Criterios de aceptación y lista priorizada de pendientes del producto .....	84
5.4.1	Redacción de los criterios de aceptación .....	85
5.4.2	Criterios mínimos de aceptación.....	86
5.4.3	Definición de terminado.....	87

5.4.4	Aceptación o rechazo de elementos de la lista priorizada de pendientes del producto	88
5.5	Gestión de calidad en Scrum	88
5.5.1	Planificación de calidad	89
5.5.2	Control de calidad y garantía de calidad	90
5.5.3	Ciclo de planificar, hacer, verificar y actuar (Ciclo PDCA, por sus siglas en inglés)	91
5.6	Resumen de responsabilidades	92
5.7	Scrum vs. Gestión tradicional de proyectos	94
6.	Cambio	95
6.1	Introducción	95
6.2	Guía de los roles	96
6.3	Descripción	96
6.3.1	Solicitudes aprobadas y no aprobadas de cambio	97
6.4	Cambio en Scrum	98
6.4.1	Equilibrio entre flexibilidad y estabilidad	98
6.4.2	El logro de la flexibilidad	99
6.5	Integración del cambio	104
6.5.1	Los cambios a un sprint	104
6.6	Cambio en carteras y programas	110
6.6.1	En carteras	110
6.6.2	En programas	110
6.7	Resumen de responsabilidades	112
6.8	Scrum vs. Gestión de proyectos tradicional	113
7.	RIESGO	114
7.1	Introducción	114
7.2	Guía de roles	115
7.3	¿Qué es un riesgo?	115
7.3.1	Diferencia entre riesgos y problemas	115
7.3.2	Actitud de riesgo	116
7.4	Procedimiento de gestión de riesgos	117
7.4.1	Identificación de riesgos	117

7.4.2	Evaluación de riesgos .....	118
7.4.3	Priorización de riesgos.....	122
7.4.4	Mitigación de riesgos.....	123
7.4.5	Comunicación de riesgo .....	124
7.5	Minimizar riesgos por medio de Scrum .....	125
7.6	Riesgos en carteras y programas .....	126
7.6.1	En la cartera.....	126
7.6.2	En los programas .....	127
7.7	Resumen de responsabilidades .....	128
7.8	Scrum vs Gestión tradicional de proyectos.....	129
8.	INICIO .....	130
8.1	Creación de la visión del proyecto .....	134
8.1.1	Entradas.....	136
8.1.2	Herramientas.....	139
8.1.3	Salidas.....	141
8.2	Identificar al Scrum Master y al socio(s) .....	142
8.2.1	Entradas.....	144
8.2.2	Herramientas.....	146
8.2.3	Salidas.....	147
8.3	<i>Formación de un equipo Scrum</i> .....	148
8.3.1	Entradas.....	150
8.3.2	Herramientas.....	151
8.3.3	Salidas.....	152
8.4	Desarrollo de épica(s) .....	154
8.4.1	Entradas.....	155
8.4.2	Herramientas.....	158
8.4.3	Salidas.....	161
8.5	Creación de la lista priorizada de pendientes del producto .....	163
8.5.1	Entradas.....	165
8.5.2	Herramientas.....	166

8.5.3	Salidas.....	168
8.6	Realizar el plan de lanzamiento .....	170
8.6.1	Entradas.....	172
8.6.2	Herramientas.....	173
8.6.3	Salidas.....	174
8.7	Diagrama de la fase del flujo de datos.....	175
9.	PLANIFICACIÓN Y ESTIMACIÓN .....	176
9.1	Creación de historias de usuario.....	180
9.1.1	Entradas.....	182
9.1.2	Herramientas.....	183
9.1.3	Salidas.....	185
9.2	Aprobación, estimación y asignación de historias de usuarios .....	187
9.2.1	Entradas.....	188
9.2.2	Herramientas.....	188
9.2.3	Salidas.....	192
9.3	Creación de tareas .....	192
9.3.1	Entradas.....	193
9.3.2	Herramientas.....	194
9.3.3	Salidas.....	196
9.4	Estimación de tareas.....	197
9.4.1	Entradas.....	198
9.4.2	Herramientas.....	199
9.4.3	Salidas.....	200
9.5	Creación de la lista de pendientes del sprint.....	201
9.5.1	Entradas.....	202
9.5.2	Herramientas.....	203
9.5.3	Salidas.....	204
9.6	Diagrama de flujo de datos de fase .....	205
10.	IMPLEMENTACIÓN.....	206
10.1	Creación de entregables .....	210

10.1.1	Entradas.....	212
10.1.2	Herramientas.....	214
10.1.3	Salidas.....	215
10.2	Realizar reunión diaria de pie .....	217
10.2.1	Entradas.....	218
10.2.2	Herramientas.....	219
10.2.3	Salidas.....	220
10.3	Mantenimiento de la lista priorizada de pendientes del producto .....	222
10.3.1	Entradas.....	223
10.3.2	Herramientas.....	225
10.3.3	Salidas.....	226
10.4	Diagrama de flujo de datos de fase .....	227
11.	REVISIÓN Y RETROSPECTIVA .....	228
11.1	Convocar a un Scrum de Scrums.....	232
11.1.1	Entradas.....	233
11.1.2	Herramientas.....	234
11.1.3	Salidas.....	235
11.2	Demostración y validación del sprint.....	237
11.2.1	Entradas.....	239
11.2.2	Herramientas.....	240
11.2.3	Salidas.....	240
11.3	Retrospectiva del sprint .....	242
11.3.1	Entradas.....	243
11.3.2	Herramientas.....	243
11.3.3	Salidas.....	245
11.4	Diagrama de flujo de datos de fase .....	247
12.	LANZAMIENTO.....	248
12.1	Envío de entregables.....	251
12.1.1	Entradas.....	252
12.1.2	Herramientas.....	253

12.1.3	Salidas.....	253
12.2	Retrospectiva del proyecto.....	255
12.2.1	Entradas.....	256
12.2.2	Herramientas.....	256
12.2.3	Salidas.....	257
12.3	Diagrama de flujo de datos de fase.....	258
APÉNDICE A. RESUMEN DE ÁGIL.....		259
APÉNDICE B. AUTORES Y REVISORES DE LA GUÍA SBOK™.....		269
REFERENCIAS.....		271
GLOSARIO.....		272
INDEX.....		308

# LISTA DE FIGURAS

Figura 1-1: Flujo de Scrum para un Sprint .....	2
Figura 1-2: Marco de la <i>Guía SBOK™</i> .....	7
Figura 1-3: Principios de Scrum .....	9
Figura 1-4: Organización en Scrum .....	12
Figura 2-1: Transparencia en Scrum .....	22
Figura 2-2: Inspección de Scrum .....	23
Figura 2-3: Adaptación en Scrum.....	24
Figura 2-4: Retos en la gestión de proyectos tradicional.....	25
Figura 2-5: Objetivos de un equipo de auto-organizado .....	27
Figura 2-6: Beneficios de la colaboración en proyectos Scrum .....	29
Figura 2-7: Priorización basada en valor.....	31
Figura 2-8: Duración de los bloques de tiempo para las reuniones de Scrum .....	34
Figura 2-9: Scrum vs Cascada tradicional .....	35
Figura 3-1: Roles de Scrum—Descripción General .....	40
Figura 3-2: Las preguntas formuladas durante una reunión de Scrum de Scrum.....	46
Figura 3-3: Características deseadas de los roles principales de Scrum .....	48
Figura 3-4: Reunión de Scrum de Scrum (SoS) .....	51
Figura 3-5: Scrum en toda la organización para proyectos, programas y carteras .....	52
Figura 3-6: Etapas de Tuckman de desarrollo de grupos.....	56
Figura 3-7: Teoría de jerarquía de necesidades de Maslow .....	61
Figura 4-1: Entrega de valor en Scrum vs. proyectos tradicionales.....	65
Figura 4-2: Jerarquía de responsabilidades en la justificación del negocio.....	66
Figura 4-3: Justificación del negocio y el ciclo de vida del proyecto .....	69
Figura 4-4: Análisis de Kano .....	73
Figura 4-5: Ejemplo de diagrama de flujo acumulativo (DFA) .....	77
Figura 5-1: Diagrama de flujo del incremento del proyecto.....	85
Figura 5-2: Criterios de aceptación en cascada .....	87
Figura 5-3: Ciclo PDCA en Scrum .....	92
Figura 6-1: Ejemplo del proceso de aprobación de cambios.....	97
Figura 6-2: Actualización de la lista priorizada de pendientes del producto con los cambios aprobados .....	98
Figura 6-3: Características de Scrum para lograr flexibilidad .....	99
Figura 6-4: Motivación de los socios para la solicitud de cambios .....	100
Figura 6-5: Motivación del equipo principal de Scrum para la solicitud de cambios .....	101
Figura 6-6: Integración del cambio en Scrum .....	105
Figura 6-7: Impacto del cambio esperado en la duración del sprint .....	107
Figura 6-8: La incorporación de cambios en carteras y programas .....	111
Figura 7-1: Ejemplo de árbol de probabilidad .....	119
Figura 7-2: Ejemplo de un diagrama de Pareto .....	120
Figura 7-3: Muestra de matriz de probabilidad e impacto .....	121
Figura 7-4: Proceso de priorización de riesgos .....	123
Figura 7-5: Ejemplo de una gráfica de trabajo pendiente del riesgo ( <i>Risk Burndown Chart</i> ).....	125
Figura 8-1: Información general sobre el inicio .....	132
Figura 8-2: Resumen sobre el inicio (esenciales) .....	133

Figura 8-3: Creación de la visión del proyecto – Entradas, herramientas y salidas .....	134
Figura 8-4: Diagrama del flujo de datos de la creación de la visión del proyecto .....	135
Figura 8-5: El proceso de análisis de brecha .....	140
Figura 8-6: Identificación del Scrum Master y el(los) socio(s)—Entradas, herramientas y salidas .....	142
Figura 8-7: Identificación del Scrum Master y el(los) socio(s)—Diagrama de flujo de datos .....	143
Figura 8-8: Formación de un equipo Scrum—Entradas, herramientas y salidas .....	148
Figura 8-9: <i>Formación de un equipo Scrum</i> —Diagrama de flujo de datos .....	149
Figura 8-10: Desarrollo de épica(s)—Entradas, herramientas y salidas .....	154
Figura 8-11: Desarrollo de épica(s)—Diagrama de flujo .....	155
Figura 8-12: Creación de la lista priorizada de pendientes del producto—Entradas, herramientas y salidas ....	163
Figura 8-13: <i>Creación de la lista priorizada de pendientes del producto</i> — Diagrama de flujo de datos .....	164
Figura 8-14: Realizar la planificación del lanzamiento—Entrada, herramientas y salidas .....	170
Figura 8-15: <i>Realizar el plan de lanzamiento</i> —Diagrama de flujo de datos.....	171
Figura 8-16: Fase de inicio—Diagrama del flujo de datos .....	175
Figura 9-1: Resumen de la planificación y estimación .....	178
Figura 9-2: <i>Plan and Estimate</i> General (Esenciales) .....	179
Figura 9-3: Creación de historias de usuario—Entradas, herramientas y salidas.....	180
Figura 9-4: Creación de historias de usuario—Diagrama de flujo de datos .....	181
Figura 9-5: Aprobación, estimación y asignación de historias de usuario—Entrada, herramientas y salidas.....	187
Figura 9-6: Aprobación, estimación y asignación de historias de usuario— Diagrama de flujo de datos.....	187
Figura 9-7: Creación de tareas—Entradas, herramientas y salidas .....	192
Figura 9-8: Creación de tareas—Diagrama de flujo de datos .....	193
Figura 9-9: Reuniones de planificación de tareas .....	194
Figura 9-10: Estimación de tareas—Entradas, herramientas y salidas.....	197
Figura 9-11: Estimación de tareas—Diagrama de flujo de datos.....	197
Figura 9-12: Creación de la lista de pendientes del Sprint—Entradas, herramientas, y salidas.....	201
Figura 9-13: Creación de la lista de pendientes del sprint—Diagrama de flujo de datos.....	201
Figura 9-14: Fase de planificación y estimación—Diagrama de flujo de datos .....	205
Figura 10-1: Resumen de la implementación .....	208
Figura 10-2: Resumen de la implementación (Esenciales) .....	209
Figura 10-3: Creación de entregables—Entradas, herramientas y salidas .....	210
Figura 10-4: Creación de entregables—Diagrama de flujo de datos.....	211
Figura 10-5: Tablero de Scrum.....	212
Figura 10-6: Realizar reunión diaria de pie—Entradas, herramientas y salidas .....	217
Figura 10-7: Realizar reunión diaria de pie—Diagrama de flujo de datos.....	217
Figura 10-8: Mantenimiento de la lista priorizada de pendientes del producto—Entradas, herramientas y salidas .....	222
Figura 10-9: Mantenimiento de la lista priorizada de pendientes del producto—Diagrama de flujo de datos ..	223
Figura 10-10: Fase de implementación—Diagrama de flujo de datos.....	227
Figura 11-1: Resumen de la revisión y retrospectiva.....	230
Figura 11-2: Resumen de la revisión y retrospectiva (esenciales) .....	231
Figura 11-3: Convocar a un Scrum de Scrums—Entradas, herramientas y salidas.....	232
Figura 11-4: Convocar a un Scrum de Scrums—Diagrama de flujo de datos .....	232
Figura 11-5: Demostración y validación del sprint—Entradas, herramientas y salidas.....	237
Figura 11-6: Demostración y validación del sprint—Diagrama de flujo de datos .....	238

Figura 11-7: Retrospectiva del sprint—Entradas, herramientas y salidas .....	242
Figura 11-8: Retrospectiva del sprint—Diagrama de flujo de datos.....	242
Figura 11-9: Fase de revisión y retrospectiva—Diagrama de flujo de datos .....	247
Figura 12-1: Resumen del lanzamiento .....	249
Figura 12-2: Resumen del lanzamiento (Esenciales) .....	250
Figura 12-3: <i>Envío de entregables</i> —Entradas, Herramientas y Salidas .....	251
Figura 12-4: Envío de entregables—Diagrama de flujo de datos .....	251
Figura 12-5: Retrospectiva del proyecto—Entrada, herramientas y salidas .....	255
Figura 12-6: Retrospectiva del proyecto—Diagrama de flujo de datos.....	255
Figura 12-7: Fase de lanzamiento—Diagrama de flujo de datos .....	258

---

## LISTA DE TABLAS

Tabla 1-1: Resumen de los procesos de Scrum.....	15
Tabla 1-2: Scrum vs. Gestión de proyectos tradicional .....	19
Tabla 3-1: Responsabilidades del propietario del producto en los procesos de Scrum.....	43
Tabla 3-2: Responsabilidades del Scrum Master en los procesos de Scrum .....	45
Tabla 3-3: Responsabilidades del equipo Scrum en los procesos de Scrum .....	47
Tabla 3-4: Resumen de las responsabilidades pertinentes a la organización .....	55
Tabla 4-1: Fórmulas del valor ganado .....	75
Tabla 4-2: Resumen de las responsabilidades pertinentes a la justificación del negocio.....	79
Tabla 5-1: Resumen de las responsabilidades pertinentes a la calidad .....	93
Tabla 6-1: Resumen de las responsabilidades relacionadas con el Cambio.....	112
Tabla 7-1: Resumen de las responsabilidades pertinentes a riesgos.....	128

# 1. INTRODUCCIÓN

Una *Guía para el cuerpo de conocimiento de Scrum (Guía SBOK™)* proporciona las pautas para la implementación exitosa de la metodología Scrum—la gestión de proyectos Ágil y la metodología de desarrollo de productos más popular. Este libro proporciona un marco integral que incluye los principios, aspectos y procesos de Scrum.

Scrum, tal como se define en la *Guía SBOK™*, es se aplica a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria
- Productos, servicios o cualquier otro resultado que se les entregarán a los socios
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria—desde proyectos pequeños o equipos con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros de un equipo.

Este primer capítulo describe la finalidad y el marco de la *Guía SBOK™* y proporciona una introducción a los conceptos claves de Scrum. Contiene un resumen de los principios de Scrum, al igual que los aspectos y procesos del mismo. El capítulo 2 amplía la información sobre los seis principios de Scrum, que son la base del mismo. Los capítulos 3 al 7 tratan en detalle los cinco aspectos de Scrum que se deben abordar a través de cualquier proyecto: la organización, justificación del negocio, la calidad, el cambio y el riesgo. Los capítulos 8 al 12 cubren los 19 procesos de Scrum que forman parte de la creación de un proyecto Scrum. Estos procesos forman parte de las cinco fases de Scrum: Inicio; Planificación y Estimación; Implementación, Revisión y Retrospectiva; y Lanzamiento. Estas fases describen en detalle las entradas y salidas asociadas con cada proceso, así como las diferentes herramientas que pueden utilizarse en cada una. Algunas entradas, herramientas y salidas son obligatorias y estas se indican como tales; otras son opcionales dependiendo del proyecto específico, los requisitos de la organización y/o lineamientos establecidos por el *Cuerpo de asesoramiento de Scrum* de la organización (del inglés *Scrum Guidance Body* - SGB). Por último, el Apéndice A contiene una descripción general del Manifiesto Ágil (Fowler y Highsmith, 2001), y un análisis de los diversos métodos Ágiles para quienes quieran más información acerca de Ágil. Este capítulo está dividido en las siguientes secciones:

## 1.1 Información general de Scrum

## 1.2 ¿Por qué utilizar Scrum?

## 1.3 Propósito de la *Guía SBOK™*

## 1.4 Marco de la *Guía SBOK™*

## 1.5 Scrum vs. Gestión de Proyectos Tradicional

## 1.1 Información general de Scrum

Un proyecto Scrum implica un esfuerzo de colaboración para crear un nuevo producto, servicio, o cualquier otro resultado como se define en la Declaración de la visión del proyecto. Los proyectos se ven afectados por las limitaciones de tiempo, costo, alcance, calidad, recursos, capacidades organizativas, y otras limitaciones que dificultan su planificación, ejecución, administración y, finalmente, su éxito. Sin embargo, la implementación exitosa de los resultados de un proyecto terminado le proporciona ventajas económicas significativas a una organización. Por lo tanto, es importante que las organizaciones seleccionen y practiquen una metodología adecuada de gestión de proyectos.

Scrum es una de las metodologías Ágil más populares. Es una metodología de adaptación, iterativa, rápida, flexible y eficaz, diseñada para ofrecer un valor significativo de forma rápida en todo el proyecto. Scrum garantiza transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y de progreso continuo. El marco de Scrum, tal como se define en la *Guía SBOK™*, está estructurado de tal manera que es compatible con los productos y el desarrollo de servicios en todo tipo de industrias y en cualquier tipo de proyecto, independientemente de su complejidad.

Una fortaleza clave de Scrum radica en el uso de equipos interfuncionales, auto-organizados, y empoderados que dividen su trabajo en ciclos de trabajo cortos y concentrados llamados *Sprints*. La Figura 1-1 proporciona una visión general de flujo de un proyecto Scrum.

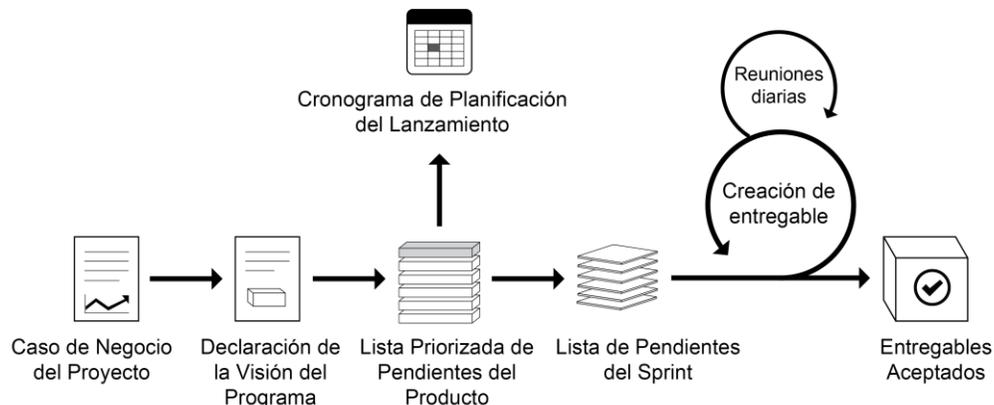


Figura 1-1: Flujo de Scrum para un Sprint

El ciclo de Scrum comienza con una reunión de los socios, durante la cual se crea la visión del proyecto. Después, el propietario del producto desarrolla una Lista priorizada de pendientes del producto que contiene una lista requerimientos del negocio por orden de importancia en forma de una Historia de usuario. Cada sprint comienza con una Reunión de planificación del sprint durante la cual se consideran las historias de usuario de alta prioridad para su inclusión en el sprint.

Un sprint suele durar entre una y seis semanas durante las cuales el equipo Scrum trabaja en la creación de Entregables (del inglés *deliverables*) en incrementos del producto potencialmente listos. Durante el sprint, se llevan cabo Reuniones diarias de pie muy breves y concretas (conocidas en inglés como *Daily Standup Meeting* —reuniones rápidas e informales en donde todos los asistentes están de pie a fin de que sean breves), en las que los miembros del equipo discuten progresos diarios. A medida que concluye el sprint, se lleva a cabo una Reunión de planificación del sprint en la cual se proporciona una demostración de los entregables al propietario del producto y a los socios relevantes. El propietario del producto acepta los entregables sólo si cumplen con los criterios de aceptación predefinidos. El ciclo del sprint termina con una Reunión de retrospectiva del sprint, donde el equipo presenta maneras para mejorar los procesos y el rendimiento a medida que avanzan al siguiente sprint.

### 1.1.1 Breve historia de Scrum

A mediados de la década de los 80s, Hirotaka Takeuchi y Ikujiro Nonaka definieron una estrategia de desarrollo de producto flexible e incluyente en la que el equipo de desarrollo trabaja como una unidad para alcanzar un objetivo común. Ellos describieron un enfoque innovador para el desarrollo de productos al que llamaron un enfoque holístico o "rugby", "en donde un equipo intenta llegar hasta el final como una unidad, pasando el balón hacia atrás y adelante". Ellos basaron su enfoque en los estudios de casos de diversas industrias de fabricación. Takeuchi y Nonaka propusieron que el desarrollo de productos no debe ser como una carrera de relevos secuencial, sino que debería ser análogo al del juego de rugby, en el que el equipo trabaja en conjunto, pasando el balón hacia atrás y hacia adelante a medida que se desplaza como una unidad por el campo. El concepto de rugby de un "Scrum" (donde un grupo de jugadores se junta para reiniciar el juego) se introdujo en este artículo para describir la propuesta de los autores de que el desarrollo de productos debe implicar "mover al Scrum campo abajo".

Ken Schwaber y Jeff Sutherland desarrollaron el concepto de Scrum y su aplicabilidad al desarrollo de software durante una presentación en la Conferencia internacional sobre programación, lenguajes y aplicaciones orientadas a objetos (llamada en inglés *Object-Oriented Programming, Systems, Languages & Applications*, o OOPSLA) en 1995 en Austin, Texas. Desde entonces, varios practicantes, expertos y autores de Scrum han seguido perfeccionando la conceptualización y metodología de Scrum. En los últimos años, Scrum ha aumentado en popularidad, y hoy en día es la metodología de desarrollo de proyectos predilecta de muchas organizaciones a nivel mundial.

## 1.2 ¿Por qué utilizar Scrum?

Algunas de las ventajas principales de la utilización de Scrum en cualquier proyecto son:

1. **Adaptabilidad**—El control del proceso empírico y el desarrollo iterativo hacen que los proyectos sean adaptables y abiertos a la incorporación del cambio.
2. **Transparencia**—Todos los radiadores de información tales como un Tablero de Scrum (del inglés *Scrumboard*) y una Gráfica del trabajo pendiente del sprint (del inglés *Sprint Burndown Chart*) se comparten, lo que conduce a un ambiente de trabajo abierto.
3. **Retroalimentación continua**—La retroalimentación continua se proporciona a través de los procesos llamados *Llevar a cabo la reunión diaria* y *Demostración y validación del sprint*.
4. **Mejora continua**—Los entregables se mejoran progresivamente sprint por sprint a través del proceso de *Mantenimiento de la lista priorizada de pendientes del producto* (del inglés *Groom Prioritized Producto Backlog*).
5. **Entrega continua de valor**—Los procesos iterativos permiten la entrega continua de valor tan frecuentemente como el cliente lo requiere a través del proceso de *Envío de entregables* (del inglés *Ship Deliverables*).
6. **Ritmo sostenible**—Los procesos Scrum están diseñados de tal manera que las personas involucradas pueden trabajar a un ritmo sostenible (del inglés *sustainable pace*) que, en teoría, se puede continuar indefinidamente.
7. **Entrega anticipada de alto valor**—El proceso de *Creación de la lista priorizada de pendientes del producto* asegura que los requisitos de mayor valor del cliente sean los primeros en cumplirse.
8. **Proceso de desarrollo eficiente**—La asignación de un bloque de tiempo fijo (del inglés *Time-boxing*) y la reducción al mínimo del trabajo que no es esencial conducen a mayores niveles de eficiencia.
9. **Motivación**—Los procesos de *Llevar a cabo la reunión diaria* y *Retrospectiva del sprint* conducen a mayores niveles de motivación entre los empleados.
10. **Resolución de problemas de forma más rápida**—La colaboración y colocación de equipos interfuncionales conducen a la resolución de problemas con mayor rapidez.
11. **Entregables efectivos**—El proceso de *Creación de la lista priorizada de pendientes del producto*, y las revisiones periódicas después de la creación de entregables aseguran entregas eficientes al cliente.
12. **Centrado en el cliente**—El poner énfasis en el valor del negocio y tener un enfoque de colaboración con los socios asegura un marco orientado al cliente.

13. **Ambiente de alta confianza**—Los procesos de *Llevar a cabo la reunión diaria* y la *Retrospectiva del sprint* promueven la transparencia y colaboración, dando lugar a un ambiente de trabajo de alta confianza que garantiza una baja fricción entre los empleados.
14. **Responsabilidad colectiva**—El proceso de *Aprobación, estimación y asignación de historias de usuarios* permite que los miembros del equipo hagan suyo el proyecto y su trabajo conlleve a una mejor calidad.
15. **Alta velocidad**—Un marco de colaboración que le permite a los equipos interfuncionales altamente cualificados alcanzar su potencial y alta velocidad.
16. **Ambiente innovador**—Los procesos de *Retrospectiva de sprint* y *Retrospectiva del proyecto* crean un ambiente de introspección, aprendizaje y capacidad de adaptación que conllevan a un ambiente de trabajo innovador y creativo.

### 1.2.1 Escalabilidad de Scrum

Para ser eficaces, los equipos *Scrum* deben tener idealmente de seis a diez miembros. Esta práctica resulta en la idea errónea de que el marco de Scrum sólo puede utilizarse para proyectos pequeños. Sin embargo, este marco puede aumentarse fácilmente para utilizarse de manera eficaz en proyectos grandes. En situaciones donde el tamaño del equipo Scrum excede diez personas, se pueden formar diversos equipos Scrum para trabajar en el proyecto. El proceso Scrum de Scrum facilita la coordinación entre los equipos Scrum, lo que permite una aplicación eficaz en los proyectos grandes.

Con frecuencia, los proyectos grandes o complejos se implementan como parte de un programa o cartera. El marco de Scrum también puede aplicarse para gestionar programas y carteras. El enfoque lógico de las directrices y los principios de este marco pueden utilizarse para gestionar proyectos de cualquier tamaño, que abarcan grandes geografías y organizaciones. Los proyectos grandes pueden tener múltiples equipos Scrum trabajando de forma paralela, por lo que es necesario sincronizarse y facilitar el flujo de información y mejorar la comunicación. El proceso de *Convocar el Scrum de Scrum* asegura esta sincronización. Los diversos equipos Scrum están representados en esta reunión y los objetivos son proporcionar actualizaciones sobre el progreso, discutir los retos que enfrentan durante el proyecto, y coordinar las actividades. No hay reglas fijas en cuanto a la frecuencia de estas reuniones. Los factores que determinan la frecuencia son la cantidad de dependencia entre los equipos, el tamaño del proyecto, el nivel de complejidad y las recomendaciones del Cuerpo de asesoramiento de Scrum.

## 1.3 Propósito de la *Guía SBOK™*

En los últimos años ha sido evidente que las organizaciones que utilizan Scrum como el marco de ejecución de proyectos preferido, obtienen altos rendimientos de inversión consistentemente. El enfoque de Scrum en la entrega impulsada por el valor ayuda a que los equipos Scrum ofrezcan resultados durante el proyecto tan pronto como les sea posible.

La *Guía SBOK™* ha sido desarrollada como un medio a fin de crear una guía necesaria para organizaciones y profesionales de gestión de proyectos que deseen implementar Scrum, así como para los que ya lo están haciendo y deseen mejorar sus procesos. Se basa en la experiencia adquirida de miles de proyectos a través de una variedad de organizaciones e industrias. Las contribuciones de muchos expertos en Scrum y profesionales de gestión de proyectos fueron tomadas en consideración en su desarrollo.

La *Guía SBOK™* es especialmente valiosa:

- para los miembros del equipo principal de Scrum incluyendo los siguientes:
  - Propietarios del producto que deseen comprender plenamente el marco de Scrum y particularmente las preocupaciones del cliente o socio que involucran la justificación del negocio, la calidad, el cambio, y los aspectos de riesgo asociados con los proyectos Scrum.
  - Scrum Masters que quieran aprender su papel específico en la supervisión de la aplicación del marco de Scrum en proyectos de este tipo.
  - Miembros del equipo Scrum que deseen comprender mejor los procesos de Scrum y las herramientas asociadas que pueden utilizarse para crear el producto o servicio del proyecto.
- como una guía integral para todos los profesionales que ya trabajan en proyectos Scrum en una organización o industria.
- como fuente de referencia para cualquier persona que interactúe con el equipo principal de Scrum, incluyendo, pero sin limitarse, al Propietario de la cartera del producto, Scrum Master, Scrum Master de la cartera, Propietario del programa del producto, Scrum Master del programa, Cuerpo de asesoramiento de Scrum, y Socios (tales como un patrocinador, cliente y usuarios).
- como un manual para cualquier persona que no tenga experiencia previa o conocimiento del marco de Scrum, pero quiera aprender más sobre el tema.

El contenido de la *Guía SBOK™* también es útil para las personas que se preparan para tomar los siguientes exámenes de certificación de SCRUMstudy™:

- Desarrollador Scrum Certificado (SDC™)
- Scrum Master Certificado (SMC™)
- Experto Ágil Certificado (AEC™)
- Propietario del Producto Scrum Certificado (SPOC™)
- Experto Scrum Master Certificado (ESMC™)

## 1.4 Marco de la *Guía SBOK™*

La *Guía SBOK™* se divide en las siguientes tres áreas:

1. Los **principios** contemplados en el capítulo 2 amplían sobre los seis principios que constituyen la fundación sobre la que se basa Scrum.
2. Los **aspectos** cubiertos en los capítulos 3 al 7 describen los cinco aspectos que se consideran importantes para todos los proyectos Scrum.
3. Los **procesos** cubiertos en los capítulos 8 al 12 incluyen los diecinueve procesos de Scrum y sus entradas, herramientas y salidas asociadas.

La Figura 1-2 ilustra el marco de la *Guía SBOK™*, lo que muestra que los principios, aspectos y procesos interactúan entre sí y son de igual importancia al tratar de obtener una mejor comprensión del marco de Scrum.

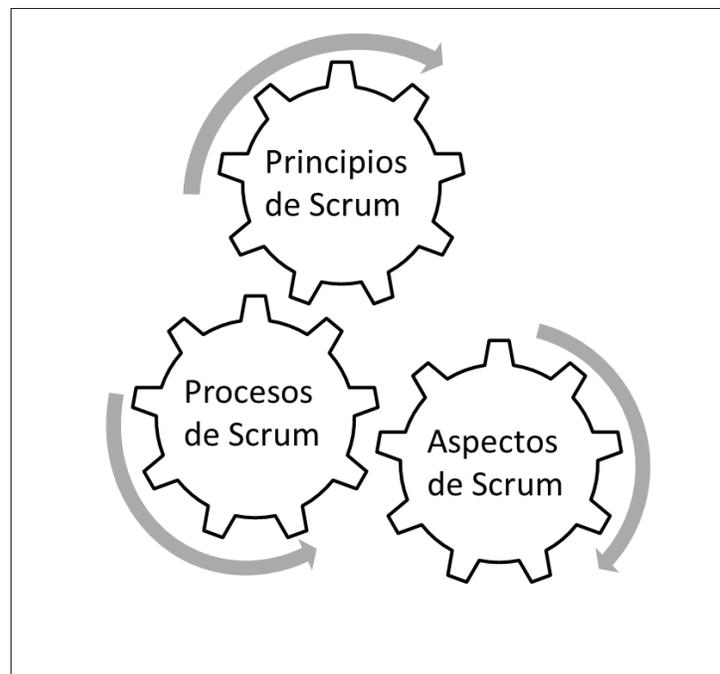


Figura 1-2: Marco de la *Guía SBOK™*

### 1.4.1 ¿Cómo utilizar la *Guía SBOK™*?

La *Guía SBOK™* puede utilizarse como una referencia y guía de conocimiento tanto por practicantes de Scrum con experiencia y otros profesionales de desarrollo de productos y servicios, como por personas sin experiencia previa o conocimiento de Scrum o de metodologías de gestión de proyectos. Los contenidos se organizan para facilitar la consulta de los tres roles centrales del equipo Scrum: Scrum Master, propietario del producto y equipo Scrum.

Los capítulos que abarcan los seis principios de Scrum (capítulo 2), y los cinco aspectos de Scrum (capítulos 3 al 7), incluyen una guía de los roles. Esta guía provee orientación sobre los roles del equipo principal de Scrum.

Con el fin de facilitar la mejor aplicación del marco de Scrum, la *Guía SBOK™* ha diferenciado claramente entre las entradas, las herramientas y las salidas obligatorias de las que son opcionales. Las entradas, herramientas y salidas que se indican con asteriscos (\*) son obligatorias, mientras que las otras sin asteriscos son opcionales. Se recomienda que las personas que recién aprenden sobre Scrum se centren principalmente en las entradas, las herramientas y las salidas obligatorias, mientras que los profesionales con más experiencia deben leer todos los capítulos del proceso.

### 1.4.2 Principios de Scrum

Los principios de Scrum son las pautas básicas para aplicar el marco de Scrum, y deben utilizarse obligatoriamente en todos los proyectos Scrum. Los seis principios de Scrum que se presentan en el capítulo 2 son los siguientes:

1. Control del proceso empírico
2. Auto-organización
3. Colaboración
4. Priorización basada en el valor
5. Asignación de un bloque de tiempo
6. Desarrollo iterativo

La Figura 1-3 ilustra los seis principios de Scrum.

1

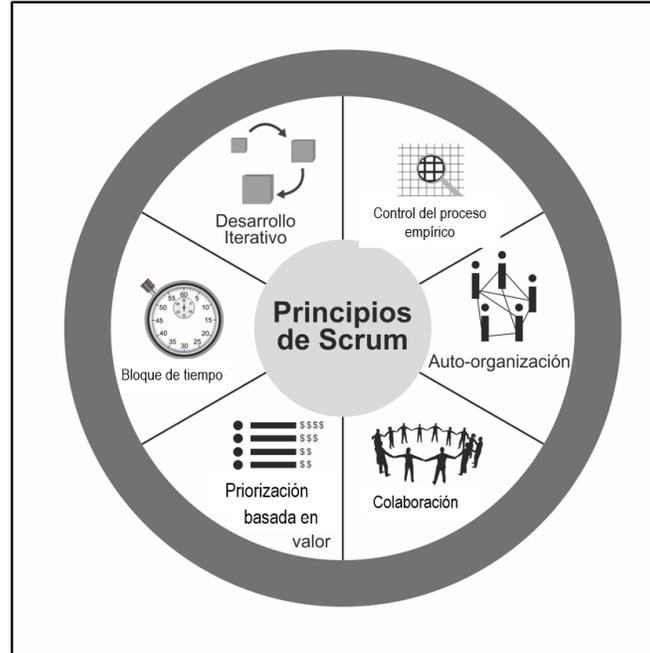


Figura 1-3: Principios de Scrum

Los principios de Scrum pueden aplicarse a cualquier tipo de proyecto en cualquier organización, y deben cumplirse a ellos a fin de garantizar la aplicación efectiva del marco de Scrum. Los principios de Scrum no están abiertos a la discusión ni pueden modificarse, y deben aplicarse como se especifica en la *Guía SBOK™*. El mantener los principios intactos y usarlos apropiadamente infunde confianza en el marco de Scrum con respecto al cumplimiento de los objetivos del proyecto. Los aspectos y procesos de Scrum, sin embargo, pueden modificarse para cumplir con los requisitos del proyecto o la organización.

1. **Control del proceso empírico**—Este principio pone de relieve la filosofía central de Scrum en base a las tres ideas principales de transparencia, inspección y adaptación.
2. **Auto-organización**—Este principio se centra en los trabajadores de hoy en día, que entregan un valor significativamente mayor cuando se organizan a sí mismos, lo cual resulta en equipos que poseen un gran sentido de compromiso y responsabilidad; a su vez, esto produce un ambiente innovador y creativo que es más propicio para el crecimiento.
3. **Colaboración**—Este principio se centra en las tres dimensiones básicas relacionadas con el trabajo colaborativo: conocimiento, articulación y apropiación. También fomenta la gestión de proyectos como un proceso de creación de valor compartido con equipos que trabajan e interactúan conjuntamente para ofrecer el mayor valor.
4. **Priorización basada en valor**—Este principio pone de relieve el enfoque de Scrum para ofrecer el máximo valor de negocio, desde el principio del proyecto hasta su conclusión.

5. **Asignación de un bloque de tiempo**—Este principio describe cómo el tiempo se considera una restricción limitante en Scrum, y cómo este se utiliza para ayudar a manejar eficazmente la planificación y ejecución del proyecto. Los elementos del bloque de tiempo en Scrum incluyen sprints, reuniones diarias de pie, reuniones de planificación del sprint, y reuniones de revisión del sprint.
6. **Desarrollo Iterativo**—Este principio define el desarrollo iterativo y enfatiza cómo manejar mejor los cambios y crear productos que satisfagan las necesidades del cliente. También delinea las responsabilidades del propietario del producto y las de la organización relacionadas con el desarrollo iterativo.

### 1.4.3 Aspectos de Scrum

Los aspectos de Scrum se deben abordar y gestionar durante todo un proyecto Scrum. Los cinco aspectos de Scrum que se presentan en los capítulos 3 al 7 son los siguientes:

#### 1.4.3.1 Organización

Entender los roles y responsabilidades definidos en un proyecto Scrum es muy importante a fin de asegurar la implementación exitosa del método de Scrum.

Los roles de Scrum se dividen en dos grandes categorías:

1. **Roles centrales**—Los roles centrales son aquellos que se requieren obligatoriamente para crear el producto o servicio del proyecto. Las personas a quienes se les asignan los roles centrales están plenamente comprometidas con el proyecto, y son las responsables del éxito de cada iteración del mismo, así como del proyecto en su totalidad.

Estas funciones incluyen:

- El **propietario del producto** es la persona responsable de lograr el máximo valor empresarial para el proyecto. Este rol también es responsable de la articulación de requisitos del cliente y de mantener la justificación del negocio para el proyecto. El propietario del producto representa la voz del cliente.
- El **Scrum Master** es un facilitador que asegura que el equipo Scrum esté dotado de un ambiente propicio para completar el proyecto con éxito. Este rol guía, facilita y les enseña las prácticas de Scrum a todos los involucrados en el proyecto; elimina los impedimentos que encuentra el equipo; y, asegura que se estén siguiendo los procesos de Scrum.
- El **equipo Scrum** es el grupo o equipo de personas responsables de la comprensión de los requisitos especificados por el propietario del producto y de la creación de los entregables del proyecto.

2. **Roles no centrales**—Los roles no centrales son los que no son obligatoriamente necesarios para el proyecto Scrum, y estos pueden incluir a miembros de los equipos que estén interesados en el proyecto. No tienen ningún papel formal en el equipo del proyecto, y pueden interactuar con el equipo, pero pueden no ser responsables del éxito del proyecto. Los roles deben tenerse en cuenta en cualquier proyecto de Scrum.

Los roles no centrales incluyen los siguientes:

- Los **socio(s)**, que es un término colectivo que incluye a clientes, usuarios y patrocinadores, con frecuencia interactúan con el equipo principal de Scrum, e influyen en el proyecto a lo largo de su desarrollo. Lo más importante es que el proyecto produzca beneficios de colaboración para los socios.
- El **cuerpo de asesoramiento de Scrum** es un rol opcional, que generalmente consiste en un conjunto de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de los objetivos relacionados con la calidad, las regulaciones gubernamentales, la seguridad y otros parámetros claves de la organización. El cuerpo guía el trabajo llevado a cabo por el propietario del producto, el Scrum Master y el equipo Scrum.
- Los **vendedores**, incluyendo a individuos u organizaciones externas, ofrecen productos y/o servicios que no están dentro de las competencias centrales de la organización del proyecto.
- El **jefe propietario del producto** es un rol para proyectos más grandes con múltiples equipos Scrum. Este rol se encarga de facilitar el trabajo del propietario del producto, y de mantener la justificación del negocio para el proyecto más grande.
- El **jefe Scrum Master** es el responsable de coordinar las actividades relacionadas con Scrum en proyectos grandes, los cuales pueden requerir que varios equipos Scrum trabajen paralelamente.

La Figura 1-4 ilustra la estructura de la organización Scrum.

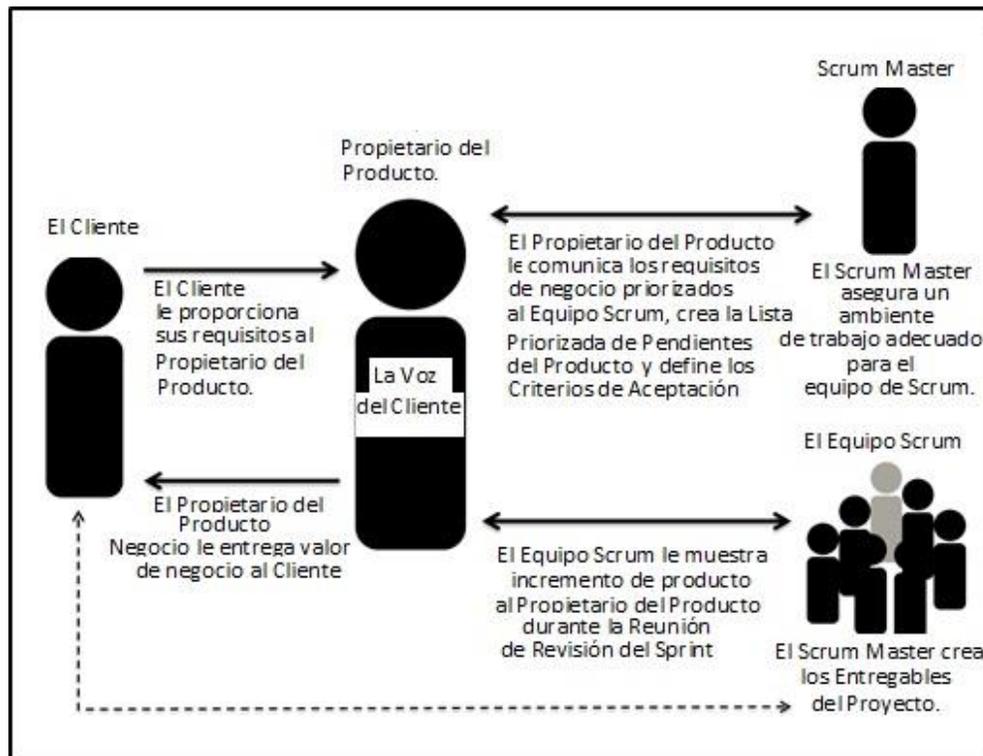


Figura 1-4: Organización en Scrum

El aspecto de Organización de Scrum también se ocupa de los requisitos de estructura del equipo para implementar Scrum en programas y carteras.

### 1.4.3.2 Justificación del negocio

Es importante para una organización llevar a cabo una evaluación adecuada del negocio antes de comenzar un proyecto. Esto ayuda a aquellas personas claves que son responsables de tomar decisiones a entender la necesidad de cambio en la empresa, o de un nuevo producto o servicio, al igual que a comprender la justificación para seguir adelante con un proyecto y su viabilidad.

En Scrum, la justificación del negocio se basa en el concepto de entrega impulsada por el valor. Una de las características claves de cualquier proyecto es la incertidumbre sobre los resultados. Es imposible garantizar el éxito de un proyecto, independientemente del tamaño o la complejidad del mismo. Teniendo en cuenta esta inseguridad de alcanzar el éxito, Scrum intenta iniciar la entrega de resultados lo antes posible en el proyecto. Esta entrega temprana de resultados, y por lo tanto de valor, proporciona una oportunidad para la reinversión y les demuestra el valor del proyecto a los socios.

La adaptabilidad de Scrum permite que los objetivos y procesos del proyecto cambien si cambia su justificación del negocio. Es importante señalar que, si bien el propietario del producto es el responsable principal de la justificación del negocio, otros miembros del equipo también contribuyen considerablemente.

### 1.4.3.3 Calidad

En Scrum, la calidad se define como la capacidad del producto o los entregables de cumplir con los criterios de aceptación y de alcanzar el valor de negocio que el cliente espera.

Para asegurar que un proyecto cumpla con los requisitos de calidad, Scrum adopta un enfoque de mejora continua mediante el cual el equipo aprende de sus experiencias y de la participación de los socios, a fin de actualizar constantemente la lista priorizada de pendientes del producto con cualquier cambio de requisitos. Dicha lista nunca se completa sino hasta el término o la conclusión del proyecto. Cualquier cambio en los requisitos debe reflejar los cambios en el entorno empresarial, ya sean internos o externos, y el equipo se debe adaptar continuamente a fin de alcanzar esos requisitos.

Puesto que Scrum requiere que el trabajo se realice en incrementos durante los sprints, esto hace que los errores o defectos se noten con más facilidad a través de pruebas de calidad repetitivas, y no simplemente cuando el producto final o servicio esté casi terminado. Por otra parte, las tareas relacionadas con la calidad (por ejemplo, desarrollo, prueba y documentación) se completan por el mismo equipo como parte del mismo sprint. Esto asegura que la calidad sea inherente a cualquier entregable que se crea como parte de un sprint. Dichas entregas de proyectos Scrum, que son potencialmente entregables, se les conoce como “terminado”.

Por lo tanto, la mejora continua con pruebas repetitivas optimiza la probabilidad de alcanzar los niveles esperados de calidad en un proyecto Scrum. Las conversaciones constantes entre el equipo principal de Scrum y los socios (incluyendo los clientes y los usuarios), junto con incrementos reales del producto que se entregan al final de cada sprint, aseguran que la diferencia entre las expectativas de los clientes del proyecto y los verdaderos entregables se reduzca constantemente.

El cuerpo de asesoramiento de Scrum también puede proporcionar directrices sobre la calidad que pueden ser de interés para todos los proyectos Scrum en la organización.

### 1.4.3.4 Cambio

Cada proyecto, independientemente del método o marco que se utilice, está expuesto a cambios. Es imperativo que los miembros del equipo del proyecto entiendan que los procesos de desarrollo de Scrum están diseñados para aceptar el cambio. Las organizaciones deben tratar de maximizar los beneficios que se derivan de los cambios, y minimizar cualquier impacto negativo a través de procesos de gestión de cambio diligentes, según los principios de Scrum.

Un principio fundamental de Scrum es su reconocimiento de que a) los socios [por ejemplo, clientes, usuarios y patrocinadores] cambian de opinión acerca de lo que quieren y necesitan durante el curso del

proyecto [a veces referido como "requisitos volátiles"]; y, b) que es muy difícil, si no imposible, que los socios definan todos los requisitos al inicio del proyecto.

Los proyectos Scrum reciben los cambios mediante el uso de los sprints cortos y repetitivos que incorporan la retroalimentación del cliente en cada entrega del sprint. Esto permite que el cliente interactúe regularmente con los miembros del equipo Scrum, vea los entregables a medida que estén listos, y cambie los requisitos si es necesario antes del siguiente sprint.

Asimismo, los equipos de gestión de programas o cartera pueden responder a las solicitudes de cambio pertenecientes a los proyectos Scrum aplicables a su nivel.

#### **1.4.3.5 Riesgo**

El riesgo se define como un evento o conjunto de eventos inciertos que pueden afectar los objetivos de un proyecto y pueden contribuir a su éxito o fracaso. A los riesgos que pueden tener un impacto positivo en el proyecto se les conoce como oportunidades, mientras que las amenazas son riesgos que podrían afectar al proyecto negativamente. Gestionar el riesgo debe hacerse de forma preventiva, y es un proceso iterativo que debe comenzar al inicio del proyecto y continuar a lo largo del ciclo de vida del mismo. El proceso de gestión de riesgos debe seguir algunos pasos estandarizados para asegurar que estos se identifiquen y evalúen, y que se determine un curso adecuado de acción y se proceda en consecuencia.

Los riesgos se deben identificar y evaluar, así como responderse a ellos en base a dos factores: la probabilidad de ocurrencia de cada riesgo, y el posible impacto en el caso de tal ocurrencia. Los riesgos con una alta probabilidad y valor de impacto (que se calculará multiplicando los dos factores), se deberán atender antes que los que tengan un valor relativamente bajo. En general, una vez que se detecte un riesgo, es importante entender el mismo en relación con las causas probables y los posibles efectos.

## 1.4.4 Procesos de Scrum

Los procesos de Scrum abordan las actividades y el flujo específico de un proyecto Scrum. En total hay diecinueve procesos que se agrupan en cinco fases. Estas fases se presentan en los capítulos 8 al 12 de la *Guía SBOK™*, como se muestra en la Tabla 1-1.

Capítulo	Fase	Procesos
8	Inicio	<ol style="list-style-type: none"> <li>1. Creación de la visión del proyecto</li> <li>2. Identificación del Scrum Master y el(los) socio(s)</li> <li>3. Formación de equipos Scrum</li> <li>4. Desarrollo de épica(s)</li> <li>5. Creación de la lista priorizada de pendientes del producto</li> <li>6. Realizar la planificación de lanzamiento</li> </ol>
9	Planificación y estimación	<ol style="list-style-type: none"> <li>7. Creación de historias de usuario</li> <li>8. Aprobación, estimación y asignación de historias de usuario</li> <li>9. Creación de tareas</li> <li>10. Estimación de tareas</li> <li>11. Creación de la lista de pendientes del sprint</li> </ol>
10	Implementación	<ol style="list-style-type: none"> <li>12. Creación de entregables</li> <li>13. Llevar a cabo la reunión diaria</li> <li>14. Mantenimiento de la lista priorizada de pendientes del producto</li> </ol>
11	Revisión y retrospectiva	<ol style="list-style-type: none"> <li>15. Convocar el Scrum de Scrums</li> <li>16. Demostración y validación del sprint</li> <li>17. Retrospectiva del sprint</li> </ol>
12	Lanzamiento	<ol style="list-style-type: none"> <li>18. Envío de entregables</li> <li>19. Retrospectiva del proyecto</li> </ol>

**Tabla 1-1: Resumen de los procesos de Scrum**

Las fases describen en detalle cada proceso, incluyendo sus entradas, herramientas y salidas asociadas. En cada proceso, algunas entradas, herramientas y salidas son obligatorias (las que tienen un asterisco [\*] después de sus nombres), mientras que otras son opcionales. La inclusión de las entradas, herramientas y/o salidas opcionales dependerá del proyecto en particular, organización o industria. Las entradas, herramientas y salidas indicadas como obligatorias son importantes para la implementación exitosa de Scrum en cualquier organización.

#### 1.4.4.1 Inicio

1. Creación de la visión del proyecto—En este proceso, se revisa el caso de negocio del proyecto a fin de crear una declaración de la visión del proyecto que servirá de inspiración y proporcionará un enfoque para todo el proyecto. En este proceso se identifica al propietario del producto.
2. Identificación del Scrum Master y el(los) socio(s)—En este proceso, se identifica al Scrum Master y al socio utilizando criterios de selección específicos.
3. Formación de equipos Scrum—En este proceso, se identifica a los miembros del equipo Scrum. Normalmente, el propietario del producto es el responsable principal de la selección de los miembros del equipo, pero con frecuencia lo hace en colaboración con el Scrum Master.
4. Desarrollo de épica(s)—En este proceso, la declaración de la visión del proyecto sirve como la base para el desarrollo de épicas. Se pueden llevar a cabo reuniones de grupos de usuarios para hablar sobre las épicas que sean adecuadas.
5. Creación de la lista priorizada de pendientes del producto—En este proceso, se refinan y crean las épicas, y luego se priorizan para crear una lista priorizada de pendientes del producto. En ese momento también se establecen los criterios de terminado.
6. Realizar la planificación del lanzamiento—En este proceso, el equipo principal de Scrum revisa las historias de usuario en la lista priorizada de pendientes del producto para desarrollar un cronograma de planificación del lanzamiento, que es esencialmente un programa de implementación por fases que se puede compartir con los socios del proyecto. También se determina la duración del sprint en este proceso.

#### 1.4.4.2 Planificación y estimación

7. Creación de historias de usuario—En este proceso, se crean las historias de usuario y los criterios de aceptación de las historias de usuario. Las historias de usuario son generalmente escritas por el propietario del producto, y están diseñadas para asegurar que los requisitos del cliente estén claramente representados y puedan ser plenamente comprendidos por todos los socios. Se pudieran llevar a cabo ejercicios de escritura de historias de usuario, lo cual involucra a los miembros del equipo Scrum, resultando en la creación de dichas historias. Estas se incorporan a la lista priorizada de pendientes del producto.
8. Aprobación, estimación y asignación de historias de usuario—En este proceso, el propietario del producto aprueba las historias de usuario para un sprint. Luego, el Scrum Master y el equipo Scrum estiman el esfuerzo necesario para desarrollar la funcionalidad descrita en cada historia de usuario, y el equipo Scrum se compromete a entregar los requisitos del cliente en forma de historias de usuario aprobadas, estimadas y asignadas.

9. Creación de tareas—En este proceso, las historias de usuario aprobadas, estimadas y asignadas se dividen en tareas específicas y se compilan en una lista de tareas. Con frecuencia, una reunión de planificación de tareas se convocará para este efecto.
10. Estimación de tareas—En este proceso, durante las reuniones de planificación de tareas, el equipo Scrum estima el esfuerzo necesario para realizar cada tarea en la lista. El resultado de este proceso es una lista de tareas de esfuerzo estimado.
11. Creación de la lista de pendientes del sprint—En este proceso, el equipo principal de Scrum lleva a cabo una reunión de planificación del sprint donde el grupo crea una lista priorizada de pendientes del Sprint, que contiene todas las tareas que deben completarse en el sprint.

#### 1.4.4.3 Implementación

12. Creación de entregables—En este proceso, el equipo Scrum trabaja en las tareas de la lista priorizada de pendientes del sprint para crear los entregables del sprint. Se utiliza a menudo un tablero de Scrum para realizar el seguimiento del trabajo y las actividades que se realizan. Las cuestiones o problemas que enfrenta el equipo Scrum pudieran actualizarse en un registro de impedimentos.
13. Realizar reunión diaria de pie—En este proceso, se lleva a cabo diariamente una reunión muy centrada que se asigna a un bloque de tiempo fijo, llamada reunión diaria de pie. Es aquí donde los miembros del equipo Scrum se actualizan el uno al otro referente a sus progresos y sobre los impedimentos que pudieran enfrentar.
14. Mantenimiento de la lista priorizada de pendientes del producto—En este proceso, la lista priorizada de pendientes del producto se actualiza y se mantiene continuamente. Se puede considerar realizar una reunión de revisión de la lista priorizada de pendientes del producto, en la que se discute y se incorpora la lista priorizada de pendientes del producto de forma adecuada.

#### 1.4.4.4 Revisión y retrospectiva

15. Convocar el Scrum de Scrums—En este proceso, representantes del equipo Scrum convocan un Scrum of Scrums en intervalos predeterminados, o cuando sea necesario, para colaborar y realizar un seguimiento de su respectivo progreso, los impedimentos, y las dependencias entre otros equipos. Esto es relevante sólo para grandes proyectos en los que participan varios equipos Scrum.
16. Demostración y validación del sprint—En este proceso, el equipo Scrum le demuestra el entregable del sprint al propietario del producto y a los socios relevantes durante una reunión de revisión del

sprint. El propósito de esta reunión es asegurar la aprobación y aceptación del propietario del producto de los entregables creados en el sprint.

17. Retrospectiva de Sprint—En este proceso, el Scrum Master y el equipo Scrum se reúnen para discutir las lecciones aprendidas durante todo el Sprint. Esta información se documenta como lecciones aprendidas que pueden aplicarse a los futuros sprints. Frecuentemente, como resultado de esta discusión, puede haber mejoras accionables aceptadas o recomendaciones actualizadas por parte del cuerpo de asesoramiento de Scrum.

#### **1.4.4.5 Lanzamiento**

18. Envío de entregables—En este proceso, los entregables que se aceptan se entregan o pasan a los socios relevantes. Un acuerdo formal de entregables funcionando documenta la finalización con éxito del sprint.
19. Retrospectiva del proyecto—En este proceso, mismo que completa el proyecto, los socios y miembros del equipo principal de Scrum se reúnen para hacer una retrospectiva del proyecto e identificar, documentar e internalizar las lecciones aprendidas. A menudo, estas lecciones llevan a la documentación de mejoras accionables aceptadas, que se aplicarán en futuros proyectos.

## 1.5 Scrum vs. Gestión de proyectos tradicional

En la tabla 1-2 se resumen muchas de las diferencias entre los modelos tradicionales de gestión de proyectos y Scrum.

	Scrum	Gestión de proyectos tradicional
El énfasis está en	Personas	Procesos
Documentación	Sólo mínima—según se requiera	Exhaustiva
Estilo de procesos	Iterativo	Lineal
Planificación por adelantado	Baja	Alta
Priorización de los requisitos	Según el valor del negocio y regularmente actualizada	Fijo en el plan de proyecto
Garantía de calidad	Centrada en el cliente	Centrada en el proceso
Organización	Auto-organizada	Gestionada
Estilo de gestión	Descentralizado	Centralizado
Cambio	Actualizaciones a la lista priorizada de pendientes del producto	Sistema formal de gestión del cambio
Liderazgo	Liderazgo colaborativo y servicial	Mando y control
Medición del rendimiento	El valor del negocio	Conformidad con el plan
Retorno sobre la inversión (RSI)	Al comienzo y a lo largo del proyecto	Al final del proyecto
Participación del cliente	Alta durante todo el proyecto	Varía en función del ciclo de vida del proyecto

Tabla 1-2: Scrum vs. Gestión de proyectos tradicional

## 2. PRINCIPIOS

### 2.1 Introducción

Los principios de Scrum son el fundamento sobre el que se basa su marco. Estos principios pueden aplicarse a cualquier tipo de proyecto u organización, y deben respetarse a fin de garantizar la aplicación adecuada de Scrum. Los aspectos y procesos de Scrum pueden modificarse para cumplir con los requerimientos del proyecto, o la organización que lo usa, pero sus principios no están abiertos a la discusión ni pueden modificarse, y deben aplicarse como se describe en el marco presentado en *Una guía para el conocimiento de Scrum (Guía SBOK™)*. El mantener los principios intactos y usarlos apropiadamente infunde confianza en el usuario del marco de Scrum respecto al cumplimiento de los objetivos del proyecto. Los principios se consideran los lineamientos básicos para la aplicación del marco de Scrum.

Los principios, tales como se definen en la *Guía SBOK™*, son aplicables a lo siguiente:

- Carteras, programas, y/o proyectos de *cualquier* industria
- Productos, servicios, o cualesquier otros resultados que se les entregarán a los socios
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria—desde proyectos pequeños o equipos con tan sólo seis miembros, hasta proyectos grandes y complejos con hasta varios cientos de miembros por equipo.

Este capítulo está dividido en las siguientes secciones:

**2.2 Guía de los roles**—Aquí se describe qué sección o subsección es más relevante para cada uno de los roles centrales de Scrum tales como el propietario del producto, Scrum Master, y equipo Scrum.

**2.3 Control del proceso empírico**—Esta sección describe el primer principio de Scrum y las tres ideas principales de transparencia, inspección y adaptación.

**2.4 Auto-organización**—Esta sección destaca el segundo principio de Scrum que se enfoca en los trabajadores de hoy en día, quienes entregan un valor significativamente mayor cuando se auto-organizan. Esta auto-organización se traduce en un mejor sentido de compromiso del equipo y responsabilidad compartida; esto a su vez proporciona un ambiente de trabajo innovador y creativo que es más propicio para el crecimiento.

**2.5 Colaboración**—Esta sección hace hincapié en el tercer principio de Scrum donde el desarrollo de los productos es un proceso de creación de valor compartido que necesita que todos los socios trabajen e

interactúen en conjunto para ofrecer el mayor valor. También se centra en las dimensiones básicas de trabajo colaborativo: la sensibilización, articulación y apropiación.

**2.6 Priorización basada en el valor**—Esta sección presenta el cuarto principio de Scrum, que pone de relieve la unidad del marco de Scrum para entregar el máximo valor empresarial en un período de tiempo.

**2.7 Asignación de un bloque de tiempo**—Aquí se explica el quinto principio de Scrum que trata al tiempo como un factor limitante. También cubre el sprint, la reunión diaria de pie, y otros sprints relacionados con las reuniones, tales como la reunión diaria de planificación del sprint y la reunión de revisión del sprint, las cuales están bajo un bloque de tiempo asignado.

**2.8 Desarrollo iterativo**—En esta sección se aborda el sexto principio de Scrum, que hace hincapié en que el desarrollo iterativo ayuda a gestionar mejores cambios y crear productos que satisfagan las necesidades del cliente.

**2.9 Scrum vs. Gestión de Proyecto Tradicional**—Esta sección destaca las principales diferencias entre los principios de Scrum y los principios de gestión tradicional de proyectos (modelo en cascada), y explica cómo Scrum funciona mejor en el mundo tan cambiante de hoy en día.

## 2.2 Guía de roles

Todas las secciones de este capítulo son importantes para todos los roles del equipo principal de Scrum — propietario del producto, Scrum Master, y equipo Scrum. Es esencial que exista una comprensión clara de los principios de Scrum por parte de todos socios para que el marco de Scrum sea exitoso en cualquier organización.

## 2.3 Control del proceso empírico

En Scrum, las decisiones se basan en la observación y la experimentación, más que en la planificación inicial detallada. El control del proceso empírico se basa en las tres ideas principales de la transparencia, inspección y adaptación.

### 2.3.1 Transparencia

La transparencia permite que todas las facetas de cualquier proceso de Scrum se observen por cualquier persona. Esto promueve un flujo de información fácil y transparente en toda la organización y crea una cultura de trabajo abierta. En Scrum, la transparencia es representada a través de las siguientes acciones:

- Una declaración de la visión del proyecto que puede ser vista por todos los socios y el equipo Scrum
- Un proceso abierto de la lista priorizada de pendientes del producto con historias de usuario priorizadas que se pueden ser vistas por todos, tanto dentro como fuera del equipo Scrum

- Un cronograma de planificación del lanzamiento que se puede coordinar a través de múltiples equipos Scrum
- Clara visibilidad sobre el progreso del equipo a través del uso de un tablero Scrum, gráfica del trabajo pendiente y otros radiadores de información
- Las reuniones diarias se conducen durante el proceso de realizar reuniones diarias en las que todos los miembros del equipo informan acerca de lo que han hecho el día anterior, lo que van a hacer hoy, y cualquier problema que les impida completar sus tareas en el sprint actual
- Las reuniones de revisión del sprint se llevan a cabo durante el proceso de demostración y validación del sprint, en el que el equipo Scrum les muestra los entregables del sprint que potencialmente se pueden enviar a los propietarios del producto y a los socios

La Figura 2-1 resume el concepto de *transparencia* en Scrum.

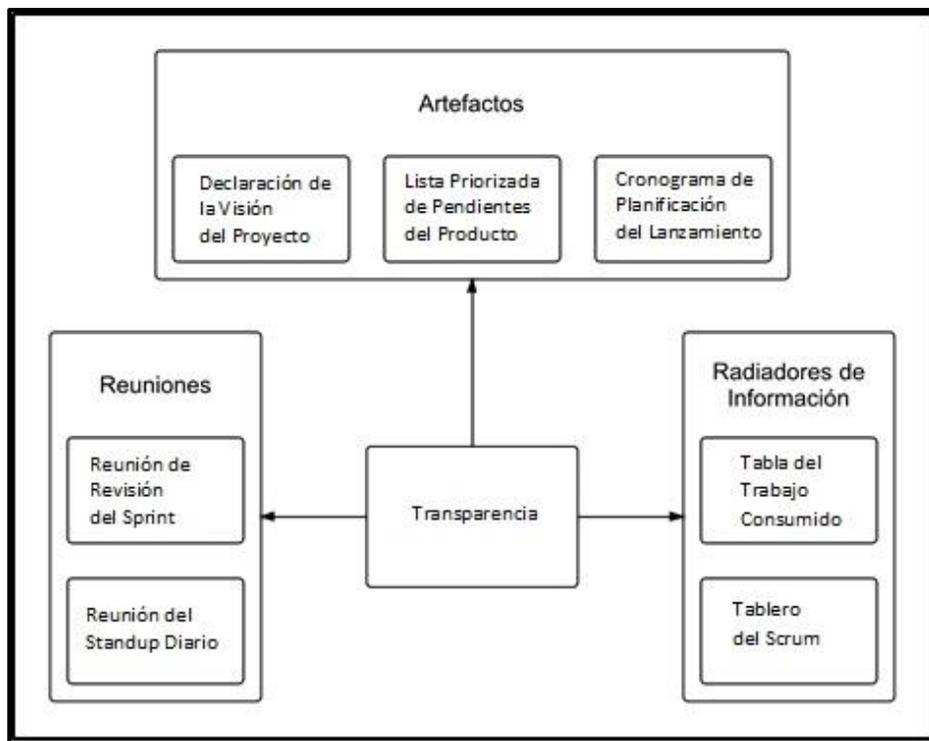


Figura 2-1: Transparencia en Scrum

### 2.3.2 Inspección

La inspección en Scrum se representa a través de las siguientes acciones:

- Utilización de un tablero Scrum común y otros radiadores de información que muestran el progreso del equipo Scrum en completar las tareas del sprint actual.

- Recopilación de la retroalimentación del cliente y otros socios durante los procesos de desarrollo de épica(s), creación de la lista priorizada de pendientes del producto, y de realizar el plan de lanzamiento.
- Inspección y aprobación de los entregables por el propietario del producto y el cliente en el proceso de demostración y validación del sprint.

La Figura 2-2 resume el concepto de inspección en Scrum.

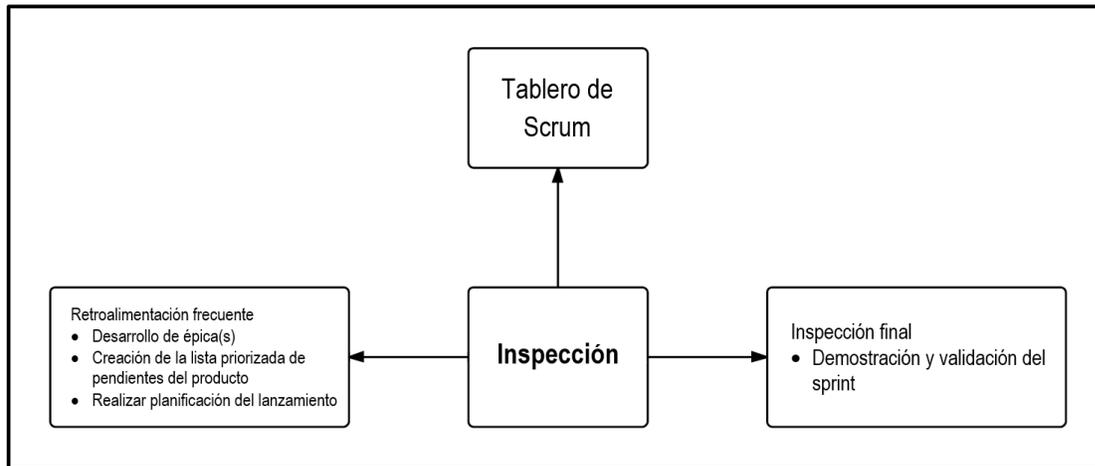


Figura 2-2: Inspección de Scrum

### 2.3.3 Adaptación

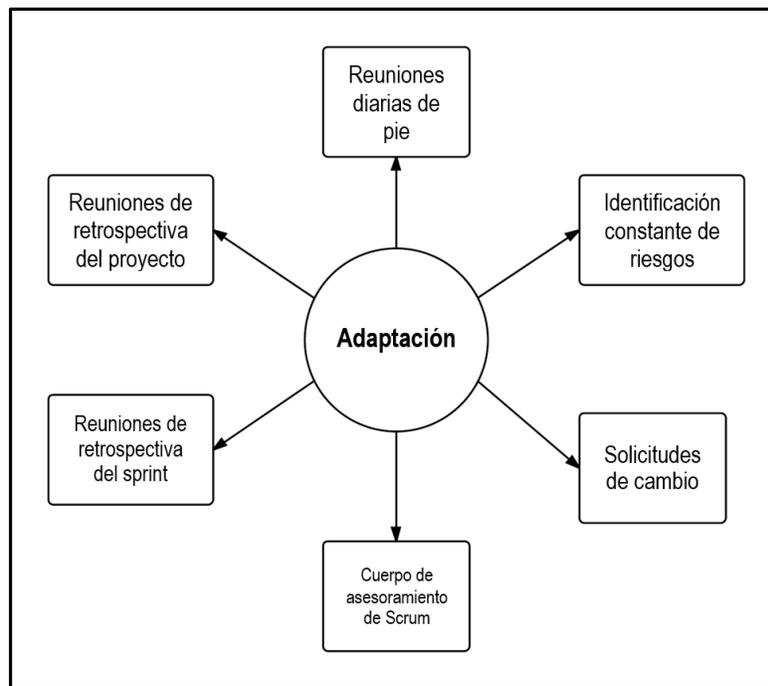
La adaptación sucede cuando el equipo principal de Scrum y los socios aprenden mediante la transparencia y la inspección, y luego se adaptan al hacer mejoras en el trabajo que llevan a cabo. Algunos ejemplos de adaptación incluyen:

- En las reuniones diarias, los miembros del equipo Scrum hablan abiertamente sobre los impedimentos para completar sus tareas, y buscan la ayuda de otros miembros del equipo. Los miembros con más experiencia en el equipo Scrum fungen como mentores de aquellos quienes tienen relativamente menos experiencia y conocimiento del proyecto o de la tecnología.
- Se lleva a cabo la identificación del riesgo y se repite a lo largo del proyecto. Los riesgos que se identifican se convierten en entradas para varios procesos de Scrum, incluyendo el de creación de la lista priorizada de pendientes del producto, de mantenimiento de la lista priorizada de pendientes del producto, y de demostración y validación del sprint.
- Las mejoras pueden resultar en solicitudes de cambios que se discuten y aprueban durante los procesos de desarrollo de épica(s), la creación de la lista priorizada de pendientes del producto, y el mantenimiento de la lista priorizada de pendientes del producto.
- El cuerpo de asesoramiento de Scrum interactúa con los miembros del equipo Scrum durante los procesos de creación de historias de usuario, estimación de tareas, creación de entregables y

mantenimiento de la lista priorizada de pendientes del producto para ofrecer orientación y también proporcionar conocimientos según sea necesario.

- En el proceso de retrospectiva del sprint se determinan las mejoras accionables aceptadas en base a las salidas del proceso de demostración y validación del sprint.
- En la reunión de retrospectiva del proyecto, los participantes documentan las lecciones aprendidas y realizan revisiones en busca de oportunidades para mejorar los procesos y abordar las ineficiencias.

La Figura 2-3 resume el concepto de adaptación en Scrum.



**Figura 2-3: Adaptación en Scrum**

Con otros métodos, como el modelo tradicional en Cascada, se requiere de una planificación considerable que debe hacerse por adelantado, y el cliente generalmente no revisa los componentes del producto hasta casi el final de una fase o del proyecto. Con frecuencia este método presenta enormes riesgos al éxito del proyecto, ya que tiene más potencial para impactar de forma significativa la entrega de proyectos y la aceptación del cliente. La interpretación y comprensión del cliente sobre el producto final puede ser muy diferente de lo que realmente se entendió originalmente y fue producido por el equipo, algo que no se sabría hasta muy tarde en el desarrollo del proyecto.

La Figura 2-4 muestra un ejemplo de estos desafíos.

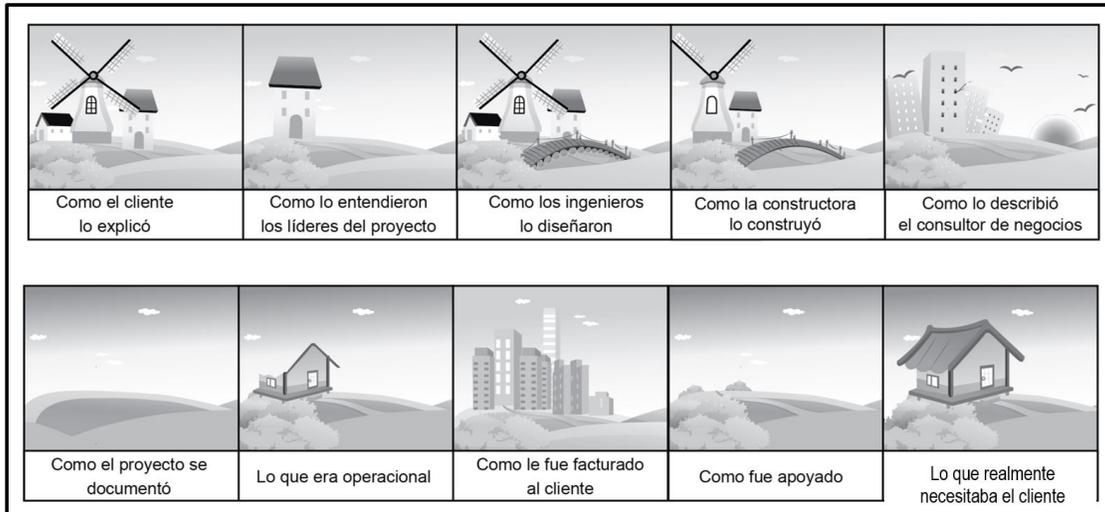


Figura 2-4: Retos en la gestión de proyectos tradicional

## 2.4 Auto-organización

Scrum sostiene que los empleados cuentan con motivación propia y que buscan aceptar mayores responsabilidades. Por tanto, ellos ofrecen mucho más valor cuando se organizan por cuenta propia.

El estilo de liderazgo preferido en Scrum es el de “liderazgo servicial”, el cual enfatiza en la obtención de resultados, centrándose en las necesidades del equipo Scrum. Consulte la sección 3.10.3, en donde se describen varios estilos de liderazgo y gestión.

### 2.4.1 Beneficios de la auto-organización

La auto-organización, como un principio esencial en Scrum, conduce a:

- Un sentido de compromiso del equipo y responsabilidad compartida
- La motivación, la que conduce a un nivel de mejor rendimiento del equipo
- Un ambiente de trabajo innovador y creativo que conduce al crecimiento

La auto-organización no significa que a los miembros del equipo se les permita actuar de la manera que deseen. Sólo significa que una vez que la visión del producto se define en el proceso de creación de la visión del proyecto, se identifica al propietario del producto, al Scrum Master y al Equipo Scrum. Asimismo, el equipo principal de Scrum trabaja estrechamente con los socios para perfeccionar los requisitos a medida que avanzan a través de los procesos de desarrollo de épica(s) y de creación de historias de usuario. El conocimiento se utiliza para evaluar las entradas necesarias para realizar la obra prevista del proyecto. Este juicio y experiencia se aplican a todos los aspectos técnicos y de gestión de proyectos durante el proceso de la creación de entregables.

Aunque la priorización se hace sobre todo por el propietario del producto, quien representa la voz del cliente, el equipo Scrum auto-organizado se involucra en la distribución y estimación de tareas durante los procesos de creación de tareas y estimación de tareas. Durante estos procesos, cada miembro del equipo tiene la responsabilidad de determinar qué tipo de trabajo va a hacer. Durante la ejecución de un sprint, los miembros del equipo pueden necesitar cierta ayuda para completar sus tareas. Scrum se ocupa de esto a través de la interacción regular obligatoria en las reuniones diarias. El equipo Scrum sí interactúa con otros equipos a través de las reuniones de Scrum of Scrums y, si es necesario, puede buscar orientación adicional del cuerpo de asesoramiento de Scrum.

Por último, el equipo Scrum y el Scrum Master trabajan muy de cerca para demostrar el incremento del producto creado durante el sprint en el proceso de demostración y validación del sprint, donde los entregables hechos de forma adecuada se aceptan. Dado que los entregables son potencialmente enviables, (y la lista priorizada de pendientes del producto se prioriza por historias de usuario en el orden del valor creado por ellas), el propietario del producto y el cliente pueden visualizar y articular claramente el valor creado después de cada sprint; y el equipo Scrum a su vez tiene la satisfacción de obtener la aceptación de su parte del cliente y los otros socios.

Los principales objetivos de los equipos auto-organizados son los siguientes:

- Entender la visión del proyecto y por qué el proyecto aporta valor a la organización
- Estimar historias de usuario durante el proceso de aprobación, estimación y asignación de historias de usuario, y asignar tareas a sí mismos durante el proceso de creación de la lista de pendientes del sprint
- Crear tareas de forma independiente durante el proceso de creación de tareas
- Aplicar y aprovechar la experiencia de ser un equipo interfuncional al trabajar en las tareas durante el proceso de creación de entregables
- Lograr resultados tangibles que sean aceptados por el cliente y otros socios durante el proceso de demostración y validación del sprint
- Resolver problemas individuales al discutirlos durante las reuniones diarias
- Aclarar cualquier discrepancia o duda y tener la disposición de aprender cosas nuevas
- Actualizar los conocimientos y habilidades de manera continua a través de interacciones regulares dentro del equipo
- Mantener la estabilidad de los miembros del equipo durante toda la duración del proyecto al no cambiar los miembros, a menos que esto sea inevitable

La Figura 2-5 ilustra los objetivos de un equipo auto-organizado.

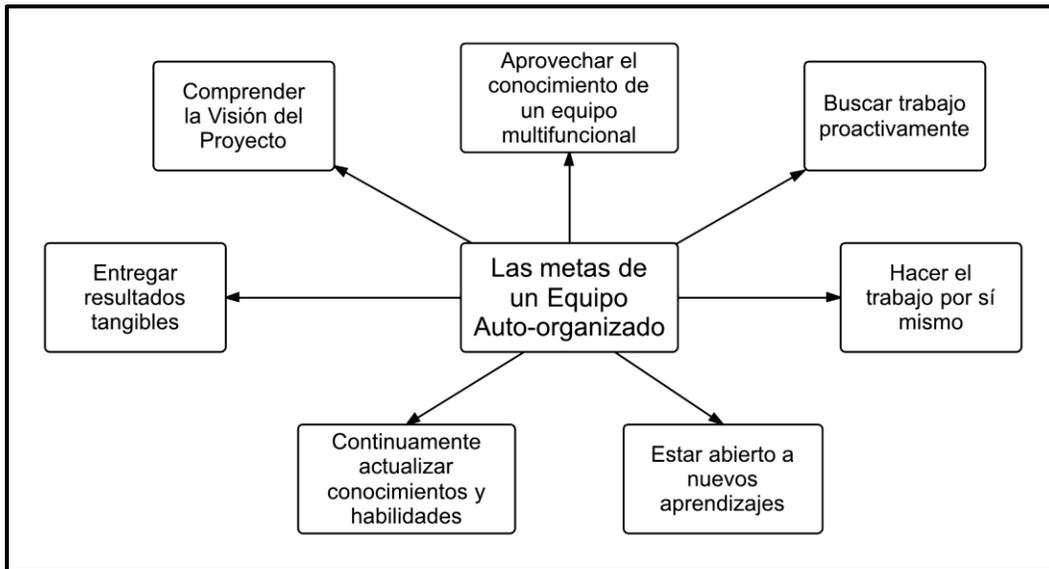


Figura 2-5: Objetivos de un equipo de auto-organizado

## 2.5 Colaboración

La colaboración en Scrum se refiere a que el equipo principal de Scrum trabaja e interactúa junto con los socios para crear y validar los resultados del proyecto, a fin de cumplir con los objetivos que se plantean en la visión del proyecto. Es importante tener en cuenta la diferencia entre la cooperación y colaboración. La cooperación ocurre cuando el trabajo que se produce consiste en la suma de los esfuerzos del trabajo de varias personas en un equipo. La colaboración, en cambio, se produce cuando un equipo trabaja en conjunto para contraponer los aportes del otro a fin de producir algo más grande.

Las dimensiones básicas de trabajo en la colaboración son las siguientes:

- **Concienciación**—Las personas que trabajan juntas deben estar al tanto del trabajo de los demás.
- **Articulación**—Los colaboradores deben repartir el trabajo en unidades, dividir las unidades entre los miembros del equipo, y después que el trabajo esté hecho, reintegrarlo.
- **Apropiación**—La adaptación de tecnología a propia situación individual; la tecnología puede utilizarse de una manera completamente diferente de lo esperado por los diseñadores.

## 2.5.1 Beneficios de la colaboración en los proyectos Scrum

El Manifiesto Ágil (Fowler y Highsmith, 2001) hace hincapié en “la colaboración del cliente en lugar de la negociación del contrato”. Por lo tanto, el marco de Scrum adopta un enfoque en el que los miembros del equipo principal de Scrum (el propietario del producto, el Scrum Master y el equipo Scrum) colaboran entre sí y con los socios para crear los entregables que proporcionan el mayor valor posible para el cliente. Esta colaboración se produce durante todo el proyecto.

La colaboración asegura que los siguientes beneficios del proyecto se realicen:

1. La necesidad de cambios debido a requisitos poco clarificados se reduce al mínimo. Por ejemplo, durante los procesos de creación de la visión del proyecto, el desarrollo de épica(s), y la creación de la lista priorizada de pendientes del producto, el propietario del producto colabora con los socios para crear la visión del proyecto, la épica(s) y la lista priorizada de pendientes del producto, respectivamente. Esto asegurará que haya claridad entre los miembros principales del equipo Scrum sobre el trabajo que se requiere para completar el proyecto. El equipo Scrum colabora continuamente con el propietario del producto y los socios a través de una lista priorizada de pendientes del producto transparente para crear los entregables del proyecto. Los procesos de realización de la reunión diaria, el mantenimiento de la lista priorizada de pendientes del producto, y la retrospectiva del sprint dan margen a los miembros del equipo principal de Scrum para discutir lo que se ha hecho y colaborar en lo que hay que hacer. De esta manera se minimiza el número de solicitudes de cambio por parte del cliente.
2. Los riesgos se identifican y se tratan de manera eficiente. Por ejemplo, los riesgos del proyecto se identifican y evalúan en los procesos del desarrollo de épica(s), creación de entregables, y la realización de la reunión diaria por parte de los miembros del equipo principal de Scrum. Las herramientas de la reunión de revisión de Scrum como la reunión diaria, la planificación de la reunión del sprint, la reunión de revisión de la lista priorizada de pendientes del producto, proporcionan oportunidades para el equipo, no sólo para identificar y evaluar los riesgos, sino también para implementar respuestas a los riesgos que se identifican como riesgos de alta prioridad.
3. Se realiza el verdadero potencial del equipo. Por ejemplo, el proceso de realización de la reunión diaria le ofrece un margen al equipo Scrum para colaborar y comprender las fortalezas y debilidades de sus miembros. Si un miembro del equipo excedió el plazo para completar una tarea, los miembros del equipo Scrum se alinean en colaboración para completarla y cumplir con los objetivos acordados para completar el sprint.
4. Se garantiza la mejora continua a través de las lecciones aprendidas. Por ejemplo, el equipo Scrum utiliza el proceso de retrospectiva del sprint para identificar lo que no salió bien y lo que salió bien en el sprint anterior. Esto proporciona una oportunidad para que el Scrum Master trabaje con el equipo y así estar más preparado para el próximo sprint. Esto también asegurará que la colaboración sea aún más eficaz en el próximo Sprint.

La Figura 2-6 ilustra los beneficios de la colaboración en los proyectos Scrum.

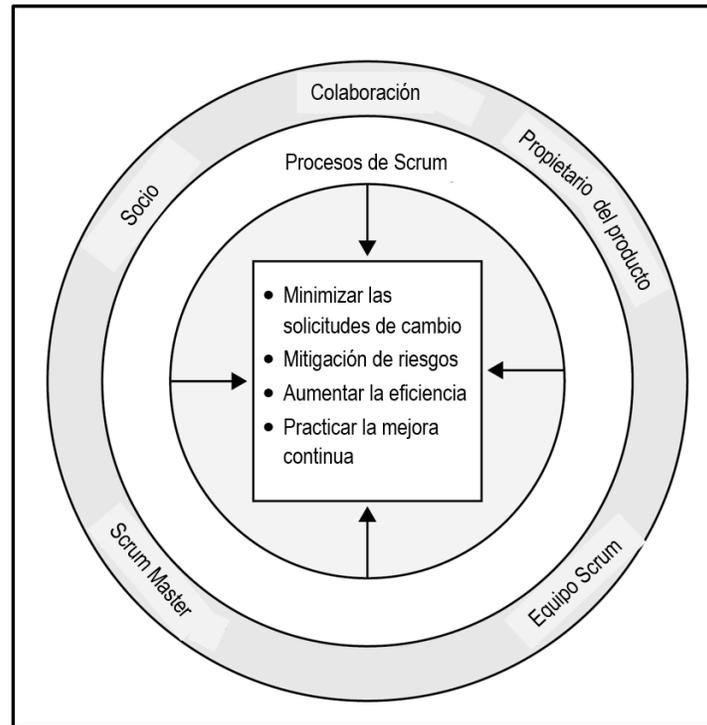


Figura 2-6: Beneficios de la colaboración en proyectos Scrum

## 2.5.2 La importancia de la co-ubicación en la colaboración

La comunicación de banda ancha se requiere para muchas de las prácticas de Scrum,. Por tanto, es preferible que los miembros del equipo estén co-ubicados. La co-ubicación (del inglés *colocation*—es decir, cuando todos los miembros del equipo del proyecto se encuentran trabajando en el mismo lugar), permite la interacción formal e informal entre los miembros del equipo. Esto proporciona la ventaja de contar con los miembros del equipo siempre a la mano para facilitar la coordinación, la resolución de problemas y el aprendizaje. Algunos de los beneficios de la co-ubicación son los siguientes:

- Las preguntas se contestan rápidamente.
- Los problemas se solucionan en ese momento.
- Existe menor fricción entre las interacciones.
- La confianza se gana con mucha más rapidez.

Las herramientas de colaboración que pueden utilizarse para los equipos co-ubicados o distribuidos geográficamente son las siguientes:

1. **Equipos co-ubicados** (es decir, los equipos que trabajan en la misma oficina)—En Scrum, es preferible tener equipos co-ubicados. Si los equipos están co-ubicados, los modos de comunicación preferidos incluyen interacciones cara-a-cara, salas de decisión o cuartos de planeación, tablero Scrum, información visual colocada en la pared, mesas compartidas, y otros.
2. **Equipos distribuidos geográficamente** (es decir, equipos que trabajan en diferentes ubicaciones físicas)—Aunque se prefiere que los equipos trabajen en un mismo lugar, a veces el equipo Scrum puede estar disperso debido a la subcontratación, la deslocalización, las diferentes ubicaciones físicas, las opciones de trabajo desde casa, y otros. Algunas herramientas que podrían utilizarse para tener una colaboración eficaz entre los equipos distribuidos geográficamente incluyen la videoconferencia, mensajes instantáneos, chats, redes sociales, pantallas compartidas y herramientas de software que simulan la funcionalidad de los tableros Scrum, la información visual colocada en la pared, y otros.

## 2.6 Priorización basada en valor

El marco de Scrum es impulsado por el objetivo de ofrecer el máximo valor empresarial en un período de tiempo mínimo. Una de las herramientas más eficaces para entregar el mayor valor en el menor tiempo posible es la priorización.

La priorización se puede definir como la determinación del orden y la separación de lo que debe hacerse ahora, de lo que debe hacerse después. El concepto de priorización no es nuevo para la gestión de proyectos. El modelo tradicional de gestión de proyectos llamado Cascada (del inglés Waterfall), propone el uso de múltiples herramientas de priorización. Desde el punto de vista del gestor del proyecto, la priorización es integral porque ciertas tareas deben llevarse a cabo primero para acelerar el proceso de desarrollo y el cumplimiento de los objetivos del proyecto. Algunas de las técnicas tradicionales de la priorización de tareas incluyen el establecimiento de plazos para las tareas delegadas y la utilización de matrices de priorización.

Sin embargo, Scrum utiliza la priorización basada en valor como uno de los principios básicos que impulsa la estructura y funcionalidad de todo el marco Scrum—ayuda a que los proyectos se beneficien a través de la capacidad de adaptación y desarrollo iterativo del producto o servicio. Y lo que es más importante, Scrum tiene como finalidad entregar un producto o servicio valioso para el cliente de forma oportuna y continua.

La priorización se lleva a cabo por el propietario del producto cuando este prioriza las historias de usuario en la lista priorizada de pendientes del producto. Esta contiene una lista de todos los requisitos necesarios para llevar el proyecto a buen término.

Una vez que el propietario del producto ha recibido los requerimientos del negocio del cliente y los ha escrito en forma de historias de usuario viables, este trabaja con el cliente y patrocinador para entender los requerimientos del negocio que proporcionan el máximo valor empresarial. El propietario del producto debe entender lo que el cliente quiere y valora a fin de organizar los elementos de la lista priorizada de pendientes del producto (historias de usuario), según su importancia relativa.

A veces, un cliente puede ordenar que todas las historias de usuario sean de alta prioridad. Aunque este pudiera ser el caso, incluso una lista de alta prioridad de historias de usuario también debe ser priorizada dentro de la lista misma. El otorgarle prioridad a una lista de pendientes implica determinar la importancia de cada historia de usuario. Los requisitos de alto valor se identifican y trasladan a la parte superior de la lista priorizada de pendientes del producto. Los procesos por los cuales el principio de la priorización basada en valor se pone en práctica son la creación de la lista priorizada de pendientes del producto y el mantenimiento de la lista priorizada de pendientes del producto.

Al mismo tiempo, el propietario del producto debe trabajar con el equipo Scrum para entender los riesgos y la incertidumbre del proyecto, ya que estos pueden tener consecuencias negativas. Estos riesgos se deben tener en cuenta al priorizar las historias de usuario con enfoque basado en el valor (consultar el capítulo de riesgos para leer más información al respecto). El equipo Scrum también alerta al propietario del producto sobre las dependencias que surgen de la implementación. Estas dependencias deben tenerse en cuenta durante la priorización. Esta puede basarse en una estimación subjetiva del valor proyectado del negocio o la rentabilidad, o puede basarse en los resultados y análisis del mercado utilizando herramientas, incluyendo, pero sin limitarse a, entrevistas del cliente, encuestas, y modelos financieros y técnicas analíticas.

El propietario del producto tiene que interpretar las entradas y las necesidades de los proyectos de los socios para crear la lista priorizada de pendientes del producto. Por lo tanto, mientras se priorizan las historias de usuario en la lista priorizada de pendientes del producto, se consideran los siguientes tres factores (véase la Figura 2-7):

1. Valor
2. Riesgo o incertidumbre
3. Dependencias

De esta forma, la priorización resulta en entregables que satisfacen los requisitos del cliente con el objetivo de ofrecer el máximo valor de negocio en el menor tiempo posible.

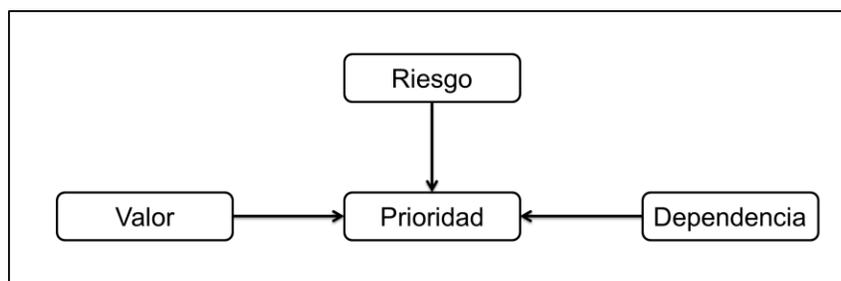


Figura 2-7: Priorización basada en valor

## 2.7 Bloque de tiempo asignado

Scrum trata al tiempo como uno de los limitantes más importantes en la gestión de un proyecto. Para hacer frente a la restricción del tiempo, Scrum introduce un concepto de asignación de un bloque de tiempo (del inglés *Time-Boxing*), que propone la fijación de una cierta cantidad de tiempo para cada proceso y actividad en un proyecto Scrum. Esto garantiza que los miembros del Equipo Scrum no ocupen demasiado o muy poco tiempo para un trabajo determinado, y que no desperdicien su tiempo y energía en un trabajo para el cual tienen poca claridad.

Algunas de las ventajas de la asignación del bloque de tiempo son las siguientes:

- Proceso de desarrollo eficiente
- Menos gastos generales
- Alta velocidad para los equipos

La asignación del bloque de tiempo puede utilizarse en muchos procesos de Scrum, por ejemplo, en el proceso de realización de la reunión diaria, la duración de la misma se asigna a un bloque de tiempo. A veces, la asignación del bloque de tiempo puede utilizarse para evitar la mejora excesiva de un elemento (por ejemplo, *gold-plating*, un término en inglés que indica la incorporación de características o refinamientos costosos e innecesarios en un producto o servicio).

La asignación del bloque de tiempo es una práctica muy importante en Scrum, y esta debe aplicarse con cuidado. Una asignación de un bloque de tiempo arbitraria puede llevar a la desmotivación del equipo y tener como consecuencia la creación de un ambiente tenso, por lo que esta se debe utilizar de manera apropiada.

### 2.7.1 Bloques de tiempo de Scrum

- **Sprint**—Un sprint es una iteración que se asigna a un bloque de tiempo de una a seis semanas de duración durante el cual el Scrum Master guía, facilita y protege al equipo Scrum de impedimentos tanto internos como externos durante el proceso de creación de entregables. Esto ayuda a evitar una expansión de la visión más allá de su objetivo original, lo que podría afectar la meta del sprint. Durante este tiempo, el equipo trabaja para convertir las necesidades de la lista priorizada de pendientes del producto en funcionalidades de productos fáciles de enviar. Para obtener los máximos beneficios de un proyecto Scrum, siempre se recomienda mantener el sprint dentro de un bloque de tiempo de 4 semanas, a menos que existan proyectos con requisitos muy estables, en los que los sprints pueden extenderse hasta seis semanas.
- **Reunión diaria**—Esto se refiere a una reunión diaria de corta duración, que se asigna a un bloque de tiempo de 15 minutos. En esta reunión rápida, los miembros del equipo se reúnen para informar sobre cómo marcha el proyecto, respondiendo a las siguientes tres preguntas:

1. ¿Qué terminé ayer?
2. ¿Qué terminaré hoy?
3. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

Esta reunión se lleva a cabo por el equipo como parte del proceso de la realización de la reunión diaria.

- **Reunión de planificación del sprint**—Esta reunión se lleva a cabo antes del sprint, como parte del proceso de creación de la lista de pendientes del sprint. Se asigna a un bloque de tiempo de ocho horas durante un sprint de un mes de duración. La reunión de planificación del sprint se divide en dos partes:
  1. Definición del objetivo—Durante la primera mitad de la reunión, el propietario del producto explica la máxima prioridad de las historias de usuario o requisitos en la lista priorizada de pendientes del producto para el equipo Scrum. Después, el equipo define el objetivo del sprint en colaboración con el propietario del producto.
  2. Estimación de tareas—Durante la segunda mitad de la reunión, el equipo Scrum decide como completar la lista priorizada de pendientes del producto seleccionados para cumplir con la meta del Sprint.

A veces, a las reuniones de planificación de tareas (efectuadas durante el proceso de creación de tareas) y a las reuniones de estimación de tareas, (efectuadas durante el proceso de estimación de tareas), también se les conoce como reuniones de planificación del sprint.

- **Reunión de revisión del sprint**—La reunión de revisión del sprint se asigna a un bloque de tiempo de cuatro horas en un sprint de un mes. Durante la reunión de revisión del sprint que se efectúa en el proceso de demostración y validación del sprint, el equipo Scrum le presenta los entregables del sprint actual al propietario del producto. Este revisa el producto (o incremento del producto) para compararlo con los criterios de aceptación acordados y luego acepta o rechaza las historias de usuario completadas.
- **Reunión de retrospectiva del sprint**—Esta reunión se asigna a un bloque de tiempo de 4 horas para un sprint de un mes, y se lleva a cabo como parte del proceso retrospectiva del sprint. Durante esta reunión, el equipo Scrum revisa y reflexiona sobre el sprint anterior en relación a los procesos que se siguieron, las herramientas empleadas, la colaboración, y los mecanismos de comunicación, al igual que otros aspectos de interés para el proyecto. El equipo discute lo que salió bien durante el sprint anterior y lo que no salió bien, con el objetivo de aprender y mejorar sprints futuros. Algunas oportunidades de mejora o las mejores prácticas de esta reunión también podrían actualizarse como parte de los documentos del cuerpo de asesoramiento de Scrum.

La Figura 2-8 ilustra las duraciones de los bloques de tiempo para las reuniones relacionadas con Scrum.

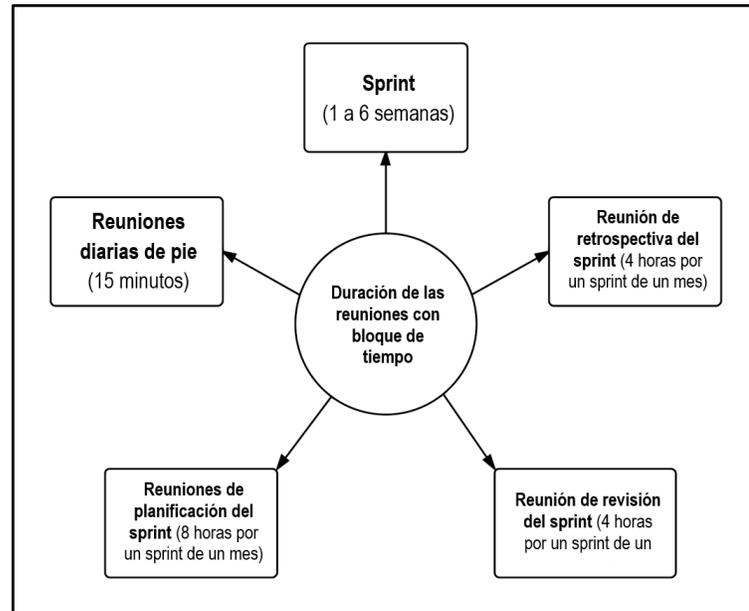


Figura 2-8: Duración de los bloques de tiempo para las reuniones de Scrum

## 2.8 Desarrollo iterativo

El marco de Scrum es impulsado por el objetivo de ofrecer el máximo valor empresarial en un período de tiempo mínimo. Para lograr esto de forma práctica, Scrum cree en el desarrollo iterativo de entregables.

En la mayoría de los proyectos complejos, el cliente puede que no sea capaz de definir unos requisitos muy concretos o puede no estar seguro de cómo debería de ser el producto final. El modelo iterativo es más flexible para asegurar que cualquier cambio que solicite el cliente se pueda incluir como parte del proyecto. Es posible que las historias de usuario tengan que ser escritas constantemente a lo largo de la duración del proyecto. En las etapas iniciales del proceso de escribirlas, la mayoría de las historias son las funcionalidades de alto nivel. Estas historias de usuario se conocen como épica(s). Las épicas, por lo general, son muy grandes para que los equipos las completen en un sólo sprint, y por lo tanto se dividen en pequeñas historias de usuario.

Cada aspecto complejo del proyecto se divide mediante la elaboración progresiva durante el proceso del mantenimiento de la lista priorizada de pendientes del producto. Los procesos de creación de historias de usuario y de estimación, aprobación y asignación de las historias de usuario se utilizan para agregar nuevos requisitos a la lista priorizada de pendientes del producto. La tarea del propietario del producto es asegurar un mayor retorno sobre la inversión (RSI), centrándose en el valor y en la entrega continua con cada sprint. El propietario del producto debería tener una buena comprensión de la justificación del negocio y el valor que el proyecto se supone debe entregar al redactar la lista priorizada de pendientes del producto, y por lo

tanto decidir qué entregables contractuales y valores se han de entregar en cada sprint. Posteriormente, los procesos de creación de tareas, estimación de tareas, y creación de la lista de pendientes del Sprint producen la lista de pendientes del sprint, la cual el equipo utiliza para crear los entregables.

En cada sprint, el proceso de creación de entregables se utiliza para desarrollar las salidas del sprint. El Scrum Master tiene que garantizar que se sigan los procesos de Scrum, y facilitar al equipo el trabajo de la manera más productiva. El equipo Scrum se auto-organiza, teniendo como objetivo el crear entregables del sprint a partir de las historias de usuario que están en la lista priorizada de pendientes del sprint. En grandes proyectos, varios equipos interfuncionales trabajan en paralelo a través de los sprints, proporcionando soluciones potencialmente entregables al final de cada sprint. Después de que cada sprint se ha completado, el propietario del producto acepta o rechaza los entregables en base a los criterios de aceptación del proceso de demostración y validación del sprint.

Como se ilustra en la Figura 2-9, los proyectos Scrum se completan de manera iterativa, entregando valor a lo largo del ciclo de vida del proyecto.

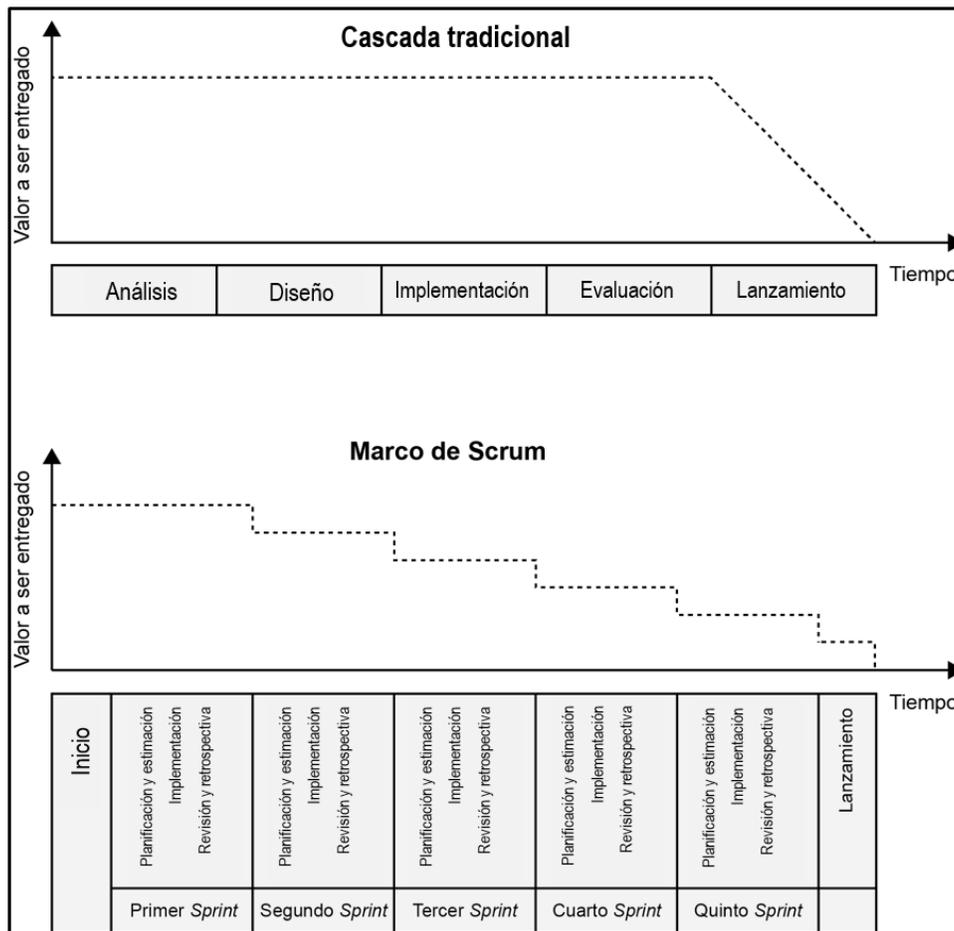


Figura 2-9: Scrum vs Cascada tradicional

El beneficio del desarrollo iterativo es que permite la corrección a medida que todas las personas involucradas obtengan una mejor comprensión de lo que se debe entregar como parte del proyecto, e incorporar lo aprendido de manera iterativa. Así, el tiempo y el esfuerzo requerido para alcanzar el punto final definitivo, se reduce considerablemente y el equipo produce entregables que se adaptan mejor al entorno empresarial.

## 2.9 Scrum vs. Gestión tradicional de proyectos

El énfasis en la gestión de proyectos tradicional es llevar a cabo la planificación detallada del proyecto por adelantado con el énfasis en gestionar y solucionar el alcance, costo, horarios y gestionar esos parámetros. A veces, la gestión tradicional de proyectos puede llevar a una situación en la que, aunque el plan se ha logrado, el cliente no está satisfecho.

El marco de Scrum se basa en la creencia de que los trabajadores de hoy en día pueden ofrecer mucho más que sus conocimientos técnicos, y en que tratar de asignar y planear en un ambiente de constante cambio no es eficiente. Por lo tanto, Scrum alienta a la toma de decisiones iterativa en base a datos. En Scrum, el enfoque principal es la entrega de productos que satisfagan los requisitos del cliente en pequeños incrementos iterativos que sean entregables.

Para entregar la mayor cantidad de valor en el menor tiempo posible, Scrum promueve la priorización y la asignación de bloques de tiempo en vez de la fijación del alcance, el costo y el cronograma de un proyecto. Una característica importante de Scrum es la auto-organización, la cual le permite a los empleados que estimen y tomen responsabilidad por las tareas.

## 3. ORGANIZACIÓN

### 3.1 Introducción

En esta sección veremos las diversas facetas de la organización de un proyecto Scrum, así como los roles centrales y los roles no centrales, y cómo formar equipos Scrum de alto rendimiento.

La organización, tal como se define en *Una guía para el conocimiento de Scrum (Guía SBOK™)*, se aplica a lo siguiente:

- Carteras, programas y/o proyectos de cualquier sector
- Productos, servicios o cualesquier otros resultados que se entregarán a los socios
- Proyectos de cualquier tamaño y complejidad

En la *Guía SBOK™*, el término "producto" puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria—desde pequeños proyectos o equipos con tan sólo seis miembros por equipo, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo está dividido en las siguientes secciones:

**3.2 Guía de los roles**—Esta sección identifica qué sección o subsección es importante para un propietario del producto, un Scrum Master y un equipo Scrum.

**3.3 Roles de un proyecto Scrum**—Esta sección cubre los roles centrales y no centrales claves con un proyecto Scrum.

**3.4 Propietario del producto**—En esta sección se destacan las principales responsabilidades del propietario del producto en relación con un proyecto Scrum.

**3.5 Scrum Master**—Esta sección se centra en las principales responsabilidades del Scrum Master en el contexto de un proyecto Scrum.

**3.6 Equipo Scrum**—Esta sección hace hincapié en las responsabilidades claves del equipo Scrum en el contexto de un proyecto Scrum.

**3.7 Scrum en proyectos, programas y carteras**—Esta sección se centra en cómo un marco de Scrum puede adaptarse y utilizarse en los diferentes contextos de los programas y las carteras. También se destacan las responsabilidades específicas de los miembros del equipo Scrum en relación con la comunicación, la integración y el trabajo con los equipos empresariales y de gestión de programas.

**3.8 Responsabilidades**—En esta sección se describen las responsabilidades pertinentes al tema de la organización, para todos quienes trabajan en un proyecto, en función de sus roles.

**3.9 Scrum vs. Gestión de proyecto tradicional**—Esta sección explica las principales diferencias y ventajas del modelo Scrum en relación con el modelo en Cascada tradicional de gestión de proyectos.

**3.10 Teorías de recursos humanos populares y su relevancia para Scrum**—Esta sección contiene algunas de las teorías de recursos humanos más populares de utilidad para todos los miembros del equipo principal de Scrum.

## 3.2 Guía de los roles

3

1. Propietario del producto—Es imperativo que el propietario del producto lea todo el capítulo.
2. Scrum Master—El Scrum Master también debe familiarizarse con todo este capítulo con enfoque principal en las secciones 3.3, 3.5, 3.6, 3.8 y 3.10.4.
3. Equipo Scrum—El equipo Scrum debe centrarse principalmente en las secciones 3.3, 3.6, y 3.8.

## 3.3 Roles de un proyecto Scrum

El entendimiento de los roles y las responsabilidades definidas es muy importante para asegurar la implementación exitosa de los proyectos Scrum.

Los roles de Scrum se dividen en dos categorías:

1. **Roles centrales**—Los roles centrales son aquellos que obligatoriamente se requieren para crear el producto del proyecto, están comprometidos con el proyecto, y por último son los responsables del éxito de cada sprint del proyecto y del proyecto en su totalidad.
2. **Roles no centrales**—Los roles no centrales son aquellos que no son obligatoriamente necesarios para el proyecto Scrum, y pueden incluir miembros de los equipos que tengan interés en el proyecto, pero que no tienen ninguna función formal en el equipo del proyecto. Ellos pueden interactuar con el equipo, pero no son responsables del éxito del proyecto. Los roles no centrales también deben tenerse en cuenta en cualquier proyecto de Scrum.

### 3.3.1 Roles centrales

Hay tres roles centrales en Scrum que en última instancia llevan la responsabilidad de cumplir con los objetivos del proyecto. Los roles centrales son el propietario del producto, el Scrum Master y el equipo Scrum. En conjunto se les conoce como el equipo principal de Scrum. Es importante tener en cuenta que, de estos tres roles, ningún rol tiene autoridad sobre los otros.

## 1. Propietario del producto

El propietario del producto es la persona responsable de maximizar el valor del negocio para el proyecto. Este rol es responsable de articular los requisitos del cliente y de mantener la justificación del negocio del proyecto. El propietario del producto representa la voz del cliente.

De manera similar al rol del propietario del producto en un proyecto, pudiera haber un propietario del producto del programa o un propietario del producto de la cartera, para un programa y una cartera, respectivamente.

## 2. Scrum Master

El Scrum Master es un facilitador que asegura que el equipo Scrum esté dotado de un ambiente propicio para completar con éxito el desarrollo del producto. El Scrum Master guía, facilita e imparte prácticas de Scrum a todos los participantes en el proyecto, elimina los impedimentos que enfrenta el equipo, y asegura que se estén siguiendo los procesos de Scrum.

Debe tenerse en cuenta que el rol de Scrum Master es muy diferente a la función que desempeña el director de un proyecto en un modelo de Cascada tradicional de gestión de proyectos, en el que el director del proyecto trabaja como gerente o líder del proyecto. El Scrum Master sólo funge como un facilitador y está en el mismo nivel jerárquico que cualquier otra persona en el equipo Scrum—cualquier persona del equipo Scrum que aprenda a facilitar proyectos Scrum puede convertirse en el Scrum Master de un proyecto o sprint.

De manera similar al rol de Scrum Master en un proyecto, también pudiera haber un Scrum Master del programa o un Scrum Master de la cartera, para un programa y una cartera, respectivamente.

## 3. Equipo Scrum

El equipo Scrum es un grupo o equipo de personas que son responsables de la comprensión de los requerimientos del negocio que se especifican por el propietario del producto, de la estimación de las historias de usuarios y de la creación final de los entregables del proyecto.

La Figura 3-1 presenta un resumen de los roles centrales del equipo Scrum.

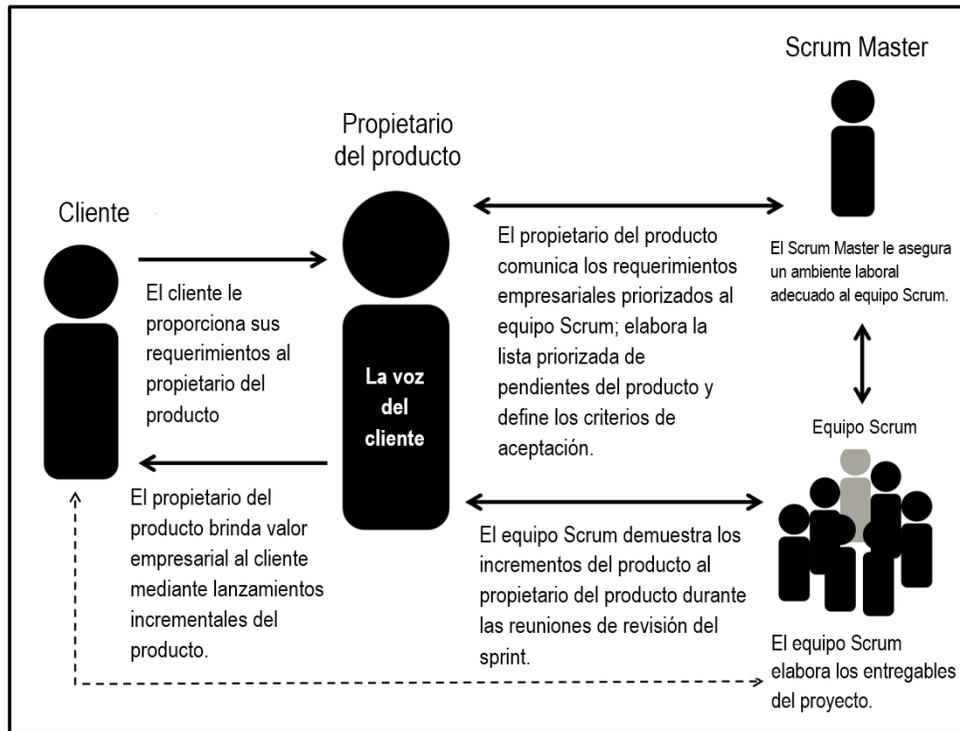


Figura 3-1: Roles de Scrum—Descripción General

### 3.3.2 Roles no centrales

Los roles no centrales son aquellos roles que no son obligatoriamente necesarios para el proyecto Scrum y pueden no participar en el proceso de Scrum. Sin embargo, es importante tener conocimiento sobre estos roles no centrales, ya que podrían desempeñar un rol importante en algunos proyectos de Scrum.

Los roles no centrales pueden incluir los siguientes:

#### 1. Socio(s)

El(los) socio(s) es un término colectivo que incluye a los clientes, usuarios y patrocinadores, que a menudo interactúan con el propietario del producto, el Scrum Master y el equipo Scrum para proporcionarles las entradas y facilitar la creación del producto del proyecto, servicio, o cualquier otro resultado. El(los) socio(s) influyen en el proyecto a lo largo del desarrollo del mismo. Los socios también pueden desempeñar un rol en los procesos importantes de Scrum tales como el desarrollo de épica(s), la creación de la lista priorizada de pendientes del producto, la realización del plan de lanzamiento y la retrospectiva del sprint.

- **Ciente**

El cliente es la persona o la organización que adquiere el producto, servicio o cualquier otro resultado del proyecto. Para cualquier organización, dependiendo del proyecto, puede haber tanto clientes internos (es decir, dentro de la misma organización), como clientes externos (es decir, fuera de la organización).

- **Usuarios**

El usuario es el individuo o la organización que utiliza directamente el producto, servicio o cualquier otro resultado del proyecto. Al igual que los clientes, para cualquier organización, puede haber tanto usuarios internos ni externos. En algunas industrias los clientes y los usuarios también pueden ser los mismos.

- **Patrocinador**

El patrocinador es la persona o la organización que provee recursos y apoyo para el proyecto. El patrocinador es también el socio a quien todos le deben rendir cuentas al final.

A veces, la misma persona u organización puede desempeñar múltiples roles de socios—por ejemplo, el promotor y el cliente pueden ser el mismo.

## 2. Vendedores

Los vendedores incluyen a individuos u organizaciones externas que ofrecen productos y servicios que no están dentro de las competencias básicas de la organización del proyecto.

## 3. Cuerpo de asesoramiento de Scrum

El cuerpo de asesoramiento de Scrum es una función opcional. Por lo general, se compone de un grupo de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de los objetivos relacionados con la calidad, las regulaciones gubernamentales, la seguridad y otros parámetros clave de la organización. Estos objetivos guían la labor llevada a cabo por el propietario del producto, el Scrum Master y el equipo Scrum. El cuerpo de asesoramiento de Scrum también ayuda a captar las mejores prácticas que deben utilizarse en todos los proyectos de Scrum en la organización.

El cuerpo de asesoramiento de Scrum no toma decisiones relacionadas con el proyecto. En cambio, actúa como una estructura de consultoría u orientación para todos los niveles de la jerarquía en el proyecto de organización de la cartera, programa y proyecto. Los equipos Scrum tienen la opción de solicitar ayuda el cuerpo de asesoramiento de Scrum sobre cualquier recomendación que requieran.

### 3.4 Propietario del producto

El propietario del producto representa los intereses de la comunidad de socios para el equipo Scrum. Este role es responsable de garantizar una comunicación clara sobre el producto y los requisitos de funcionalidad del servicio con el equipo Scrum, al igual que de definir los criterios de aceptación, y asegurar que se cumplan dichos criterios. En otras palabras, el propietario del producto es responsable de asegurar que el equipo Scrum ofrezca valor. Este rol central siempre debe mantener una visión dual. También debe entender y apoyar las necesidades e intereses de todos los socios, al tiempo que comprende las necesidades y el funcionamiento del equipo Scrum. Puesto que el Propietario del producto debe entender las necesidades y prioridades de los socios, incluyendo los clientes y los usuarios, este papel se conoce comúnmente como la voz del cliente.

La Tabla 3-1 resume las responsabilidades del propietario del producto en los diferentes procesos de Scrum.

Proceso	Responsabilidades del propietario del producto
8.1 Creación de la visión del proyecto	<ul style="list-style-type: none"> <li>Definir la visión del proyecto</li> <li>Ayudar a crear el acta constitutiva del proyecto y el presupuesto del proyecto</li> </ul>
8.2 Identificación del Scrum Master y el(los) socio(s)	<ul style="list-style-type: none"> <li>Ayudar a finalizar la elección del Scrum Master para el proyecto</li> <li>Identificar al(los) socio(s)</li> </ul>
8.3 Formación del equipo Scrum	<ul style="list-style-type: none"> <li>Ayudar a determinar a los miembros del equipo Scrum</li> <li>Ayudar a desarrollar un plan de colaboración</li> <li>Ayudar a desarrollar el plan del equipo con el(los) Scrum Master(s)</li> </ul>
8.4 Desarrollo de épica(s)	<ul style="list-style-type: none"> <li>Crear épica(s) y prototipos</li> </ul>
8.5 Creación de la lista priorizada de pendientes del producto	<ul style="list-style-type: none"> <li>Priorizar los elementos de la lista priorizada de pendientes del producto</li> <li>Definir los criterios de terminado</li> </ul>
8.6 Realización del plan del lanzamiento	<ul style="list-style-type: none"> <li>Crear el cronograma de planificación del lanzamiento</li> <li>Ayudar a determinar la duración del sprint</li> </ul>
9.1 Creación de historias de usuario	<ul style="list-style-type: none"> <li>Ayudar a crear historias de usuario</li> <li>Definir los criterios de aceptación para cada historia de usuario</li> </ul>
9.2 Aprobación, estimación y asignación de las historias de usuario	<ul style="list-style-type: none"> <li>Aprobar las historias de usuario</li> <li>Facilitar al equipo Scrum y asignar historias de usuario</li> </ul>
9.3 Creación de tareas	<ul style="list-style-type: none"> <li>Explicar las historias de usuario al equipo Scrum, al tiempo que crea la lista de tareas</li> </ul>
9.4 Estimación de tareas	<ul style="list-style-type: none"> <li>Proporcionar orientación y claridad al equipo Scrum sobre la estimación del esfuerzo para las tareas</li> </ul>

9.5 Elaboración de la lista de pendientes del Sprint	<ul style="list-style-type: none"> <li>• Clarificar los requisitos al equipo Scrum al tiempo que crea la lista de pendientes del sprint</li> </ul>
10.1 Creación de entregables	<ul style="list-style-type: none"> <li>• Clarificar los requerimientos al equipo Scrum</li> </ul>
10.3 Mantenimiento de la lista priorizada de pendientes del producto	<ul style="list-style-type: none"> <li>• Mantener la lista priorizada de pendientes del producto</li> </ul>
11.2 Demostración y validación del sprint	<ul style="list-style-type: none"> <li>• Aceptar/rechazar los entregables</li> <li>• Proporcionar la retroalimentación necesaria para el Scrum Master y los equipos Scrum</li> <li>• Actualizar el plan de lanzamiento y la lista priorizada de pendientes del producto</li> </ul>
12.1 Envío de entregables	<ul style="list-style-type: none"> <li>• Ayudar con el lanzamiento del producto y coordinar esto con el cliente</li> </ul>
12.2 Retrospectiva del proyecto	<ul style="list-style-type: none"> <li>• Participar en reuniones de retrospectiva del sprint</li> </ul>

Tabla 3-1: Responsabilidades del propietario del producto en los procesos de Scrum

Las demás responsabilidades de un propietario del producto incluyen:

- Determinar los requisitos generales iniciales del proyecto y dar inicio a las actividades del mismo; esto puede implicar una interacción con el propietario del producto del programa y el propietario del producto de la cartera, a fin de asegurar que el proyecto se alinee con la dirección que proporciona la alta gerencia.
- Representar al(los) usuario(s) del producto o servicio con un profundo conocimiento de la comunidad de usuarios.
- Asegurar los recursos financieros del proyecto.
- Centrarse en la creación de valor y el retorno sobre la inversión en general.
- Evaluar la viabilidad y garantizar la entrega del producto o servicio.

### 3.4.1 Voz del cliente

Como representante del cliente, se dice que el propietario del producto es la voz del cliente, ya que es quien asegura que las necesidades explícitas e implícitas del cliente se reflejen en las historias de usuario en la lista priorizada de pendientes del producto, y que más adelante se utilicen para crear los entregables del proyecto para el cliente.

### 3.4.2 Jefe propietario del producto

En los grandes proyectos, con numerosos equipos Scrum, el contar con un jefe propietario del producto puede ser algo necesario. Este rol se encarga de coordinar el trabajo de múltiples propietarios del producto. Es el jefe propietario del producto quien prepara y mantiene la lista priorizada de pendientes del producto en su totalidad para los proyectos grandes, usándolo para así coordinar el trabajo a través de los propietarios

del producto de los equipos Scrum. Estos, a su vez, se encargan de administrar sus respectivas partes en la lista priorizada de pendientes del producto.

El jefe propietario del producto también se comunica con el propietario del producto del programa para asegurar la alineación del proyecto con las metas y objetivos del programa.

### 3.5 Scrum Master

El Scrum Master es el líder servicial del equipo Scrum, y es quien modera y facilita las interacciones del equipo como entrenador y motivador del mismo. Este rol es responsable de asegurarse que el equipo tenga un ambiente de trabajo productivo mediante al protegerlo de influencias externas, eliminando todos los obstáculos, y confirmando que se cumplan los principios, aspectos y procesos de Scrum.

La Tabla 3-2 resume las responsabilidades del Scrum Master en los diferentes procesos de Scrum.

Procesos	Responsabilidades del Scrum Master
8.2 Identificación del Scrum Master y el(los) socio(s)	<ul style="list-style-type: none"> <li>• Ayudar a identificar al(los) socio(s) para el proyecto</li> </ul>
8.3 Formación del equipo Scrum	<ul style="list-style-type: none"> <li>• Facilitar la selección del equipo Scrum</li> <li>• Facilitar la creación del plan de colaboración y el plan de desarrollo del equipo</li> <li>• Garantizar que los recursos de respaldo estén disponibles para el funcionamiento del proyecto sin problemas</li> </ul>
8.4 Desarrollo de épica(s)	<ul style="list-style-type: none"> <li>• Facilitar la creación de épica(s) y prototipos</li> </ul>
8.5 Creación de la lista priorizada de pendientes del producto	<ul style="list-style-type: none"> <li>• Ayudar al propietario del producto en la creación de la lista priorizada de pendientes del producto y en la definición de los criterios de terminado</li> </ul>
8.6 Realización del plan del lanzamiento	<ul style="list-style-type: none"> <li>• Coordinar la creación del cronograma de planificación del lanzamiento</li> <li>• Determinar de la duración del sprint</li> </ul>
9.1 Creación de las historias de usuario	<ul style="list-style-type: none"> <li>• Asistir al equipo Scrum en la creación de historias de usuario y sus criterios de aceptación</li> </ul>
9.2 Aprobación, estimación y asignación de las historias de usuario	<ul style="list-style-type: none"> <li>• Facilitar las reuniones del equipo Scrum para estimar y crear las historias de usuario</li> </ul>
9.3 Creación de tareas	<ul style="list-style-type: none"> <li>• Facilitar al equipo Scrum en la creación de la lista de tareas para el próximo sprint</li> </ul>
9.4 Estimación de tareas	<ul style="list-style-type: none"> <li>• Asistir al equipo Scrum en estimar el esfuerzo necesario para completar las tareas acordadas para el sprint</li> </ul>
9.5 Creación de la lista de pendientes del sprint	<ul style="list-style-type: none"> <li>• Asistir al equipo Scrum en el desarrollo de la lista de pendientes del sprint y la tabla del trabajo pendiente del sprint</li> </ul>

10.1 Creación de entregables	<ul style="list-style-type: none"> <li>• Apoyar al equipo Scrum en la creación de los entregables acordados para el sprint</li> <li>• Ayudar a actualizar el tablero de Scrum y el registro de impedimentos</li> </ul>
10.2 Realización de la reunión diaria	<ul style="list-style-type: none"> <li>• Asegurar que el tablero Scrum y el registro de impedimentos permanezcan actualizados</li> </ul>
10.3 Mantenimiento de la lista priorizada de pendientes del producto	<ul style="list-style-type: none"> <li>• Facilitar las reuniones de revisión de la lista priorizada de pendientes del producto</li> </ul>
11.1 Convocar el Scrum de Scrums	<ul style="list-style-type: none"> <li>• Asegurar que los problemas que afectan al equipo Scrum se discutan y resuelvan</li> </ul>
11.2 Demostración y validación de sprints	<ul style="list-style-type: none"> <li>• Facilitar la presentación de los entregables ya completados por el equipo Scrum para la aprobación del propietario del producto</li> </ul>
11.3 Retrospectiva de Sprint	<ul style="list-style-type: none"> <li>• Garantizar que exista un ambiente ideal para el equipo Scrum del proyecto en los sucesivos sprints</li> </ul>
12.2 Retrospectiva del proyecto	<ul style="list-style-type: none"> <li>• Representar al equipo principal de Scrum) para proporcionar lecciones del proyecto actual, de ser necesario</li> </ul>

Tabla 3-2: Responsabilidades del Scrum Master en los procesos de Scrum

### 3.5.1 Jefe Scrum Master

Los grandes proyectos requieren que varios equipos Scrum trabajen en paralelo. Es muy posible que la información que obtiene un equipo se tenga que comunicar adecuadamente a otros equipos. El jefe Scrum Master es el responsable de esta actividad.

La coordinación entre los distintos equipos Scrum que trabajan en un proyecto Scrum se realiza normalmente a través de la reunión de Scrum de Scrums (véase la sección 3.7.2.1). Esta es análoga a la reunión diaria de pie y es facilitada por el jefe Scrum Master. Este rol suele ser generalmente responsable de resolver los impedimentos que afectan a más de un equipo Scrum.

La Figura 3-2 proporciona las preguntas que se hacen durante una reunión de Scrum de Scrums.

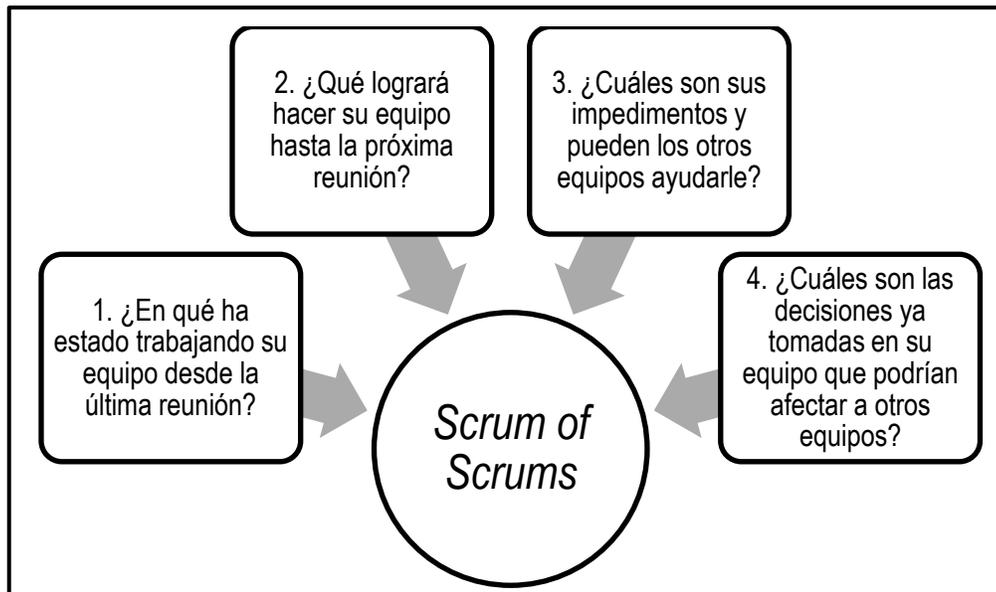


Figura 3-2: Las preguntas formuladas durante una reunión de Scrum de Scrums

Por lo general, cualesquier problemas que se susciten entre varios equipos se discuten y resuelven por las propias partes interesadas. Esto se hace en una sesión, inmediatamente después de la reunión de Scrum de Scrums, la cual es facilitada por el jefe Scrum Master.

### 3.6 Equipo Scrum

Al equipo Scrum se le llama a veces equipo de desarrollo, ya que este es responsable del desarrollo del producto, servicio o de cualquier otro resultado. Consiste en un grupo de personas que trabajan en las historias de usuario en la lista de pendientes del sprint para crear los entregables del proyecto.

La Tabla 3-3 resume las responsabilidades del equipo Scrum en los diversos procesos de Scrum.

Procesos	Responsabilidades del equipo Scrum
8.3 Formación del equipo Scrum	<ul style="list-style-type: none"> <li>• Proporcionar entradas para la creación del plan de colaboración y del plan de desarrollo del equipo</li> </ul>
8.4 Desarrollo de épica(s)	<ul style="list-style-type: none"> <li>• Asegurar una comprensión clara de la épica(s) y prototipos</li> </ul>
8.5 Lista priorizada de pendientes del producto	<ul style="list-style-type: none"> <li>• Entender las historias de usuario en la lista priorizada de pendientes del producto</li> </ul>
8.6 Realización del plan de lanzamiento	<ul style="list-style-type: none"> <li>• Estar de acuerdo con los demás miembros del equipo principal de Scrum sobre la duración del sprint</li> </ul>

	<ul style="list-style-type: none"> <li>• Buscar clarificación sobre los nuevos productos o cambios, si los hay, en los productos existentes en la lista priorizada de pendientes del producto</li> </ul>
9.1 Creación de las historias de usuario	<ul style="list-style-type: none"> <li>• Proporcionar entradas al propietario del producto en la creación de historias de usuario</li> </ul>
9.2 Aprobación, estimación y asignación de las historias de usuario	<ul style="list-style-type: none"> <li>• Estimar las historias de los usuarios aprobadas por el propietario del producto</li> <li>• Asignar las historias de usuario que se hacen en un Sprint</li> </ul>
9.3 Creación de tareas	<ul style="list-style-type: none"> <li>• Desarrollar la lista de tareas en base a las historias de usuario ya convenidas y las dependencias</li> </ul>
9.4 Estimación de tareas	<ul style="list-style-type: none"> <li>• Calcular el esfuerzo para las tareas identificadas y, si es necesario, actualizar la lista de tareas</li> </ul>
9.5 Creación de la lista de pendientes del sprint	<ul style="list-style-type: none"> <li>• Desarrollar la lista de pendientes del sprint y la tabla del trabajo pendiente del sprint</li> </ul>
10.1 Creación de entregables	<ul style="list-style-type: none"> <li>• Crear entregables</li> <li>• Identificar riesgos y ejecutar acciones de mitigación de riesgos, si los hay</li> <li>• Actualizar el registro de impedimentos y las dependencias</li> </ul>
10.2 Realización de la reunión diaria	<ul style="list-style-type: none"> <li>• Actualizar la tabla del trabajo pendiente, el tablero Scrum, y el registro de impedimentos</li> <li>• Discutir problemas que enfrenta cada miembro y buscar soluciones para motivar al equipo</li> <li>• Identificar riesgos, si lo hay</li> <li>• Presentar solicitudes de cambio, si se requieren</li> </ul>
10.3 Mantenimiento de la lista priorizada de pendientes del producto	<ul style="list-style-type: none"> <li>• Participar en las reuniones de revisión de la lista priorizada de pendientes del producto</li> </ul>
11.1 Convocar el Scrum de Scrums	<ul style="list-style-type: none"> <li>• Proporcionar entradas al Scrum Master para las reuniones de Scrum de Scrums</li> </ul>
11.2 Demostración y validación de Sprints	<ul style="list-style-type: none"> <li>• Mostrar entregables completados al propietario del producto para su aprobación</li> </ul>
11.3 Retrospectiva del Sprint	<ul style="list-style-type: none"> <li>• Identificar oportunidades de mejora, si las hay, del Sprint actual y decir si está de acuerdo sobre las posibles mejoras viables para el próximo sprint</li> </ul>
12.2 Retrospectiva del proyecto	<ul style="list-style-type: none"> <li>• Participar en la reunión de retrospectiva del proyecto</li> </ul>

**Tabla 3-3: Responsabilidades del equipo Scrum en los procesos de Scrum**

### 3.6.1 Selección de personal

La Figura 3-3 enumera las características deseables para las funciones básicas de Scrum.

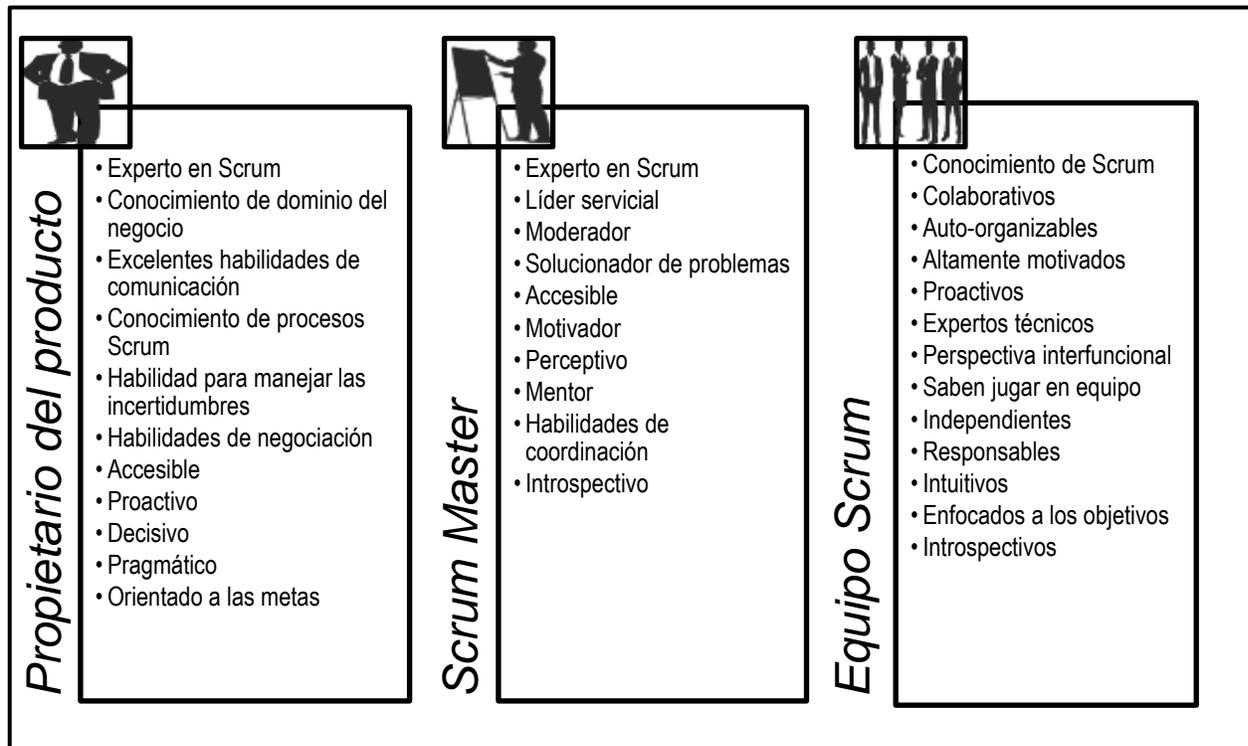


Figura 3-3: Características deseadas de los roles principales de Scrum

### 3.6.2 Tamaño del equipo Scrum

Es importante que el equipo Scrum posea todas las habilidades esenciales necesarias para llevar a cabo el trabajo del proyecto. También es necesario contar con un alto nivel de colaboración para maximizar la productividad, de modo que se requiera una mínima coordinación para llevar a cabo el trabajo.

El tamaño óptimo de un equipo Scrum es de seis a diez miembros, lo suficientemente grande para asegurar habilidades adecuadas, pero lo suficientemente pequeño como para facilitar la colaboración. Un beneficio clave de un equipo de seis a diez miembros es que la comunicación y la gestión suelen ser simples y requieren un esfuerzo mínimo. Sin embargo, también puede haber inconvenientes. Una desventaja importante es que los equipos más pequeños se ven afectados más significativamente por la pérdida de un miembro del equipo, en comparación a los equipos más grandes, aunque esta pérdida sea por un corto tiempo. Este problema se puede solucionar si los miembros del equipo tienen conocimientos especializados y habilidades fuera de su rol específico. Sin embargo, esto puede ser difícil y depende del tipo de proyecto, la industria, y el tamaño de la organización. También se recomienda tener suplentes para reemplazar a cualquier persona que pueda tener que dejar el equipo Scrum.

## 3.7 Scrum en proyectos, programas, y carteras

### 3.7.1 Definición de proyecto, programa, y cartera

- Proyecto—Un proyecto es una empresa de colaboración para crear nuevos productos o servicios, o para obtener resultados como los que se definen en la declaración de la visión del proyecto. Los proyectos son por lo general afectados por limitaciones de tiempo, costo, alcance, calidad, el personal y la capacidad de la organización. El objetivo del equipo de proyecto es crear entregables, como se define en la lista priorizada de pendientes del producto.
- Programa—Un programa es un grupo de proyectos relacionados con el objetivo de entregar resultados de negocio definidos en la declaración de la visión del programa. La lista priorizada de pendientes del programa incorpora la lista priorizada de pendientes del producto de todos los proyectos del programa.
- Cartera—Una cartera es un grupo de programas relacionados, con el objetivo de entregar resultados de negocio como se define en la declaración de la visión de la cartera. La lista priorizada de pendientes de la cartera incorpora la lista priorizada de pendientes del programa de todos los programas en la cartera.

Los siguientes son ejemplos de proyectos, programas y carteras de diferentes industrias y sectores:

#### *Ejemplo 1: Empresa constructora*

- Proyecto—Construcción de una casa
- Programa—Construcción de un complejo de viviendas
- Cartera—Todos los proyectos de vivienda de la empresa

#### *Ejemplo 2: Organización aeroespacial*

- Proyecto—Construcción del vehículo de lanzamiento
- Programa—Lanzamiento exitoso de un satélite
- Cartera—Todos los programas de satélites activos

#### *Ejemplo 3: Empresa de Tecnología de la información (TI)*

- Proyecto—Desarrollo del módulo del carrito de compras
- Programa—Desarrollo de un sitio web de comercio electrónico (e-commerce) completamente funcional
- Cartera—Todos los sitios web desarrollados por la empresa hasta ahora

## 3.7.2 Scrum en Proyectos

Debido a que Scrum favorece a equipos pequeños, se pudiera pensar que este método sólo puede utilizarse en proyectos pequeños, pero ese no es el caso. Scrum también puede utilizarse con eficacia en proyectos de escala grande. Cuando se requieren más de diez personas para llevar a cabo el trabajo, se pueden formar múltiples equipos Scrum. El equipo del proyecto está formado por múltiples equipos Scrum que trabajan juntos para crear entregables y lanzamientos de productos, con el fin de lograr los resultados deseados para el proyecto en general.

Dado que un proyecto puede tener múltiples equipos Scrum trabajando en paralelo, la coordinación entre los diferentes equipos se convierte en algo sumamente importante. Por lo general, los equipos Scrum se comunican y coordinan entre sí en una variedad de maneras, pero el enfoque más común se conoce como reunión de Scrum de Scrums. Los miembros que representan a cada equipo Scrum se reúnen para discutir el progreso, los problemas y para coordinar las actividades entre los equipos. Estas reuniones son similares en formato a las reuniones diarias; sin embargo, la frecuencia del Scrum de Scrums podría ser en intervalos predeterminados o coordinada tal como es requerido por los diferentes equipos Scrum.

### 3.7.2.1 Reuniones de Scrum de Scrums

Una reunión de Scrum de Scrums es un elemento importante al escalar o ajustar Scrum a proyectos grandes. Por lo general, hay un representante en la reunión de cada uno de los equipos Scrum. Generalmente, el representante es el Scrum Master, pero también es común para cualquier persona del equipo asistir a la reunión si es necesario. Esta reunión es por lo general facilitada por el jefe Scrum Master, y su objetivo es centrarse en las áreas de coordinación e integración entre los diferentes equipos Scrum. Tal reunión se lleva a cabo en intervalos predeterminados o cuando lo requieran los equipos Scrum.

En organizaciones donde hay varios equipos Scrum trabajando en varias partes de un proyecto a la misma vez, el Scrum de Scrums se puede escalar a otro nivel de lo que se conoce como reunión de Scrum de Scrum de Scrums. En esta situación, este tipo de reunión mantiene la coordinación de cada grupo de los equipos Scrum, y luego se puede llevar a cabo una reunión de Scrum de Scrum de Scrums para coordinar e integrar los proyectos a un nivel mayor. Los equipos tienen que evaluar cuidadosamente los beneficios de contar con una reunión de este tipo, ya que la tercera capa añade una cantidad significativa de complejidad logística.

La Figura 3-4 ilustra el concepto de las reuniones de Scrum of Scrums y Scrum de Scrum de Scrums

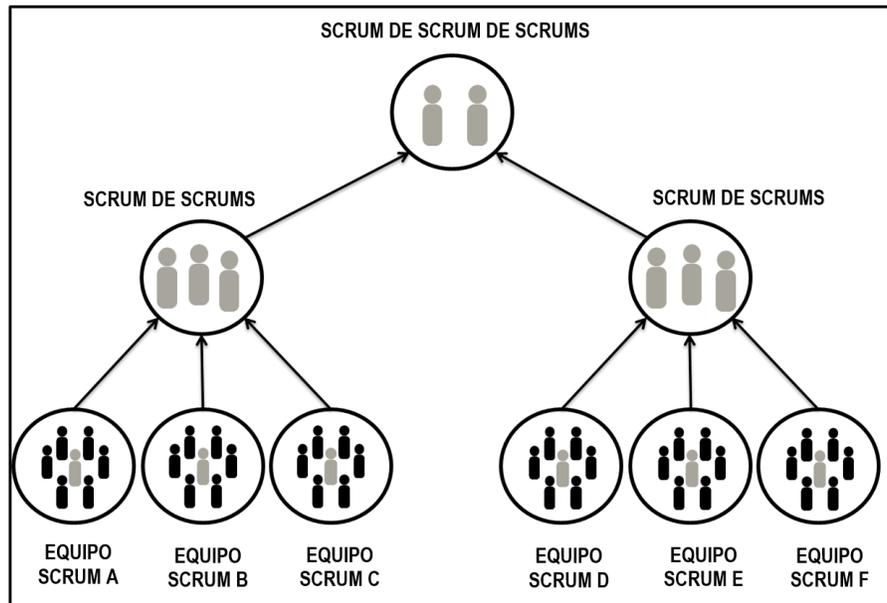


Figura 3-4: Reunión de Scrum de Scrums (SoS)

En este ejemplo, hay seis equipos Scrum que trabajan simultáneamente. Los equipos A, B y C están trabajando en las partes de un proyecto relacionado, mientras que los equipos D, E y F están trabajando en porciones de otro proyecto relacionado. Una reunión de Scrum de Scrum se lleva a cabo para coordinar las interdependencias entre los proyectos relacionados. Una reunión de Scrum de Scrum de Scrums se llevará a cabo para coordinar y gestionar las dependencias en todos los proyectos.

### 3.7.3 Scrum en carteras y programas

#### 3.7.3.1 Carteras

En las carteras, unos roles importantes para la gestión de la cartera del Scrum son:

1. **Propietario del producto de la cartera**—Define los objetivos estratégicos y las prioridades de la cartera.
2. **Scrum Master de la cartera**—Resuelve problemas, elimina Impedimentos, facilita, y lleva a cabo las reuniones para la cartera.

Estas funciones son similares a las del propietario del producto y el Scrum Master, con la diferencia que satisfacen las necesidades de su cartera o de la empresa en lugar de simplemente las de un equipo Scrum.

### 3.7.3.2 Programas

En los programas, los roles importantes para la gestión de programas de Scrum son:

1. **Propietario del producto del programa**—define los objetivos y las prioridades estratégicas para el programa.
2. **Scrum Master del programa**—Resuelve problemas, remueve impedimentos, facilita, y lleva a cabo reuniones para el programa.

Estas funciones son similares a las del propietario del producto y el Scrum Master, con la diferencia que satisfacen las necesidades de su programa o unidad de negocio en lugar de las de sólo un equipo Scrum.

La Figura 3-5 ilustra cómo Scrum puede utilizarse en toda la organización para las carteras, programas o proyectos.

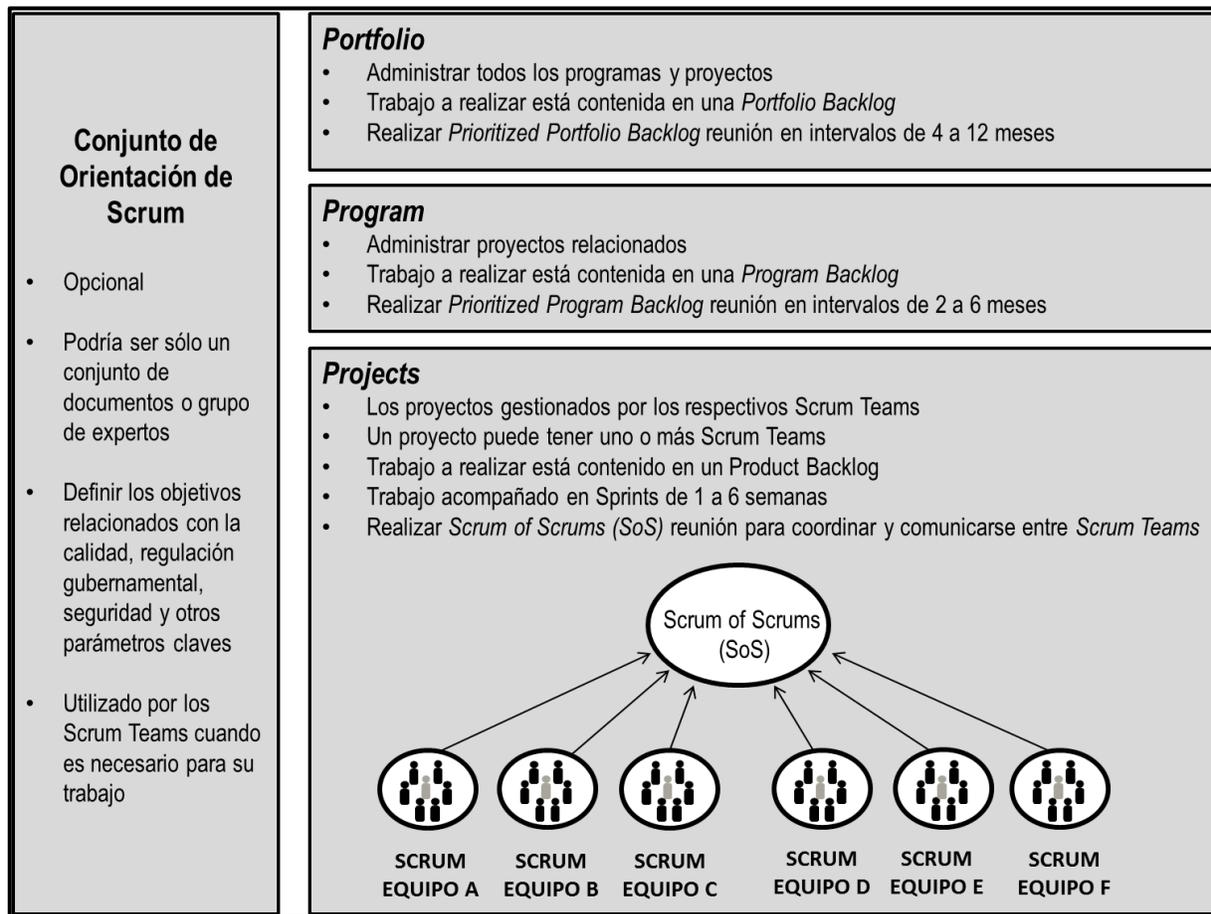


Figura 3-5: Scrum en toda la organización para proyectos, programas y carteras

### 3.7.3.3 Trabajar con equipos de carteras y programas

Al aplicar Scrum para gestionar proyectos en el marco de un programa o una cartera, se recomienda que los principios generales de Scrum que se presentan en esta guía se cumplan. Sin embargo, se entiende que, a fin de adaptar el programa en su totalidad o actividades relacionadas con la cartera y las interdependencias, pueden ser necesarios pequeños ajustes en el conjunto de herramientas, así como a la estructura organizativa. Si existe un cuerpo de asesoramiento de Scrum, éste puede ser responsable de examinar la organización a diferentes niveles para entender y definir la aplicación adecuada de Scrum, y actuar como facilitador de consulta para todos los que trabajan en un proyecto, programa o cartera.

Los carteras y programas cuentan con equipos separados y con diferentes conjuntos de objetivos. Los equipos de gestión de programas tienen por objetivo ofrecer capacidades y llevar a cabo ciertas metas que contribuyan a objetivos específicos del programa. Por el contrario, el equipo de la cartera tiene que equilibrar los objetivos de los distintos programas para alcanzar los objetivos estratégicos de la organización en su totalidad.

### 3.7.3.4 Gestión de la comunicación con equipos de carteras y programas

Los problemas y los asuntos que se enfrentan al utilizar Scrum dentro de un programa o cartera implican principalmente la coordinación entre los numerosos equipos. Esto puede conducir al fracaso si no se maneja con cuidado. Las herramientas que se utilizan para la comunicación deben ampliarse para que coincidan con los requisitos de los varios equipos que participan en un programa o una cartera. Cada equipo Scrum debe atender no sólo la comunicación interna, sino también la comunicación externa con otros equipos y los socios relevantes del programa o cartera.

## 3.7.4 Mantenimiento de la participación de los socios

Scrum requiere el apoyo completo de los socios de los proyectos. La responsabilidad de mantener la participación de los socios depende del propietario del producto. Las siguientes son las acciones que recomiendan para el mantenimiento de la participación y el apoyo de los socios.

- Asegurar la colaboración efectiva y la participación de los socios en el proyecto
- Evaluar continuamente el impacto en el negocio
- Mantener una comunicación regular con los socios
- Administrar las expectativas de los socios

Un socio clave es el patrocinador—el individuo que provee los fondos y otros recursos para un proyecto. Los patrocinadores quieren entender los resultados financieros relacionados con un producto o servicio, y están por lo general más interesados en los resultados finales, que con las tareas individuales.

Es importante que los patrocinadores que financian el proyecto tengan claridad sobre los siguientes asuntos:

- Beneficios de la implementación de Scrum

- Plazos del objetivo y los costos estimados de los proyectos Scrum
- Los riesgos en general involucrados en proyectos Scrum y las medidas para mitigarlos
- Fechas de lanzamiento esperadas y entregables finales

### 3.8 Resumen de responsabilidades

Rol	Responsabilidades
Cuerpo de asesoramiento de Scrum	<ul style="list-style-type: none"> <li>• Establecer las directrices generales y los parámetros para el desarrollo de descripciones de roles para los miembros del equipo Scrum</li> <li>• Actuar como consultor de proyectos en toda la organización a diferentes niveles</li> <li>• Entender y definir los niveles apropiados de la agrupación, los roles y las reuniones de los proyectos Scrum</li> </ul>
Propietario de la cartera del producto	<ul style="list-style-type: none"> <li>• Definir los objetivos y las prioridades estratégicas de las carteras</li> </ul>
Scrum Master de la cartera	<ul style="list-style-type: none"> <li>• Resolver problemas y coordinar reuniones de carteras</li> </ul>
Propietario del producto del programa	<ul style="list-style-type: none"> <li>• Definir los objetivos estratégicos y las prioridades de los programas</li> </ul>
Scrum Master del programa	<ul style="list-style-type: none"> <li>• Resolver problemas y coordinar reuniones para los programas</li> </ul>
Socio(s)	<ul style="list-style-type: none"> <li>• Es un término colectivo que incluye a clientes, usuarios y patrocinadores</li> <li>• Frecuentemente se relaciona con el propietario del producto, el Scrum Master y el equipo Scrum para proporcionarles las entradas y facilitar la creación de los entregables del proyecto.</li> </ul>
Propietario del producto	<ul style="list-style-type: none"> <li>• Crear requisitos globales iniciales del proyecto y poner el proyecto en marcha</li> <li>• Seleccionar a las personas adecuadas para los roles de Scrum Master y el equipo Scrum</li> <li>• Proporcionar los recursos financieros iniciales y continuos para el proyecto</li> <li>• Determinar la visión del producto</li> <li>• Evaluar la viabilidad y garantizar la entrega del producto o servicio</li> <li>• Garantizar la transparencia y la claridad de los elementos de la lista priorizada de pendientes del producto</li> <li>• Decidir el contenido mínimo del lanzamiento comercial</li> <li>• Proporcionar los criterios de aceptación para las historias de usuario que se desarrollarán en un sprint</li> <li>• Inspeccionar los entregables</li> <li>• Decidir la duración del sprint</li> </ul>
Scrum Master	<ul style="list-style-type: none"> <li>• Asegurar que los procesos de Scrum se sigan correctamente por todos los miembros del equipo, incluyendo el propietario del producto</li> <li>• Asegurar que el desarrollo del producto o servicio esté progresando sin problemas,</li> </ul>

	<p>y que los miembros del equipo Scrum tengan todas las herramientas necesarias para hacer el trabajo</p> <ul style="list-style-type: none"> <li>• Supervisar las reuniones de planificación del lanzamiento y los horarios de otras reuniones</li> </ul>
Equipo Scrum	<ul style="list-style-type: none"> <li>• Asumir la responsabilidad colectiva y asegurar que los entregables del proyecto se creen según las necesidades</li> <li>• Asegurar al propietario del producto y al Scrum Master que se está realizando el trabajo asignado de acuerdo al plan</li> </ul>

Tabla 3-4: Resumen de las responsabilidades pertinentes a la organización

### 3.9 Scrum vs. Gestión de proyectos tradicional

La estructura de la organización y definición de roles y responsabilidades asociadas son algunas de las áreas en las que Scrum difiere de manera importante de los métodos tradicionales de gestión de proyectos.

En los métodos tradicionales de gestión de proyectos, la estructura de la organización es jerárquica y la autoridad para todos los aspectos del proyecto se delega desde el nivel superior al inferior (por ejemplo, el patrocinador del proyecto delega autoridad al gerente del proyecto, y éste a su vez la delega a los miembros del equipo). Los métodos tradicionales de gestión de proyectos hacen hincapié en la rendición de cuentas del individuo hacia las responsabilidades del proyecto, en lugar de la responsabilidad del grupo o la rendición de cuentas hacia el proyecto. Cualquier desviación de la autoridad delegada es vista como una señal de problemas y puede ser escalada a un nivel más alto en la jerarquía de la organización. Por lo general, es el director del proyecto, quien es responsable de la finalización con éxito del proyecto, así como de la toma de decisiones sobre diversos aspectos del proyecto, incluyendo el inicio, planificación, estimación, ejecución, seguimiento y control, y cierre.

El énfasis en Scrum es la auto-organización y la auto-motivación, donde el equipo asume una mayor responsabilidad en la toma de un proyecto exitoso. Esto también asegura que existe un sentido de compromiso del equipo y responsabilidad compartida. Esto, a su vez, da lugar a la motivación del equipo que lleva a una optimización de su eficiencia. El propietario del producto, el Scrum Master y el equipo Scrum trabajan de cerca con el(los) socio(s) relevante(s) para ajustar los requisitos a medida que avanzan a través de los procesos de desarrollo de épica(s), la creación de la lista priorizada de pendientes del producto, y la creación de las historias de usuario. Esto asegura que no hay margen para la planificación aislada en Scrum. El conocimiento y las experiencias del equipo sobre el desarrollo de productos se utilizan para evaluar las entradas necesarias para planificar, calcular y ejecutar el trabajo del proyecto. La colaboración entre los miembros del equipo principal de Scrum asegura que el proyecto se lleve a cabo en un ambiente innovador y creativo el cual es propicio para el crecimiento y la armonía del equipo.

## 3.10 Teorías de recursos humanos populares y su relevancia para Scrum

### 3.10.1 Modelo de dinámica de grupo de Tuckman

El enfoque y método de Scrum pueden parecer al principio muy diferentes y difíciles para un nuevo equipo Scrum. Al igual que cualquier otro equipo nuevo, un nuevo equipo Scrum se desarrolla, por lo general, a través de un proceso de cuatro etapas durante su primer proyecto de Scrum. Este proceso se conoce como modelo de dinámica de grupo de Tuckman (Tuckman, 1965). La idea principal es que las cuatro etapas—Formación, Enfrentamiento, Normalización y Desempeño—son imprescindibles para que un equipo se desarrolle y mitigue los problemas y desafíos, encuentre soluciones, planifique el trabajo, y entregue resultados.

Las cuatro etapas del modelo son las siguientes:

1. Formación—Esto a menudo se experimenta como un escenario ameno porque todo es nuevo y el equipo aún no ha encontrado ninguna dificultad con el proyecto.
2. Enfrentamiento—Durante esta etapa, el equipo trata de cumplir con el trabajo; sin embargo, puede encontrar conflictos de poder y, con frecuencia, existe un caos o confusión entre los miembros del equipo.
3. Normalización—Esto es cuando el equipo comienza a madurar, resolver sus diferencias internas, y encontrar soluciones para así trabajar juntos. Se considera un período de ajuste.
4. Desempeño—Durante esta etapa, el equipo está unido y opera en su nivel más alto en términos de rendimiento. Los miembros se han convertido en un equipo eficiente de profesionales que son consistentemente productivos.

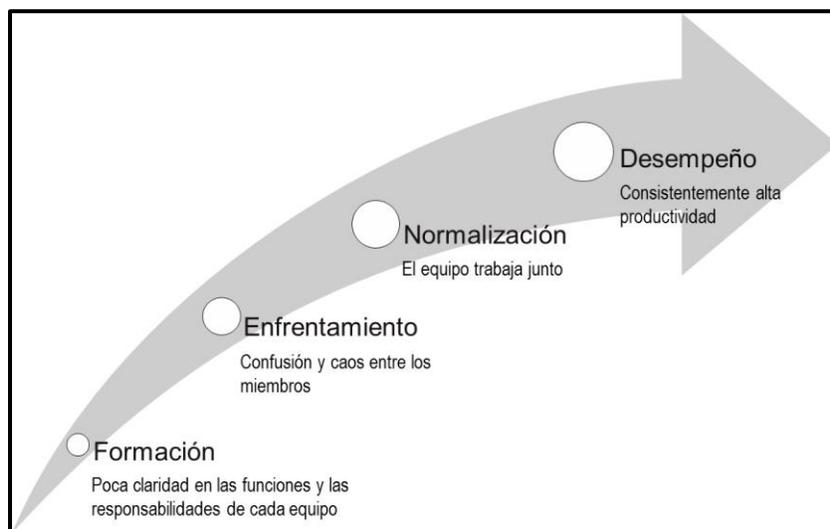


Figura 3-6: Etapas de Tuckman de desarrollo de grupos

### 3.10.2 Gestión de conflictos

Las organizaciones que aplican el marco de Scrum fomentan un ambiente abierto y de diálogo entre los empleados. Los conflictos entre los miembros del equipo Scrum generalmente se resuelven de forma independiente, con poca o ninguna participación de la gerencia o de otros fuera del equipo.

El conflicto puede ser saludable cuando promueve las discusiones del equipo y alienta al debate, ya que por lo general esto se traduce en beneficios para el proyecto y los miembros de los equipos respectivos. Por lo tanto, es importante que se fomente la resolución de los conflictos, promoviendo un entorno abierto donde los miembros del equipo se sientan alentados a expresar sus inquietudes y opiniones con los demás sobre el proyecto, y por último que estén de acuerdo en los entregables y cómo se realizará el trabajo en cada Sprint.

Las técnicas de gestión de conflictos se utilizan por los miembros del equipo para gestionar los conflictos que surgen durante un proyecto Scrum. Las fuentes de conflictos evolucionan principalmente debido a los horarios, prioridades, recursos, informes de jerarquía, cuestiones técnicas, procedimientos, personalidad, y costos.

### 3.10.3 Técnicas de gestión de conflictos

Por lo general, hay cuatro enfoques para la gestión de conflictos en una organización que aplica procesos de Scrum:

1. Ganar- Ganar
2. Perder-Ganar
3. Perder-Perder
4. Ganar-Perder

#### 3.10.3.1 Ganar-Ganar

Por lo general, es mejor para los miembros del equipo hacer frente a los problemas directamente con una actitud de cooperación y un diálogo abierto para trabajar a través de cualquier desacuerdo así llegar a un consenso. Este enfoque se denomina ganar-ganar. Las organizaciones que implementan Scrum deben promover un ambiente donde los empleados se sientan cómodos para hablar y enfrentarse a los problemas o asuntos, y trabajar resolverlos para llegar a los resultados de ganar-ganar.

### 3.10.3.2 Perder-Ganar

Algunos miembros del equipo a veces pueden sentir que sus contribuciones no son reconocidas ni valoradas por los demás, o que no están siendo tratados con igualdad. Esto puede conducir a que no contribuyan de manera efectiva al proyecto y a que estén de acuerdo con todo lo que se les dice que se necesita hacer, incluso si están en desacuerdo. Este enfoque se llama perder-ganar. Esta situación puede ocurrir si hay miembros en el equipo (inclusive los administradores) que utilizan un estilo autoritario o directivo para emitir órdenes y/o no tratan a todos los miembros del equipo con igualdad. Este enfoque no es una técnica que se prefiera para la gestión de conflictos en los proyectos Scrum, ya que la contribución activa de todos los miembros del equipo es obligatoria para completar con éxito cada uno de los sprints. El Scrum Master debe fomentar la participación de los miembros del equipo que parecen estar retirándose de las situaciones de conflicto. Por ejemplo, es importante para todos los miembros del equipo que hablen y contribuyan en cada reunión diaria para que cualquier problema o impedimento se haga saber y se maneje eficazmente.

### 3.10.3.3 Perder-Perder

En situaciones de conflicto, los miembros del equipo pueden intentar negociar o buscar soluciones que aporten sólo un grado parcial o medida temporal de satisfacción a las partes en una disputa. Esta situación podría suceder en equipos Scrum si los miembros del equipo tratan de negociar soluciones sub-óptimas a un problema. Este enfoque suele implicar un poco de “dar para recibir” para satisfacer a cada miembro del equipo, en lugar de tratar de realmente resolver el verdadero problema. En general, esto se traduce en un resultado de perder-perder para los individuos involucrados y, en consecuencia, el proyecto. El equipo Scrum debe tener cuidado de que los miembros del equipo no lleguen entren en una mentalidad de perder-perder. Las reuniones diarias y otras reuniones de Scrum se llevan a cabo para asegurar que los problemas reales se resuelvan a través de discusiones mutuas.

### 3.10.3.4 Ganar-Perder

A veces, un Scrum Master u otro miembro del equipo influyente, pueden creer ser un líder *de facto* o gerente y tratar de ejercer su punto de vista, sin tomar en cuenta los puntos de vista de los demás. Esta técnica de gestión de conflictos a menudo se caracteriza por la competitividad y por lo general resulta en ganar-perder. Este enfoque no es recomendable cuando se trabaja en proyectos Scrum, porque los equipos Scrum son por naturaleza auto-organizados y empoderados, donde ninguna persona tiene verdadera autoridad sobre otro miembro del equipo. Aunque el equipo Scrum puede incluir a personas con diferentes niveles de experiencia y conocimientos, cada miembro se trata con igualdad, y ninguna persona tiene la autoridad de ser la autoridad máxima.

### 3.10.4 Estilos de liderazgo

Los estilos de liderazgo varían dependiendo de la organización, la situación, e incluso en los individuos y los objetivos específicos del proyecto Scrum. Algunos estilos de liderazgo comunes son los siguientes:

- **Líder servicial**—El liderazgo servicial implica escuchar cuidadosamente, tener empatía, comprometerse al servicio, tener visión, y compartir el poder y la autoridad con los miembros del equipo. Este estilo de liderazgo logra resultados centrándose en las necesidades del equipo. Asimismo, es la personificación del rol del Scrum Master.
- **Líder delegador**—Los líderes delegadores participan en la mayoría de la toma de decisiones; sin embargo, delegan parte de las responsabilidades de planificación a los miembros del equipo, particularmente si son competentes para manejar tareas. Este estilo de liderazgo es apropiado en situaciones en las que el líder está en sintonía con los detalles específicos de proyectos y cuando el tiempo es limitado.
- **Autocrático**—Los líderes autocráticos toman decisiones por su cuenta, dándoles poco o nada de tiempo a los miembros del equipo antes de tomar una decisión. Este estilo de liderazgo debe utilizarse solamente en raras ocasiones.
- **Líder directivo**—Los líderes directivos instruyen a los miembros del equipo sobre las tareas que se requieren y sobre cómo y cuándo deben llevarse a cabo.
- **Líder *Laissez Faire***—Es un estilo de liderazgo en el que el equipo se queda sin supervisión y donde el líder no interfiere con las actividades laborales diarias. Esto a menudo conduce a un estado de anarquía.
- **Líder de apoyo y entrenamiento**—Los líderes de apoyo y entrenamiento dan instrucciones; después apoyan y monitorean a los miembros del equipo al escuchar, ayudar, alentar y presentar una perspectiva positiva en tiempos de incertidumbre.
- **Líder enfocado en las tareas**—Los líderes que se enfocan en las tareas hacen cumplir la realización de las tareas y el apego a los plazos.
- **Líder asertivo**—Los líderes asertivos confrontan los problemas y demuestran confianza para establecer autoridad con respeto.

#### 3.10.4.1 Liderazgo servicial

El estilo de liderazgo preferido para los proyectos Scrum es el liderazgo servicial. Este término fue descrito por primera vez por Robert K. Greenleaf, en un ensayo titulado *El siervo como líder (The Servant as Leader)* en inglés). A continuación, se proporciona un extracto en el que se explica este concepto:

El líder servicial es primero que nada un servidor... Empieza con el sentimiento natural de que uno quiere servir, servir *primero*. Luego la elección consciente lleva a uno a aspirar a liderar. Esa persona es completamente diferente a aquel que es *líder primero*, tal vez debido a la necesidad de mando inusual o de adquirir posesiones materiales... El líder primero y el líder servicial son dos

tipos extremos. Entre ellos hay matices y mezclas que forman parte de la infinita variedad de la naturaleza humana...

La diferencia se manifiesta a sí misma en el cuidado del que es siervo primero para asegurarse de que las necesidades más prioritarias de la gente estén siendo atendidas. La mejor prueba, que es difícil de administrar, es: ¿A los que se les sirve, crecen como personas? ¿Mientras se les sirve, se vuelven más sanos, más sabios, más libres, más autónomos, más propensos a ser servidores ellos mismos? Y, ¿cuál es el efecto sobre los más desfavorecidos en la sociedad? ¿Van a ser beneficiados o al menos ya no más desfavorecidos? (Greenleaf 1970, 6)

Abundando sobre los escritos de Greenleaf, Larry Spears identifica diez rasgos que cada líder servicial eficaz debe poseer:

1. **Escuchar**—Se espera que los líderes serviciales escuchen con atención y sean receptivos a lo que se dice o no se dice. Ellos son capaces de ponerse en contacto con su voz interior para comprender y reflexionar sobre sus propios sentimientos.
2. **Empatía**—Los líderes serviciales buenos aceptan y reconocen a los individuos por sus destrezas únicas y habilidades especiales. Ellos asumen que los trabajadores tienen buenas intenciones y los aceptan como individuos, incluso cuando existen problemas de comportamiento o rendimiento.
3. **Recuperación**—La motivación y la capacidad de recuperarse a sí mismo y la relación con los demás es un fuerte rasgo de los líderes serviciales. Ellos toman la oportunidad de ayudar a sus colegas que están pasando por dolor emocional.
4. **Toma de conciencia**—Ser consciente, especialmente ser auto-consciente, es un rasgo de los líderes serviciales. Esto les permite entender mejor e integrar los problemas, tales como los relacionados con la ética, el poder y los valores.
5. **Persuasión**—Los líderes serviciales usan la persuasión, en lugar de su posición de autoridad, para obtener el consenso del grupo y tomar decisiones. En lugar de forzar el cumplimiento y la coerción como es típico en algunos estilos autoritarios de gestión, los líderes serviciales practican la persuasión.
6. **Conceptualización**—Una habilidad especial de los líderes serviciales es ver y analizar los problemas (en una organización) desde una perspectiva conceptual y visionaria más amplia, en lugar de centrarse en los objetivos inmediatos a corto plazo.
7. **Prospectiva**—Su mente intuitiva les permite a los líderes serviciales utilizar y aplicar las lecciones del pasado y la realidad actual para prever el resultado de situaciones y decisiones actuales.

8. **Administración**—La administración exige un compromiso de servir a los demás. Los líderes serviciales prefieren la persuasión por encima del control para obtener la confianza de los demás en la organización.
9. **Compromiso con el crecimiento de los demás**—Los líderes serviciales tienen un profundo compromiso con el crecimiento de las personas dentro de su organización. Asumen la responsabilidad de nutrir el crecimiento personal, profesional y espiritual de los demás (por ejemplo, facilitando el acceso a los recursos para el desarrollo personal y profesional, alentando a los trabajadores a participar en la toma de decisiones).
10. **Desarrollo de una comunidad**—Los líderes serviciales están interesados en el desarrollo de comunidades dentro de un ambiente de trabajo. Esto es de gran importancia, en especial dado al cambio en muchas sociedades que dejan de ser comunidades pequeñas para convertirse en grandes instituciones que dan forma y que controlan las vidas humanas.

Scrum cree que todos los líderes de proyectos Scrum (incluyendo al Scrum Master y el propietario del producto) deben ser líderes serviciales que tengan las características mencionadas anteriormente.

### 3.10.5 Teoría de jerarquía de necesidades de Maslow

Maslow (1943) presenta una jerarquía de necesidades la cual reconoce que diferentes personas se encuentran en diferentes niveles en sus necesidades. Por lo general, la gente se preocupa de las necesidades fisiológicas y después se desplaza progresivamente hacia las necesidades jerárquicas.

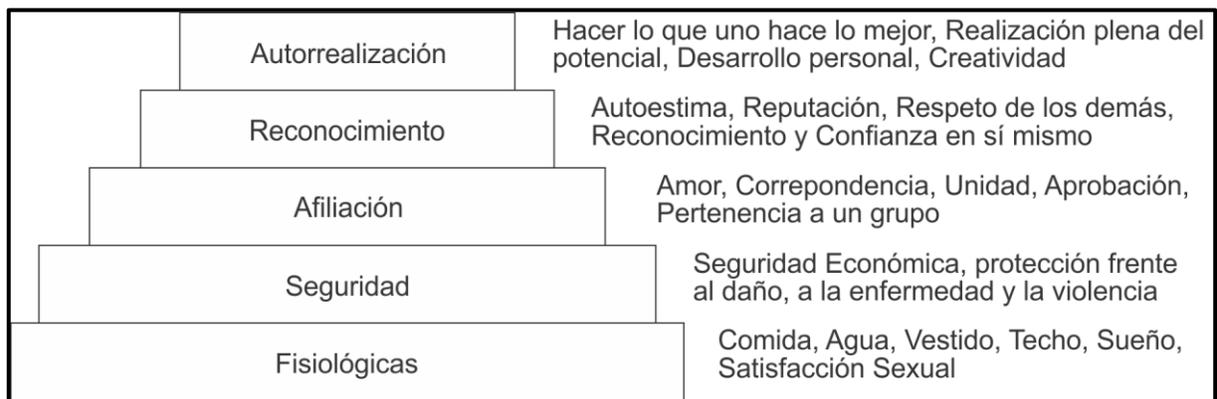


Figura 3-7: Teoría de jerarquía de necesidades de Maslow

Para tener éxito, un equipo Scrum necesita que tanto los miembros del equipo centrales como los no centrales hayan alcanzado los niveles de reconocimiento y autorrealización. El concepto de equipos auto-organizados, lo cual es un principio clave en Scrum, requiere que los miembros del equipo sean auto-motivados, que participen y contribuyan plenamente en el cumplimiento de los objetivos del proyecto.

Como líder, el Scrum Master tiene que entender en dónde es que cada persona del equipo está en relación la pirámide. Este entendimiento le ayudará a determinar el mejor enfoque para motivar a cada individuo.

Todas las personas fluctúan hacia arriba y abajo en los niveles de jerarquía de necesidades a lo largo de la vida. Esto puede ser debido a su propia motivación y esfuerzos para avanzar en la jerarquía o, a veces debido a factores fuera de su control. El objetivo del Scrum Master es trabajar con personas en el equipo para mejorar sus habilidades y conocimientos y ayudarlo/la a ascender en la jerarquía de necesidades. Este apoyo resulta en un equipo que está formado por individuos que están motivados y son fuertes colaboradores al proyecto y a la organización.

### 3.10.6 Teoría X y Teoría Y

Douglas McGregor (1960) propuso dos teorías de gestión:

- **Teoría X**— En la Teoría X, los líderes suponen que los empleados están inherentemente desmotivados y evitarán el trabajo si es posible, lo cual justifica un estilo autoritario de gestión.
- **Teoría Y**— En la Teoría Y, los líderes asumen que los empleados son auto motivados y buscan aceptar una mayor responsabilidad. La Teoría Y implica un estilo más participativo de gestión.

Es improbable que los proyectos Scrum tengan éxito si el Scrum Master o el propietario del producto son líderes de la Teoría X. Todos los líderes de proyectos Scrum deben apegarse a la Teoría Y, mediante la cual vean a los empleados por sus cualidades importantes, a la misma vez que ayudan a desarrollar las habilidades de los miembros de su equipo y empoderarlos. Es de suma importancia también expresar a los miembros del equipo un gran agradecimiento por el trabajo que han hecho para lograr los objetivos del proyecto.

## 4. JUSTIFICACIÓN DEL NEGOCIO

### 4.1 Introducción

El propósito de este capítulo es entender el concepto y el propósito de la justificación del negocio en su relación a los proyectos Scrum. Es importante que una organización lleve a cabo una adecuada justificación del negocio y elabore una declaración de la visión del proyecto antes viable de dar inicio a cualquier proyecto. Esto ayudará a las personas claves que toman decisiones a entender la necesidad empresarial de cambio o de un nuevo producto o servicio, así como la justificación para seguir adelante con un proyecto. Esto también ayuda a que el propietario del producto elabore una lista priorizada de pendientes del producto, tomando en cuenta las expectativas empresariales de la alta gerencia y de (los) socio(s).

La *justificación del negocio*, según se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, aplica a lo siguiente:

- Carteras, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se les entregará a los socios;
- Proyectos de cualquier tamaño o complejidad.

El término “producto” en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo se divide en las siguientes secciones:

**4.2 Guía de roles**—Esta sección ofrece orientación sobre cuáles secciones son relevantes para cada uno de los roles centrales de Scrum: El propietario del producto, el Scrum Master y el equipo Scrum.

**4.3 Entrega basada en el valor**—Esta sección describe el concepto del valor del negocio y su importancia en cualquier proyecto. Brinda también información sobre las responsabilidades de los distintos individuos, incluyendo el propietario del producto, quien participa en el logro del valor para el negocio.

**4.4 Importancia de la justificación del negocio**—En esta sección se detalla la importancia de la justificación del negocio, los factores que la determinan y cómo se mantiene y se verifica durante todo el proyecto.

**4.5 Técnicas de justificación del negocio**—Esta sección describe a detalle cómo se evalúa y verifica la justificación del negocio utilizando diversas herramientas.

**4.6 Justificación continua de valor**—En esta sección se detalla la importancia de la justificación continua de valor y abunda sobre la forma en la que esta se logra.

**4.7 Confirmar la realización de beneficios**—Esta sección describe la forma en la que se logran los beneficios durante el proyecto.

**4.8 Resumen de responsabilidades**—Esta sección define las responsabilidades pertinentes a la justificación del negocio para los miembros del equipo del proyecto en función de sus roles.

**4.9 Scrum vs. Gestión tradicional de proyectos**—Esta sección destaca los beneficios empresariales del método Scrum en comparación a los modelos tradicionales de gestión de proyectos.

## 4.2 Guía de roles

1. Propietario del producto—La justificación del negocio la realiza principalmente por el propietario del producto; por lo tanto, todo este capítulo aplica más a este rol.
2. Scrum Master—El Scrum Master debe estar familiarizado con todo este capítulo, con un enfoque principal en las secciones 4.3, 4.4, 4.6, 4.7 y 4.8.
3. Equipo Scrum—El equipo Scrum debe enfocarse principalmente en las secciones 4.3, 4.7 y 4.8.

## 4.3 Entrega basada en valor

Un proyecto es un negocio colaborativo para crear, ya sea nuevos proyectos o servicios, o para entregar resultados según se define en la declaración de visión del proyecto. Los proyectos generalmente se ven afectados por limitaciones de tiempo, costos, enfoque, calidad, personal y capacidades organizacionales. Por lo general, se busca que los resultados que generen los proyectos resulten en algún tipo de valor de negocio o servicio.

Dado a que el valor es una razón principal de cualquier organización para seguir adelante con un proyecto, la entrega basada en valor (del inglés: *Value-Driven Delivery*) debe ser el principal enfoque. El ofrecer valor es algo que está arraigado en el marco de Scrum. Scrum facilita la entrega anticipada de valor en el proyecto y lo sigue haciendo a lo largo del ciclo de vida del mismo.

Una de las características claves de cualquier proyecto es la incertidumbre de los resultados. Es imposible garantizar el éxito del proyecto, independientemente de su tamaño o complejidad. Por lo tanto, tomando en cuenta esta incertidumbre de alcanzar el éxito, es importante empezar a entregar resultados durante el proyecto tan pronto como sea posible. Esta entrega temprana de buenos resultados, y por lo tanto de valor, brinda una oportunidad para la reinversión, demostrando el valor del proyecto a los socios interesados.

A fin de ofrecer una entrega basada en valor, es importante:

1. Entender lo que agrega valor a los clientes y usuarios y dar prioridad a los requerimientos de alto valor que encabezan la lista priorizada de pendientes del producto.
2. Disminuir la incertidumbre y atender constantemente de los riesgos que potencialmente pudieran reducir el valor en caso de materializarse. También es importante trabajar en estrecha colaboración con los socios del proyecto mostrándoles incrementos del producto al final de cada sprint, lo cual permite una gestión efectiva de cambios.
3. Crear entregables basados en las prioridades determinadas por la producción de incrementos del producto potencialmente entregables durante cada sprint. De esta forma, los clientes empiezan a darse cuenta del valor desde el principio del proyecto.

4

En Scrum, el concepto de la entrega basada en valor, hace que el marco de Scrum sea muy atractivo para los socios del negocio y para la alta gerencia. Este concepto es muy diferente en comparación con los modelos tradicionales de gestión de proyectos, donde:

1. Los requerimientos no son priorizados por el valor del negocio;
2. Los cambios en los requerimientos después del inicio del proyecto resultan difíciles y solo se pueden hacer mediante un tardado proceso de gestión;
3. El valor se logra sólo al final del proyecto, al entregarse el producto o servicio final.

La figura 4-1 contrasta la entrega basada en valor en Scrum en comparación a los proyectos tradicionales.

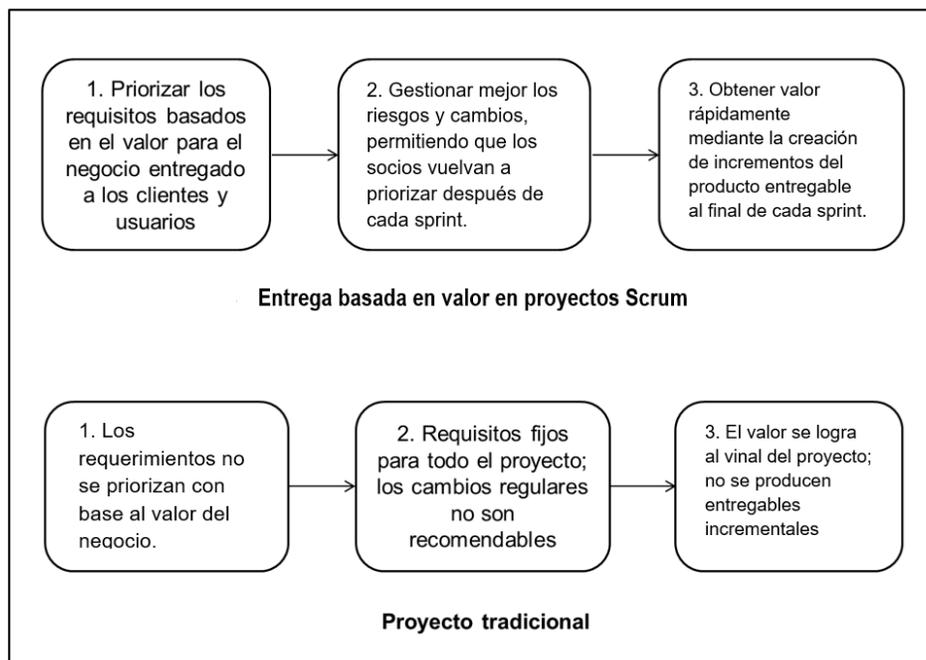


Figura 4-1: Entrega de valor en Scrum vs. proyectos tradicionales

### 4.3.1 Responsabilidades del propietario del producto en la justificación del negocio

La responsabilidad de priorizar y entregar valor de negocio para los proyectos en una organización le corresponde principalmente al propietario del producto. Para los programas y carteras, la responsabilidad recae en el propietario del producto del programa y en el propietario del producto de la cartera, respectivamente. Su función es la de actuar como representantes efectivos del cliente y/o del patrocinador. Las directrices para la evaluación y medición del valor de negocio típicamente pueden ser establecidas por un cuerpo de asesoramiento de Scrum.

La figura 4-2 ilustra las responsabilidades de la justificación del negocio en orden jerárquico.

4

Propietario del producto de la cartera	<ul style="list-style-type: none"> <li>• Proporciona el valor para las carteras</li> <li>• Crea la justificación del negocio para las carteras</li> <li>• Proporciona una guía de valor para los programas</li> <li>• Aprueba la justificación del negocio para los programas</li> </ul>
Propietario del producto del programa	<ul style="list-style-type: none"> <li>• Proporciona el valor de los programas</li> <li>• Crea la justificación del negocio para los programas</li> <li>• Proporciona una guía de valor para los proyectos</li> <li>• Aprueba la justificación del negocio para los proyectos</li> </ul>
Propietario del producto	<ul style="list-style-type: none"> <li>• Entrega el valor para proyectos</li> <li>• Crea la justificación del negocio para los proyectos</li> <li>• Le confirma la realización de beneficios a los socios</li> </ul>

Figura 4-2: Jerarquía de responsabilidades en la justificación del negocio

### 4.3.2 Responsabilidades de otros roles de Scrum en la justificación del negocio

Es importante señalar que, aunque el propietario del producto es el responsable principal de la justificación del negocio, hay otras personas trabajando en proyectos Scrum que también contribuyen considerablemente de la siguiente forma:

1. El **patrocinador** proporciona los fondos para el proyecto y supervisa constantemente el proyecto para confirmar el logro de los beneficios.

2. Los **clientes** y **usuarios** participan en la definición de la lista priorizada de los requisitos y de las historias de usuario en la lista priorizada de pendientes del producto; en la revisión de los entregables después de cada sprint o lanzamiento y en la confirmación de los beneficios.
3. El **cuerpo de asesoramiento de Scrum** puede proporcionar directrices y recomendaciones relacionadas a las técnicas de justificación del negocio y confirmar la realización de beneficios, etc. Tales directrices y recomendaciones pueden después remitirse los equipos centrales de Scrum y a los socios.
4. El **Scrum Master** facilita la creación de entregables del proyecto; está a cargo de la gestión riesgos, de cambios e impedimentos durante los procesos de realizar la reunión diaria de pie y de la retrospectiva del sprint, entre otros. El Scrum Master se coordina con el equipo Scrum para crear los entregables y con el propietario del producto y otros socios para asegurar que los beneficios del proyecto se materialicen.
5. El **equipo Scrum** trabaja en la creación de entregables del proyecto y contribuye a la creación de valor del negocio para todos los socios y para el proyecto. El equipo Scrum también participa en el *desarrollo de épica(s)*; en la *creación de la lista priorizada de pendientes del producto*; en la *creación de historias de usuario*; en la *aprobación, estimación y asignación de historias de usuario*; así como en procesos asociados donde se definen y se priorizan los requerimientos del negocio. El equipo Scrum también ayuda en la identificación de riesgos y presenta solicitudes de cambio para las mejoras durante las reuniones de retrospectiva del sprint y otras reuniones.

## 4.4 Importancia de la justificación del negocio

La justificación del negocio demuestra las razones para emprender un proyecto. Responde a la pregunta: “¿Por qué es necesario este proyecto?” La justificación del negocio es lo que impulsa todas las decisiones relacionadas aun proyecto. Por lo tanto, es importante evaluar la viabilidad de un proyecto, no solo antes de comprometerse a gastos o inversiones considerables en las etapas iniciales, sino también a lo largo del ciclo de vida del proyecto. Un proyecto debe darse por terminado si se encuentra que no es viable; la decisión debe ser escalada a los socios pertinentes y a la alta gerencia. La justificación del negocio de un proyecto debe ser evaluada al inicio de este, en intervalos predefinidos durante todo el proyecto y en cualquier momento cuando surgen grandes problemas o riesgos que amenacen su viabilidad.

### 4.4.1 Factores que se utilizan para determinar la justificación del negocio

Existen numerosos factores que un propietario del producto debe tomar en cuenta para determinar la justificación del negocio de un proyecto. Los siguientes son algunos de los factores más importantes:

**1. Razonamiento del proyecto**

El razonamiento del proyecto incluye todos los factores que este requiere, ya sean positivos, negativos, elegidos o no (por ejemplo: capacidad inadecuada para cumplir con la demanda actual y la demanda prevista, la disminución en la satisfacción del cliente, baja utilidad, requerimientos legales, etc.).

**2. Necesidades del negocio**

Las necesidades del negocio son aquellos resultados del negocio que se espera que cumpla el proyecto, tal como se documenta en la declaración de visión del proyecto.

**3. Beneficios del proyecto**

Los beneficios del proyecto incluyen todas las mejoras cuantificables de un producto, servicio o resultado que se pudieran obtener durante la conclusión satisfactoria de un proyecto.

**4. Costo de oportunidad**

El costo de oportunidad es el valor de la siguiente mejor opción de negocio o proyecto que fue descartada en favor del proyecto seleccionado.

**5. Riesgos mayores**

Los riesgos incluyen eventos inciertos o no planeados que pudieran afectar la viabilidad y el posible éxito del proyecto.

**6. Escalas de tiempo del proyecto (*Project Timescales*)**

Las escalas de tiempo reflejan la duración de un proyecto y el tiempo durante el cual se obtendrán los beneficios del mismo.

**7. Costos del proyecto**

Los costos del proyecto son las inversiones y demás costos de desarrollo en un proyecto.

### 4.4.2 Justificación del negocio y el ciclo de vida del proyecto

Antes de iniciar un proyecto, primero se evalúa la justificación del negocio y se verifica constantemente durante todo el ciclo de vida del proyecto. Los siguientes pasos captan la forma en la que se determina la justificación del negocio:

**1. Evaluar y presentar un caso de negocio**

La justificación del negocio para un proyecto normalmente la analiza y la confirma el propietario del producto. Se documenta y se presenta en forma de caso de negocio del proyecto antes de la fase de inicio y consiste en tomar en cuenta los diversos factores especificados en la sección 4.4.1. Una vez documentado, el propietario del producto debe crear una declaración de la visión del proyecto y obtener la aprobación de aquellos que toman las decisiones claves en la organización. Generalmente, son los ejecutivos y/o algún consejo administrativo del proyecto o del programa.

## 2. Justificación continua de valor

Una vez que los encargados de tomar decisiones aprueban la declaración de la visión del proyecto, esta se utiliza como base de referencia y forma la justificación del negocio. La justificación del negocio se valida durante toda la ejecución del proyecto, por lo general en intervalos predefinidos, como en reuniones de la cartera, del programa o de la lista priorizada de pendientes del producto y cuando se identifican los principales problemas y riesgos que amenazan la viabilidad del proyecto. Esto puede darse en varios procesos de Scrum, incluyendo el proceso de *realización de reunión diaria de pie* y en el *mantenimiento de la lista priorizada de pendientes del producto*. A lo largo del proyecto, el propietario del producto debe mantener actualizada la justificación del negocio en la declaración de la visión del proyecto con información relevante del proyecto para que los que toman decisiones importantes continúen tomando decisiones informadas.

### 2. Confirmar la realización de beneficios

El propietario del producto confirma el logro de los beneficios organizacionales durante el proyecto y al completar las historias de usuario en la lista priorizada de pendientes del producto. Los beneficios de los proyectos Scrum se materializan durante los procesos de *demonstración y validación del sprint*, *retrospectiva del sprint*, *envío de entregables* y *retrospectiva del proyecto*.

La figura 4-3 resume los pasos para determinar justificación del negocio.

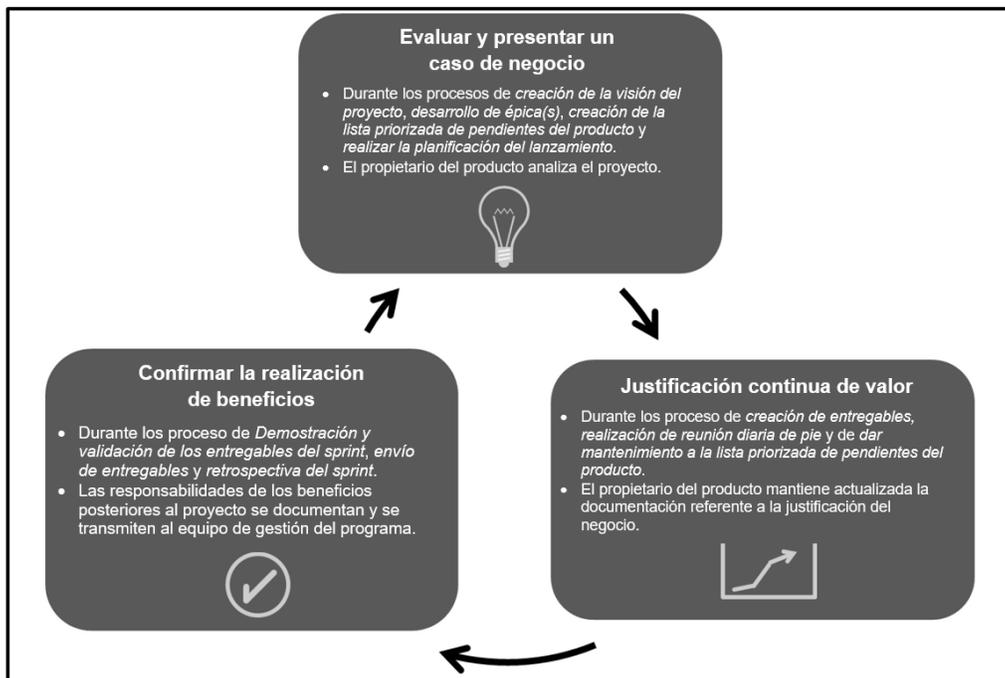


Figura 4-3: Justificación del negocio y el ciclo de vida del proyecto

## 4.5 Técnicas de justificación del negocio

Las siguientes secciones abordan algunas de las herramientas que se utilizan para valorar y evaluar la justificación del negocio, así como otros aspectos relacionados con la justificación y selección del proyecto. No es necesario, ni se recomienda utilizar todas las técnicas disponibles para cada proyecto. Algunas técnicas no son apropiadas dependiendo del proyecto específico y también se pueden utilizar para evaluar los proyectos en forma individual o para comparar el valor esperado de múltiples proyectos.

El cuerpo de asesoramiento de Scrum —que puede consistir en un grupo de expertos o en un conjunto de documentos sobre normas y procedimientos de la organización—, define los lineamientos y parámetros que se utilizarán para evaluar el valor del negocio. Sin embargo, cada propietario del producto es responsable de llevar a cabo las actividades para verificar y dar seguimiento al valor de negocio para sus respectivos proyectos, programas o carteras.

4

### 4.5.1 Estimación del valor del proyecto

El valor que habrán de brindar los proyectos empresariales puede estimarse utilizando diversos métodos, tales como el retorno sobre la inversión (RSI), valor presente neto (VPN) y la tasa interna de retorno (TIR).

#### 1. Retorno sobre la inversión (RSI)

El retorno sobre la inversión (RSI), al utilizarse para la justificación de un proyecto, evalúa los ingresos netos esperados que se buscan obtener a partir de un proyecto. Se calcula deduciendo los costos esperados o la inversión en un proyecto de su ingreso previsto; después se divide (la utilidad neta) por los costos previstos a fin de obtener la tasa de retorno. Otros factores, tales como la inflación y las tasas de interés sobre el dinero prestado pudieran incluirse en los cálculos del RSI.

Fórmula del RSI:

$$\text{RSI} = (\text{Ingresos del proyecto} - \text{Costo del proyecto}) / \text{Costo del proyecto}$$

*Ejemplo:* El RSI para un proyecto que tendrá un costo de \$125,000 en desarrollarse y con beneficios económicos estimados en \$300,000, se calcula de la siguiente forma:

$$\text{RSI} = (\$300,000 - \$125,000) / \$125,000 = 1.4$$

Por lo tanto, el RSI es 1.4 veces la inversión (o 140 %).

Los incrementos frecuentes del servicio o producto son una base fundamental de Scrum que permiten la verificación temprana del ROI. Esto ayuda en la evaluación de la justificación continua de valor.

## 2. Valor presente neto (VPN)

El valor presente neto (VPN) es un método que se utiliza para determinar el valor neto actual de un futuro beneficio económico, dada una inflación o tasa de interés prevista. En otras palabras, el VPN es el ingreso total esperado o los ingresos de un proyecto, restando el costo total previsto del proyecto y tomando en cuenta el valor actual de la divisa.

*Ejemplo:* ¿Cuál de los siguientes dos proyectos es la mejor opción si se utiliza el VPN como criterio de selección?

- El proyecto A tiene un VPN de \$1,500 y se completará en 5 años.
- El proyecto B tiene un VPN de \$1,000 y se completará en 1 año.

*Solución:* El proyecto A, ya que su VPN es más elevado. Aquí no se toma en cuenta el hecho de que el proyecto B tiene una duración más corta que el proyecto A, pues el tiempo ya está representado en los cálculos del VPN (debido a que es el valor actual y no el valor futuro que se considera en el cálculo).

## 3. Tasa interna de retorno (TIR)

La tasa interna de retorno (TIR), es una tasa de descuento sobre una inversión en la cual el valor actual de los flujos de efectivo se iguala al valor actual del flujo de salidas de efectivo a fin de evaluar la tasa de retorno del proyecto. Al hacer un comparativo de los proyectos, generalmente resulta mejor aquel que cuenta con una tasa interna de retorno más elevada.

Aunque no se utiliza para justificar proyectos con tanta frecuencia como con algunas otras técnicas, tales como el VPN, la TIR es un concepto importante que hay que conocer.

*Ejemplo:* Basado en el TIR, ¿cuál proyecto es más conveniente?

- Proyecto A, que tiene una TIR del 15 % y se completará en 5 años.
- Proyecto B, que tiene una TIR del 10 % y se completará en 1 año.

*Solución:* El proyecto A, ya que su TIR es mayor. Aquí no se toma en cuenta el hecho de que el proyecto B tiene una duración más corta que el proyecto A, pues el tiempo ya está representado en los cálculos del TIR (tal como en el VPN, es el valor actual y no el valor futuro el que se utiliza para determinar la TIR).

## 4.5.2 Planificar para el valor

Después de justificar y confirmar el valor de un proyecto, el propietario del producto debe considerar las políticas de la organización, los procedimientos, las plantillas y las normas generales dictadas por el cuerpo de asesoramiento de Scrum (o el puesto similar o una junta organizacional del proyecto) en la planificación de un proyecto; y a la vez, maximizar la entrega basada en valor. La planificación para el valor es la justificación y confirmación del valor del proyecto. La responsabilidad de determinar cómo se crea valor recae en los socios (patrocinadores, clientes y/o usuarios), mientras que el equipo Scrum se concentra en lo que se habrá de desarrollar. Algunas de las herramientas comunes recomendadas por un cuerpo de asesoramiento de Scrum pudieran ser las siguientes:

### 1. Mapa de flujo de valor (Mapa de flujo de valor)

El mapa de flujo de valor utiliza diagramas de flujo para ilustrar el flujo de información necesaria para completar un proceso. Esta técnica pudiera utilizarse para racionalizar un proceso ayudando a determinar los elementos que no aportan valor.

### 2. Priorización basada en el valor que se entrega cliente (*Customer Value-based Prioritization*)

La priorización basada en el valor que se ofrece al cliente le da importancia primordial al cliente y se esfuerza primero en implementar las historias de usuario con más alto valor. Dichas historias de usuario de alto valor se identifican y se colocan en la parte superior de la lista priorizada de pendientes del producto.

Un equipo puede utilizar una variedad de esquemas de priorización para determinar las características de alto valor.

#### a. Esquemas simples

Los esquemas simples implican etiquetar elementos como prioridad “1”, “2”, “3” o “alta”, “media” y “baja”, y así sucesivamente. Aunque se trata de un método sencillo y directo, puede llegar a ser problemático, ya que a menudo hay una tendencia en etiquetar todo como prioridad “1” o “alta”. Incluso los métodos de priorización tales como “alta”, “media” y “baja” pueden encontrarse con dificultades similares.

#### b. Priorización MoSCoW

El esquema de priorización MoSCoW obtiene su nombre de la versión en inglés de las frases: “Debe tener” (*Must have*), “Debería tener” (*Should have*), “Podría tener” (*Could have*) y “No tendrá” (*Won't have*). Las etiquetas están en orden de prioridad decreciente con historias de usuario con características de “debería tener”, siendo aquellas sin las que el producto no tendrá valor, e historias de usuarios con características de “gustaría que tuviera” siendo aquellas que, a pesar de que sería bueno tener, no se es necesario incluir.

#### c. Dinero de Monopoly

El dinero de Monopoly es una técnica que consiste en darle al cliente dinero del juego Monopoly, o “dinero falso”, equivalente a la cantidad del presupuesto del proyecto, solicitando que lo distribuyan entre las historias de usuario que están a consideración. De esta forma, el cliente prioriza con base en lo que está dispuesto a pagar por cada historia de usuario.

#### d. Método de los 100 puntos

El método de los 100 puntos fue desarrollado en el 2003 por Dean Leffingwell y Don Widrig. Dicho método implica otorgar 100 puntos al cliente a fin de que los pueda utilizar para votar por las características que consideren más importantes.

#### e. Análisis de Kano

El análisis de Kano fue desarrollado en 1984 por Noriaki Kano y consiste en clasificar las características o requisitos en cuatro categorías con base en las preferencias del cliente:

1. *Calidad atractiva (Exciters/Delighters)*: Características que son nuevas o de gran valor para el cliente
2. *Calidad unidimensional (Satisfiers)*: Características que le ofrecen valor al cliente
3. *Calidad requerida (Dissatisfiers)*: Características que, si no están presentes, pudieran causar la insatisfacción del cliente respecto al producto, pero que no afectan el nivel de satisfacción si se cuenta con ellas.
4. *Calidad indiferente (Indifferent)*: Características que no afectarán al consumidor de ninguna manera y deben ser eliminadas.

La figura 4-4 muestra una ilustración del Análisis de Kano:

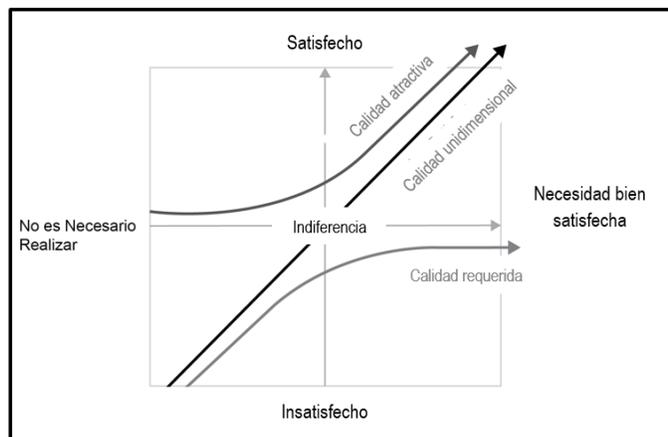


Figura 4-4: Análisis de Kano

Curiosamente, con el tiempo, las características por lo general se desplazan hacia abajo en la lista de clasificación; los clientes esperan ciertas características (por ejemplo, las cámaras en los teléfonos) y dichas funciones dejaron de ser una calidad atractiva para convertirse, eventualmente, en calidad requerida.

### 4.5.3 Clasificación relativa de priorización

La clasificación relativa de priorización es una simple enumeración de las historias de usuario en el orden de prioridad. Es un método eficaz para determinar las historias deseadas para cada iteración o lanzamiento del producto o servicio. El objetivo es crear una lista simple, con el único objetivo de dar prioridad a las características, en vez de distraerse con múltiples esquemas de priorización.

Esta sencilla lista también brinda una base para incorporar los cambios y los riesgos identificados cuando sea necesario. Cada cambio o riesgo identificado se puede incluir en la lista con base en su prioridad relativa a las demás historias de usuario en la lista. Típicamente, los nuevos cambios se incluirán a expensas de las características que se han asignado como con prioridad más baja.

Definir las Características Mínimas de Mercado (*Minimum Marketable Features - MMF*) es de suma importancia durante este proceso, de modo que la primera versión o iteración ocurra tan pronto como sea posible, lo cual lleva a un aumento en el retorno sobre la inversión. Normalmente, estas historias de usuario se clasifican como alta prioridad.

### 4.5.4 Mapeo de historias

El mapeo de historias, conocido en inglés como *Story Mapping*, es una técnica para proporcionar un esquema visual del producto y sus componentes clave. El mapeo de historias, formulado por primera vez por Jeff Patton (2005), se utiliza comúnmente para ilustrar la ruta del producto.

Los mapas de historia representan la secuencia de las iteraciones de desarrollo del producto y trazan las características que serán incluidas en el primer, segundo, tercero y subsecuentes lanzamientos.

## 4.6 Justificación continua de valor

La justificación continua de valor describe la evaluación con regularidad del valor del negocio para determinar si aún existe la justificación o viabilidad en la ejecución del proyecto. La evaluación frecuente de la inversión en el proyecto en relación con el valor del negocio que se está creando, califica la viabilidad de un proyecto. Los requisitos esperados del proyecto pueden cambiar con frecuencia, lo que puede afectar tanto la inversión del proyecto, como la creación de valor. Un aspecto clave de Scrum es su capacidad para adaptarse rápidamente al caos creado por un modelo de negocio que cambia rápidamente. En los proyectos donde los requerimientos de usuario son ambiguos y los cambios son frecuentes, Scrum ofrece considerables ventajas sobre otros modelos de desarrollo.

Monitorear la tasa de entrega de valor es un requisito importante para los proyectos Scrum. El seguimiento constante y los reportes sobre la creación de valor ayudan a evaluar el estado del proyecto y ofrecen información importante para el cliente y otros socios.

## 4.6.1 Análisis del valor ganado

Aunque se utilizan comúnmente, las herramientas tales como las gráficas de barras y diagramas de Gantt tienen sus limitaciones en el seguimiento y presentación de informes de progreso cuando se trata de rendimiento del proyecto. El análisis del valor ganado (AVG) se utiliza para tales efectos.

El análisis del valor ganado analiza el verdadero desempeño del proyecto en comparación al rendimiento planeado en un punto previsto. Para que las técnicas de rastreo sean eficaces, el plan de base del proyecto debe ser preciso. El AVG generalmente utiliza gráficas y demás elementos visuales (Curva S) como forma de representar la información sobre el estado del proyecto.

El análisis del valor ganado mide las variaciones actuales en la agenda del proyecto, así como el costo de funcionamiento y prevé del costo final con base en el rendimiento actual determinado. El AVG generalmente se lleva a cabo al final de cada sprint una vez que se han concluido las historias de usuario y la lista de pendientes del sprint.

La tabla 4-1 resume las fórmulas que se utilizan en el análisis del valor ganado.

Definición de término	Siglas (Inglés)	Fórmula
Valor planeado	PV	
Valor ganado	EV	
Costo actual	AC	
Presupuesto al finalizar	BAC	
Varianza del cronograma	SV	$EV - PV$
Varianza del costo	CV	$EV - AC$
Índice de desempeño del programa	SPI	$EV / PV$
Índice de desempeño de costo	CPI	$EV / AC$
Porcentaje completado	% Completado	$(EV / BAC) \times 100$
Estimado al completar 1. Las suposiciones de estimación no válidas 2. Las diferencias actuales son atípicas 3. Las variaciones actuales son típicas	EAC	1. $AC + ETC$ 2. $AC + BAC - EV$ 3. $BAC / CPI$
Estimado para completar	ETC	$EAC - AC$
Variación al completar	VAC	$BAC - EAC$

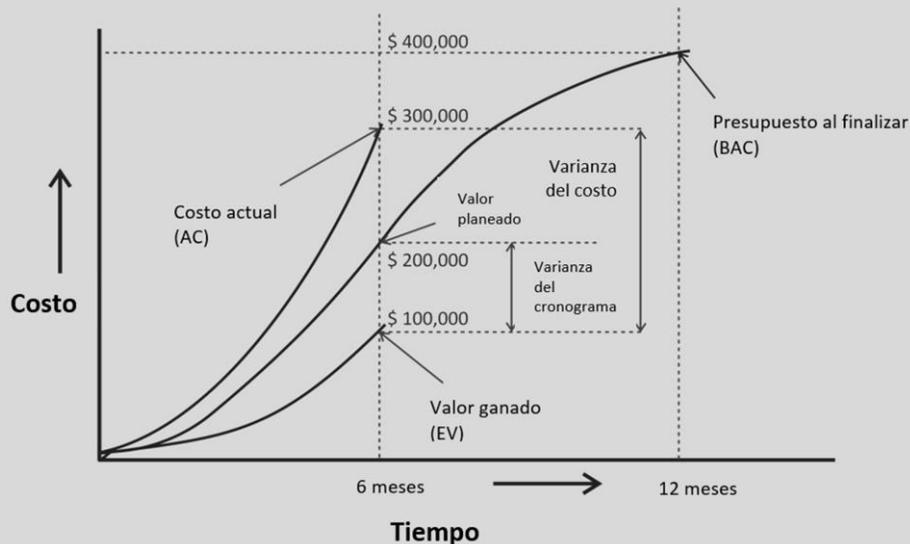
Tabla 4-1: Fórmulas del valor ganado

*Ejemplo:* Un sitio web con 4,000 páginas necesita ser desarrollado: asumimos que cada página web lleva el mismo tiempo en completar, y que cada página web es una historia de usuario única de igual prioridad en la lista priorizada de pendientes del producto. El costo estimado para completar el proyecto es de \$400,000 y el límite de tiempo para el proyecto es de 12 meses. Después de 6 meses, se han gastado \$300,000 y el trabajo realizado es 1,000 páginas web.

¿Qué información se nos ha proporcionado?

- Presupuesto al finalizar (BAC) = \$400,000 (Costo base del proyecto)
- Valor planeado (PV) = \$200,000 (ya que habíamos planeado completar 2,000 páginas web)
- Valor ganado (EV) = \$100,000 (el valor de 1,000 páginas web que se han completado)
- Costo actual (AC) = \$300,000 (lo que se ha gastado hasta el momento)

Curva S para los datos:



Fórmulas:

- Varianza del cronograma (SV) =  $EV - PV = \$100,000 - \$200,000 = -\$100,000$
- Varianza del costo (CV) =  $EV - AC = \$100,000 - \$300,000 = -\$200,000$ 
  - Las variaciones negativas en nuestro proyecto indican que estamos atrasados en el programa y por encima del presupuesto.
- Índice de desempeño del programa (SPI) =  $EV / PV = \$100,000 / \$200,000 = 0.5$ 
  - El SPI < 1 indica que el trabajo realizado hasta el momento es solo el 50 % de lo que habíamos planeado terminar en 6 meses.
- Índice de desempeño de costo (CPI) =  $EV / AC = \$100,000 / \$300,000 = 0.33$ 
  - El CPI < 1 indica que solo termina el 33 % del trabajo por la cantidad de dinero que se gasta.
- Porcentaje completado =  $EV / BAC \times 100 = \$100,000 / \$400,000 \times 100 = 25 \%$ 
  - Por lo tanto, hasta el momento se ha completado el 25 % del trabajo en el proyecto.

## 4.6.2 Diagrama de flujo acumulativo (DFA)

El diagrama de flujo acumulativo (DFA) es una herramienta útil para la elaboración de informes y para el seguimiento de los resultados del proyecto. Proporciona una representación sencilla y visual del progreso del proyecto en un punto de tiempo determinado. Se utiliza generalmente para brindar un estado de mayor nivel de la totalidad del proyecto y no para actualizaciones individuales diarias de sprints.

La figura 4-5 muestra un ejemplo de un DFA para un proyecto grande. Muestra cuántas historias de usuario están por crearse, las que están en proceso de creación y las que ya se han creado. A medida que cambian los requisitos de los clientes, se produce un cambio en las historias de usuario acumuladas que han de ser entregadas. Los puntos de cambio 1 y 2 son donde el propietario del producto ha eliminado las historias de usuario existentes en la lista priorizada de pendientes del producto del riesgo ajustado, mientras que los puntos de cambio 3 y 4 son donde el propietario del producto agregó nuevas historias de usuario en dicha lista.

Este tipo de diagrama puede ser una gran herramienta para la identificación de obstáculos y embotellamiento en los procesos. Por ejemplo, si el diagrama muestra una banda cada vez más estrecha, mientras la banda anterior con el tiempo está cada vez más amplia, pudiera haber un embotellamiento, por lo que pudieran ser necesarios cambios para aumentar la eficiencia y/o mejorar el desempeño del proyecto.

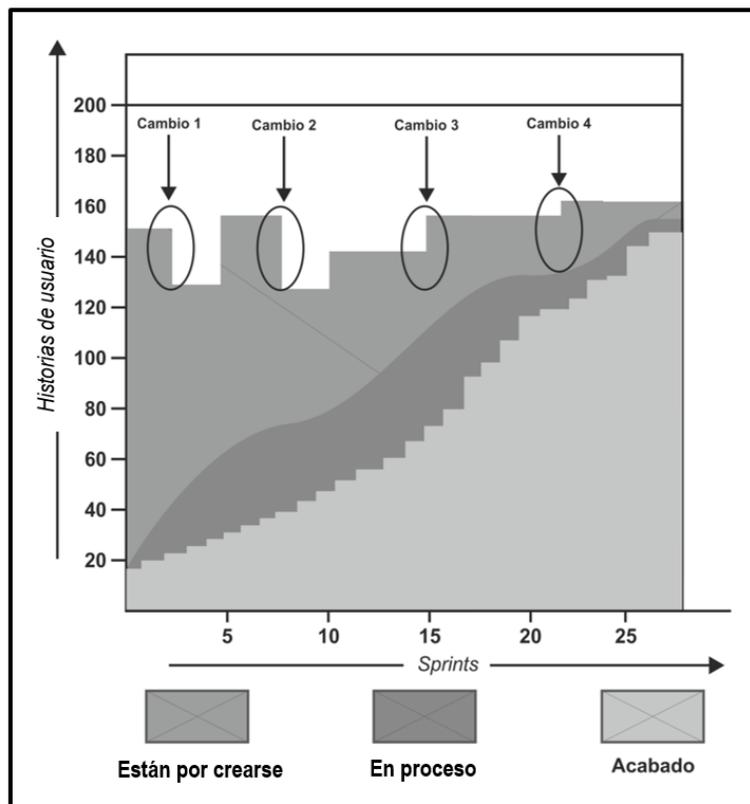


Figura 4-5: Ejemplo de diagrama de flujo acumulativo (DFA)

## 4.7 Confirmar la realización de beneficios

Durante todo un proyecto, es importante verificar si se están logrando los beneficios. Ya sea si los productos de un proyecto Scrum son tangibles o intangibles, se requieren técnicas adecuadas de verificación para confirmar que el equipo esté creando los entregables que lograrán los beneficios y el valor definido al inicio del proyecto.

### 4.7.1 Prototipos, simulaciones y demostraciones

La demostración de los prototipos (no confundirse con los “prototipos de usuario”, conocidos en inglés como *personas*) a los clientes y simular su funcionalidad, son técnicas comúnmente utilizadas para confirmar el valor.

Generalmente, después de usar las características o después de haberlas visto, los clientes pueden determinar con mayor claridad si las características son aptas y adecuadas para sus necesidades. Pueden darse cuenta de la necesidad de características adicionales, o pueden decidir modificar los requisitos de características previamente definidos. En el desarrollo de productos, esta experiencia del cliente ha llegado a ser conocida como IKIWISI, por sus siglas en inglés (*I'll know it when I see it*) o “Lo sabré cuando lo vea”.

Por medio de las demostraciones o del acceso a las iteraciones anticipadas, los clientes pueden también evaluar a qué grado el equipo ha sabido interpretar sus necesidades y cumplir con sus expectativas.

## 4.8 Resumen de responsabilidades

Rol	Responsabilidades
Cuerpo de asesoramiento de Scrum	<ul style="list-style-type: none"> <li>• Establece las directrices generales y los parámetros para evaluar el valor</li> <li>• Funge como consultor y brinda guías para los proyectos, programas y carteras, según sea necesario.</li> </ul>
Propietario del producto de la cartera	<ul style="list-style-type: none"> <li>• Garantiza la entrega de valor para las carteras</li> <li>• Crea la justificación del negocio para las cartetas</li> <li>• Proporciona una guía de valor para los programas en las carteras</li> <li>• Aprueba el justificación del negocio de los programas en una cartera</li> </ul>
Scrum master de la cartera	<ul style="list-style-type: none"> <li>• Garantiza que se logren los resultados deseados de la cartera</li> <li>• Lleva a cabo la justificación continua de valor para las carteras</li> </ul>
Propietario del producto del programa	<ul style="list-style-type: none"> <li>• Garantiza la entrega de valor para los programas</li> <li>• Crea la justificación del negocio para los programas</li> <li>• Proporciona una guía de valor para los programas en los programas</li> <li>• Aprueba el justificación del negocio de los programas en un programa</li> </ul>
Scrum Master del programa	<ul style="list-style-type: none"> <li>• Garantiza que los resultados esperados del programa se comuniquen y se entiendan</li> <li>• Lleva a cabo la justificación continua de valor para los programas</li> </ul>
Socio(s)	<ul style="list-style-type: none"> <li>• Ayuda a priorizar las historias de usuario y los requerimientos de la lista priorizada de pendientes del producto</li> <li>• Se comunica con el equipo Scrum y confirma la realización del valor al final de cada sprint, de cada lanzamiento y del proyecto.</li> </ul>
Propietario del producto	<ul style="list-style-type: none"> <li>• Garantiza la entrega de valor para los proyectos</li> <li>• Mantiene la justificación del negocio de los proyectos</li> <li>• Confirma y comunica los beneficios del proyecto a los socios</li> </ul>
Scrum Master	<ul style="list-style-type: none"> <li>• Garantiza que los resultados esperados del proyecto sean comunicados y comprendidos por el equipo Scrum</li> <li>• Lleva a cabo la justificación continua de valor para los proyectos</li> </ul>
Equipo Scrum	<ul style="list-style-type: none"> <li>• Asegura que los entregables del proyecto se completen de acuerdo con los criterios de aceptación acordados</li> <li>• Lleva a cabo la justificación continua de valor para los proyectos</li> </ul>

Tabla 4-2: Resumen de las responsabilidades pertinentes a la justificación del negocio

## 4.9 Scrum vs. Gestión tradicional de proyectos

Los proyectos tradicionales hacen énfasis en una amplia planificación y el apego al plan de proyecto creado por el administrador del proyecto. Por lo general, los cambios se administran a través de un sistema formal de gestión de cambio y el valor se crea al final del proyecto cuando se entrega el producto final.

En los proyectos Scrum, no se realiza una extensa planificación de largo plazo antes de la ejecución del proyecto. La planificación se realiza de manera iterativa antes de cada sprint. Esto permite una respuesta rápida y eficaz a los cambios, lo que se traduce en menores costos y en última instancia, aumenta la rentabilidad y el retorno sobre la inversión (RSI). Asimismo, la entrega basada en valor (sección 4.3) es un beneficio clave del marco de Scrum y brinda una mejor priorización considerablemente mejor, así como una más rápida realización del valor negocio. Debido a la naturaleza iterativa del desarrollo Scrum, siempre hay al menos una versión disponible del producto con las características mínimas de mercado (MMF, por sus siglas en inglés). Incluso si un proyecto se termina antes de completarse, por lo general hay algunos beneficios o valor generado antes de su interrupción.

## 5. CALIDAD

### 5.1 Introducción

El propósito de este capítulo es definir la calidad en lo que respecta a los proyectos y presentar el enfoque de Scrum para alcanzar los niveles de calidad requeridos.

La *calidad*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se les entregará a los socios;
- Proyectos de cualquier tamaño o complejidad.

El término “producto” en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo se divide en las siguientes secciones:

**5.2 Guía de roles**—Esta sección ofrece orientación sobre cuáles secciones son relevantes para cada uno de los roles centrales de Scrum: El propietario del producto, el Scrum Master y el equipo Scrum.

**5.3 Definición de calidad**—Esta sección brinda la definición de Scrum sobre calidad, con una clara distinción de alcance y describe la relación entre la calidad y el valor del negocio.

**5.4 Criterios de aceptación y la lista priorizada de pendientes del producto**—Esta sección hace hincapié en la importancia de los criterios de aceptación, la lista priorizada de pendientes del producto y la relación entre estos ambos. También se explica la definición de *terminado* en el contexto de Scrum.

**5.5 Gestión de calidad en Scrum**—Esta sección proporciona detalles sobre planificación de calidad, control de calidad y garantía de calidad en el contexto de Scrum.

**5.6 Resumen de responsabilidades**—Esta sección describe las responsabilidades pertinentes a la calidad para cada persona o rol en un proyecto.

**5.7 Scrum vs. Gestión tradicional de proyectos**—Esta sección destaca los beneficios de la gestión de calidad en el método Scrum en comparación a los modelos tradicionales de gestión de proyectos.

## 5.2 Guía de roles

1. Propietario del producto—Es importante que cualquier persona que asuma el rol de propietario del producto en proyectos Scrum, lea completamente este capítulo.
2. Scrum Master—El Scrum Master debe estar familiarizado con todo este capítulo, con un enfoque principal en las secciones 5.3, 5.4, 5.5.3 y 5.6.
3. Equipo Scrum—El equipo Scrum debe enfocarse principalmente en las secciones 5.3, 5.4 y 5.6.

## 5.3 Definición de calidad

Hay muchas maneras de definir la calidad.

En Scrum, la calidad se define como la capacidad que tiene un producto terminado o los entregables de cumplir con los criterios de aceptación y lograr el valor del negocio que espera el cliente.

Para asegurar que un proyecto cumpla con los requisitos de calidad, Scrum adopta un enfoque de mejora continua donde el equipo aprende de sus experiencias y de la participación de los socios para mantener constantemente actualizada la lista priorizada de pendientes del producto con cualquier cambio en los requerimientos. Dicha lista nunca está completa hasta el cierre o conclusión del proyecto. Cualquier cambio en los requisitos refleja los cambios en el entorno empresarial interno y externo y permite que el equipo funcione continuamente y se adapte para alcanzar esos requisitos. Scrum requiere que el trabajo se realice en incrementos durante los sprints, lo cual significa que los errores o defectos se detectan durante las pruebas de calidad repetitivas y no cuando el producto final o servicio está casi terminado. Asimismo, las tareas importantes relacionadas a la calidad (por ejemplo: el desarrollo, pruebas y documentación) se completan como parte del mismo sprint por el mismo equipo; esto asegura que la calidad sea inherente a cualquier entregable terminado creado como parte de un sprint. Por lo tanto, la mejora continua con sus repetitivas pruebas optimiza la probabilidad de alcanzar los niveles esperados de calidad en un proyecto Scrum. Las discusiones constantes entre el equipo principal de Scrum y los socios (incluyendo los clientes y usuarios) con incrementos reales del producto que se entregan al final de cada sprint, aseguran que se reduzca constantemente la brecha entre las expectativas de los clientes sobre el proyecto y los entregables producidos.

### 5.3.1 Calidad y alcance

Los requerimientos de alcance y calidad para un proyecto se determinan al tomarse en cuenta varios factores tales como los siguientes:

- La necesidad del negocio que habrá de cumplir el proyecto
- La capacidad y la buena disposición de la organización para cumplir con las necesidades del negocio

- Las necesidades futuras y actuales de la audiencia meta

El alcance de un proyecto es la suma total de todos los incrementos del producto, así como el trabajo necesario para el desarrollo del producto final. La calidad es la capacidad que tienen los entregables para cumplir con los requisitos de calidad del producto y satisfacer las necesidades del cliente. En Scrum, el alcance y la calidad del proyecto se capturan en la lista priorizada de pendientes del producto, mientras que el alcance de cada sprint se determina por la refinación de los amplios elementos de la lista priorizada de pendientes del producto (PBIs, por sus siglas en inglés) en un conjunto de pequeñas, pero detalladas historias de usuario que pueden ser planeadas, desarrolladas y verificadas en un sprint.

El propietario del producto da mantenimiento constante a la lista priorizada de pendientes del producto. El propietario del producto se asegura de que cualquier historia de usuario que se espera que realice el equipo Scrum en un sprint sea refinada antes del inicio del sprint. En general, los requerimientos más importantes en la resolución de los problemas del cliente o en el cumplimiento de sus necesidades, se consideran de alta prioridad, mientras que al resto se les da menor prioridad. Las historias de usuario de menor importancia se desarrollan en posteriores sprints o se pueden dejar de lado por completo según los requerimientos del cliente. Durante la ejecución del sprint, el propietario del producto, el cliente y el equipo Scrum pueden analizar la lista de características del producto para cumplir con las necesidades cambiantes de los clientes.

### 5.3.2 Calidad y valor del negocio

La calidad y el valor del negocio están estrechamente vinculados. Por lo tanto, es fundamental entender la calidad y el alcance de un proyecto a fin de trazar correctamente los resultados y beneficios que debe lograr tanto el proyecto como su producto para ofrecer valor empresarial. Para determinar el valor empresarial de un producto, es importante entender la necesidad del negocio que impulsa los requisitos del mismo. Por lo tanto, la necesidad del negocio determina cuál es el producto requerido, y este, a su vez, proporciona el valor empresarial esperado.

La calidad es una variable compleja. Aumentar el alcance sin incrementar el tiempo o los recursos, tiende a reducir la calidad. De igual forma, una reducción de tiempo o de recursos sin disminuir el alcance generalmente resulta en una disminución de la calidad. En Scrum se cree en el trabajo a “ritmo sostenible”, lo cual ayuda a mejorar la calidad durante cierto periodo.

El cuerpo de asesoramiento de Scrum puede definir los requisitos mínimos de calidad y estándares que se deben cumplir en todos los proyectos de la organización. Todos los equipos de Scrum en la empresa deben apegarse a dichos estándares.

## 5.4 Criterios de aceptación y lista priorizada de pendientes del producto

La lista priorizada de pendientes del producto es un solo documento de requisitos que define el alcance del proyecto, proporcionando una lista de prioridades de las características del producto o servicio a ser entregado por el proyecto. Las características necesarias se describen en forma de historias de usuario. Dichas historias son requisitos específicos señalados por varios socios que se relacionan con el producto o servicio propuesto. Cada historia de usuario contará con sus criterios de aceptación de historia de usuario (también conocidos como “criterios de aceptación”), que son los componentes objetivos mediante los cuales se juzga la funcionalidad de una historia de usuario. Los criterios de aceptación los desarrolla el propietario del producto según su experiencia en los requerimientos del cliente. El propietario del producto después comunica las historias de usuario que están en la lista priorizada de pendientes del producto a los miembros del equipo Scrum, buscando un común acuerdo. Los criterios de aceptación deben delinear explícitamente las condiciones que deben satisfacer las historias de usuario. Los criterios de aceptación claramente definidos son de suma importancia la entrega eficaz y oportuna de la funcionalidad definida en las historias de usuario, lo cual, en última instancia, determinan el éxito del proyecto.

Al final de cada sprint, el propietario del producto utiliza estos criterios para verificar los entregables completados y puede aceptar o rechazar entregables individuales, así como sus respectivas historias de usuario. Si los entregables son aceptados por el propietario del producto, la historia de usuario se considera entonces como terminada. Es importante contar con una clara definición de “terminado”, ya que ayuda a poner en claro los requerimientos y permite que el equipo se apegue a las normas de calidad. Ayuda también a que el equipo piense desde la perspectiva del usuario al momento de trabajar con historias de usuario.

Las historias de usuario que corresponden a los entregables rechazados se agregan de nuevo a la lista priorizada de pendientes del producto durante el proceso de dar mantenimiento a dicha lista para que puedan ser completados en futuros sprints. El rechazo de unos cuantos entregables individuales y sus correspondientes historias de usuario no significa el rechazo del producto final o del incremento del producto. El producto o el incremento del producto pudiera ser potencialmente enviable incluso si se rechazan algunas historias de usuario.

La figura 5-1 ilustra el concepto de los criterios de aceptación y el flujo de incremento del producto.

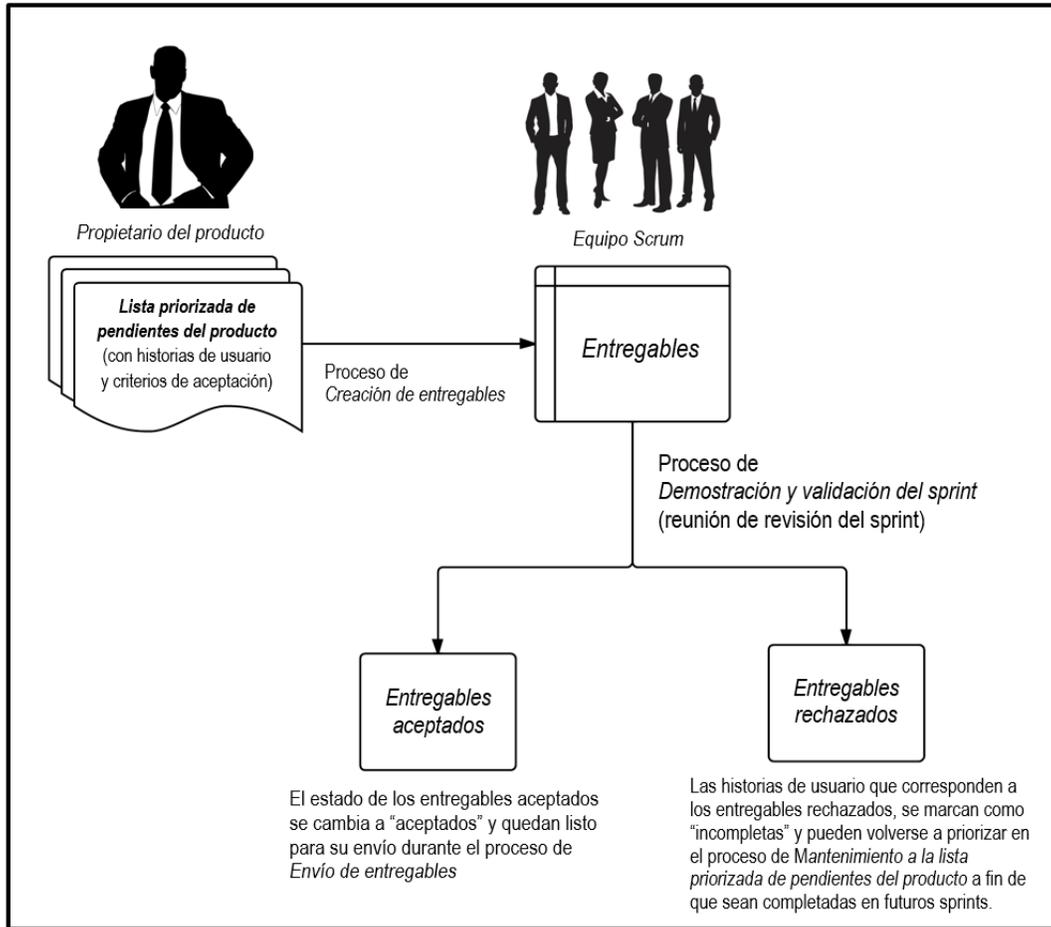


Figura 5-1: Diagrama de flujo del incremento del proyecto

### 5.4.1 Redacción de los criterios de aceptación

Los criterios de aceptación son únicos en cada historia de usuario y no son un sustituto de la lista de requerimientos.

*Ejemplo:*

**Personaje:** Janine es una profesionalista de 36 años; está casada y tiene tres hijos. Es una mujer ocupada y exitosa que equilibra su vida profesional y personal. Se siente cómoda con la tecnología y le gusta adoptar productos y servicios innovadores. Siempre está conectada al Internet a través de múltiples dispositivos y regularmente hace compras en portales de comercio electrónico.

**Historia de usuario:** “Como compradora en línea, debo contar con la posibilidad de ahorrar y ver mi orden desde cualquiera de mis dispositivos para poder completar el proceso de orden cuando mejor me convenga”.

**Criterios de aceptación:**

- Todas las órdenes en progreso deben guardarse como pendiente (*draft*) cada 5 segundos en la cuenta del usuario que haya iniciado una sesión.
- Las nuevas órdenes pendientes deben mostrarse como notificaciones en cualquier dispositivo de donde el usuario inicie sesión.

5

Es importante que el propietario del producto tome nota que las historias de usuario que cumplan con la mayoría, aunque no con todos los criterios de aceptación, no pueden aceptarse como terminadas. Los proyectos Scrum operan en sprints con un bloque de tiempo asignado, con una lista de pendientes asignada para cada sprint. Generalmente, la última parte del trabajo pudiera ser la parte más complicada de la historia de usuario y pudiera tardar más de lo esperado. Si se les dio crédito a historias de usuario incompletas como si estuvieran terminadas —y si se llevaron al siguiente sprint—, el progreso de posteriores sprints pudiera verse interrumpido. Por lo tanto, el estado de “terminado” es a blanco y negro. Una historia de usuario puede ser, ya sea “terminada” o “no terminada”.

#### 5.4.2 Criterios mínimos de aceptación

Una unidad empresarial de más alto nivel puede anunciar un criterio de aceptación mínimo y obligatorio, lo cual después forma parte de los criterios de aceptación para cualquier historia de usuario en esta unidad empresarial. Cualquier funcionalidad definida por la unidad empresarial debe satisfacer dichos criterios mínimos de aceptación si es que se busca la aceptación del respectivo propietario del producto. La introducción a estos criterios de aceptación puede llevar a una serie en cascada de criterios de aceptación para la cartera, el programa y el proyecto (véase la figura 5-2). Los estándares generales de calidad, las directrices y las plantillas para toda la cartera, los establece el propietario del producto de la cartera, mientras que los criterios mínimos de aceptación los establece el propietario del producto del programa. De tal forma, los criterios de aceptación de una historia de usuario en un proyecto habrán de incluir implícitamente todos los criterios mínimos de aceptación de los niveles más elevados, según corresponda.

Propietario del producto de la cartera	<ul style="list-style-type: none"> <li>• Establece los criterios mínimos de aceptación para toda la cartera</li> <li>• Revisa los entregables de la cartera</li> </ul>
Propietario del producto del programa	<ul style="list-style-type: none"> <li>• Establece los criterios mínimos de aceptación de todo el programa, el cual incluye los criterios de aceptación de la cartera</li> <li>• Revisa los entregables del programa</li> </ul>
Propietario del producto	<ul style="list-style-type: none"> <li>• Establece los criterios mínimos de aceptación para el proyecto, el cual incluye los criterios de aceptación del programa</li> <li>• Revisa los entregables del proyecto</li> </ul>

Figura 5-2: Criterios de aceptación en cascada

Una vez que se definen los criterios de aceptación, estos se pueden incluirse en la documentación del cuerpo de asesoramiento de Scrum y enviarse a los equipos Scrum según sea necesario.

### 5.4.3 Definición de terminado

Hay una diferencia clave entre los “criterios de terminado” y los “criterios de aceptación”. Mientras que los criterios de aceptación son únicos para en las historias de usuario individuales, los criterios de terminado son una serie de reglas aplicables a todas las historias de usuario en un determinado sprint. Los criterios de terminado pueden incluir cualquiera de los siguientes:

- Fueron revisados por otros miembros del equipo;
- Completaron la prueba de unidad de la historia de usuario;
- Conclusión de las pruebas de garantía de calidad;
- Conclusión de toda la documentación relacionada a la historia de usuario;
- Se corrigieron todos los problemas;
- Demostración satisfactoria a los socios y/o representantes empresariales.

Al igual que con los criterios de aceptación, se deben cumplir todas las condiciones de los criterios de terminado para que la historia de usuario se considere terminada.

El equipo Scrum debe utilizar una lista de verificación de los criterios de terminado generales para garantizar que una tarea está terminada y de que el resultado cumpla con la con la definición de terminado (DoD, por sus siglas en inglés). Es importante contar con una clara definición de terminado, ya que ayuda a eliminar la ambigüedad y permite al equipo apegarse a las normas de calidad requeridas. La definición de terminado típicamente la determina y la documenta el cuerpo de asesoramiento de Scrum.

Los registros y datos necesarios para cumplir con los requisitos de documentación del proyecto se pueden generar a medida que el equipo procede a través de sprints y publicaciones.

La inclusión de actividades tales como reuniones de revisión y la redacción de documentos de diseño pueden ayudar a asegurar el cumplimiento de las normas de calidad internas y externas. Los principios básicos de Scrum, tales como iteraciones cortas, la construcción gradual, la participación del cliente, la adaptación a las nuevas necesidades y el constante ajuste de tiempo y costo en el proyecto seguirán siendo aplicables.

#### **5.4.4 Aceptación o rechazo de elementos de la lista priorizada de pendientes del producto**

Hacia el final de cualquier iteración, los socios y la unidad empresarial correspondiente participan en una reunión de revisión del sprint en la cual se demuestran los incrementos del producto al propietario del producto, al patrocinador, al cliente y al usuario. Aunque se recoge la opinión de todos los usuarios, únicamente el propietario del producto tiene la autoridad para aceptar o rechazar como terminada una historia de usuario en particular, según los criterios de aceptación que fueron acordados. Por lo tanto, el rol de los criterios de aceptación en conservar la calidad es de suma importante y debe entenderse claramente por el equipo. Es responsabilidad del Scrum Master garantizar que los criterios de aceptación de una historia de usuario no sean modificados por el propietario del producto a la mitad de un sprint. Las historias de usuario parcialmente completadas son rechazadas como “no terminadas” y se devuelven a la lista priorizada de pendientes del producto.

### **5.5 Gestión de calidad en Scrum**

El cliente es el socio más importante en cualquier proyecto; por lo tanto, es importante entender sus necesidades y requerimientos. La voz del cliente puede describirse como los requerimientos explícitos e implícitos del cliente, los cuales deben entenderse antes del diseño de un producto o servicio. Generalmente, en un ambiente de Scrum, el propietario del producto se enfoca en los requerimientos y objetivos del negocio, los cuales, juntos, representan la voz del cliente. El propietario del producto se puede beneficiar considerablemente de la guía disponible del cuerpo de asesoramiento de Scrum (ya sea a través de documentos o estándares de calidad o de expertos en calidad). Estos especialistas deben trabajar con el propietario del producto y con el cliente a fin de garantizar el nivel apropiado de detalle e información en las historias de usuario, ya que estas son la base del éxito de cualquier proyecto Scrum.

Cabe señalar que los socios externos no participan directamente al nivel del equipo Scrum y, en cambio, interactúan principalmente con el propietario del producto. Para cualquier proyecto Scrum, el cliente puede ser cualquiera de los siguientes:

- Interno (dentro de la misma organización)
- Externo (fuera de la organización)

En Scrum, el control de calidad les permite a los clientes conocer cualquier problema en el proyecto en forma anticipada, ayudándoles a reconocer si el proyecto habrá o no de funcionarles. En Scrum, la calidad gira en torno a la satisfacción del cliente y de un producto funcional y no necesariamente en cumplir con

parámetros arbitrarios. Dicha distinción resulta muy importante desde el punto de vista del cliente, ya que son quienes invierten tiempo y dinero en el proyecto.

En Scrum, la gestión de calidad se facilita mediante tres actividades interrelacionadas:

1. Planificación de calidad
2. Control de calidad
3. Garantía de calidad

## 5.5.1 Planificación de calidad

Uno de los principios rectores de Scrum es desarrollar primero la funcionalidad de la más alta prioridad para cliente. Las características de menor importancia se desarrollan en posteriores sprints o pueden dejarse de lado completamente según los requerimientos del cliente. Este enfoque brinda al equipo Scrum el tiempo para centrarse en la calidad de la funcionalidad esencial. Un beneficio clave de la planificación de calidad es la reducción de la deuda técnica. La deuda técnica, conocida también como deuda de diseño o deuda de código, es el trabajo al que los equipos dan menor prioridad; el trabajo que omiten o que no terminan a medida que trabajan en la creación de los principales entregables asociados al producto del proyecto. La deuda técnica se acumula y se debe saldar a futuro.

Algunas de las causas de la deuda técnica pueden ser:

- Rápida solución y elaboración de entregables que no cumplan con los estándares de calidad, seguridad, metas arquitectónicas a largo plazo, etc.
- Evaluación inadecuada o incompleta;
- Documentación inadecuada o incompleta;
- Falta de coordinación entre los distintos miembros del equipo, o si hay diferentes equipos de Scrum que empiezan a trabajar en forma aislada con menor enfoque en la integración final de los componentes requeridos para realizar un proyecto o programa exitoso.
- Intercambio deficiente del conocimiento empresarial y de procesos entre los socios y equipos del proyecto.
- Demasiado énfasis en los objetivos del proyecto a corto plazo en vez de objetivos a largo plazo en la empresa. Esta supervisión puede resultar en una baja calidad de los entregables funcionales que pudiera incurrir en considerables costos de mantenimiento y actualización.

En los proyectos Scrum, cualquier deuda técnica no debe llevarse más allá de un sprint, ya que debe haber criterios de aceptación y de terminado debidamente definidos. La funcionalidad debe satisfacer dichos criterios para considerarlos terminados. Conforme se da mantenimiento a la lista priorizada de pendientes del producto y se da prioridad a las historias de usuario, el equipo elabora regularmente entregables funcionales, previniendo con ello la acumulación de una deuda técnica considerable. El cuerpo de asesoramiento de Scrum pudiera también incluir documentación y definición de los procesos que ayudan a disminuir dicha deuda.

Para conservar una cantidad mínima de deuda técnica, es importante definir el producto requerido en un Sprint y del proyecto, así como los criterios de terminado, cualquier método de desarrollo a seguir y las responsabilidades clave de los miembros del equipo Scrum respecto a la calidad. Definir los criterios de calidad es una parte importante de la planificación de calidad y permite que se lleve a cabo un control eficaz de la misma durante el proyecto.

La deuda técnica es un gran reto con algunas técnicas de gestión tradicional de proyectos donde el desarrollo, la evaluación, la documentación, etc., se realizan en forma subsecuente y por lo general por distintas personas, donde ninguna es responsable de cualquier entregable funcional en particular. Como resultado, la deuda técnica se acumula, llevando a un mantenimiento considerablemente más elevado, integración y costos de lanzamiento del producto en las etapas finales su lanzamiento. Asimismo, el costo de los cambios es muy elevado en tales circunstancias, ya que los problemas surgen en etapas posteriores del proyecto. El marco de Scrum evita los problemas relacionadas a la deuda técnica, garantizando que se definan los entregables terminados con criterios de aceptación como parte de la lista de pendientes del sprint y que las tareas clave, incluyendo el desarrollo, la evaluación y la documentación se lleven a cabo como parte del mismo sprint y por el mismo equipo Scrum.

#### **5.5.1.1 Integración continua y ritmo sostenible**

Mantener un ritmo sostenible es uno de los principios más importantes de Scrum. El ritmo sostenible se traduce en una mayor satisfacción del empleado, en estabilidad y una mayor precisión en la estimación; todo ello conlleva, en última instancia, a un aumento en la satisfacción del cliente. Para desarrollar un producto verdaderamente de alta calidad y conservar un sano ambiente laboral, es importante realizar periódicamente actividades de integración, en vez de retrasar el trabajo de integración hasta el final en tales circunstancias. Para brindar valor en intervalos frecuentes, el equipo debe continuamente desarrollar, evaluar e integrar las funcionalidades de cada elemento en la lista priorizada de pendientes del producto en cada sprint con el uso de técnicas, tales como la integración continua y la evaluación automática del producto. También es importante, desde la perspectiva del equipo, garantizar que el esfuerzo realizado en el actual sprint sea similar al esfuerzo realizado en el sprint anterior a fin de sostener un ritmo constante durante los sprints en el proyecto. Esto ayuda al equipo a evitar fases de intensos periodos de trabajo, garantizando que puedan siempre presentar el esfuerzo requerido para lograr el trabajo que debe realizarse.

#### **5.5.2 Control de calidad y garantía de calidad**

El control de calidad es la ejecución de las actividades de calidad planeadas por el equipo Scrum en el proceso de creación de entregables que la potencialidad de enviarse. Incluye también el aprendizaje de cada serie de actividades realizadas a fin de lograr una mejora continua. Dentro del equipo inter-funcional, es importante contar con las habilidades necesarias para llevar a cabo actividades de control de calidad. Durante la reunión de retrospectiva del sprint, los miembros del equipo analizan las lecciones aprendidas. Dichas lecciones funcionan como entradas en la mejora continua y contribuyen al mejoramiento del constante control de calidad.

Se requiere de la calidad no solo en los productos, sino también en los procesos. La garantía de calidad es el proceso de evaluación y los estándares que rigen la gestión de calidad en un proyecto a fin de garantizar que continúen siendo relevantes. Las actividades relacionadas a la garantía de calidad se llevan a cabo como parte del trabajo. De hecho, la garantía de calidad es un factor considerable en la definición de terminado. El entregable no se considera completo si no se ha realizado una garantía adecuada de calidad. Generalmente, la garantía de calidad se demuestra durante la reunión de revisión del sprint.

El propietario del producto de los proyectos respectivos, programas y carteras, puede monitorear y evaluar las actividades de garantía de calidad para asegurarse de que cada equipo siga de acuerdo y cumplir con los estándares de calidad que se han establecido. La garantía de calidad de un extremo al otro puede abordarse durante la evaluación final del producto, de un lanzamiento de un sprint. Se puede realizar una comparación de la cantidad de problemas que se encontraron en relación a la cantidad de historias de usuario completadas. Los componentes del producto que tienen defectos se pueden incorporar como elementos de la lista priorizada de pendientes del producto, mismas que se pueden abordar ya sea por el equipo o por una persona durante ciertos periodos durante el sprint, dependiendo del número de defectos. En ocasiones, el cuerpo de asesoramiento de Scrum puede definir los procesos y documentos que pueden remitirse a los equipos de Scrum al momento de hacer sus proyectos a fin de asegurarse de que se les dé seguimiento a las normas uniformes de calidad en todos los proyectos al interior de la empresa.

### 5.5.3 Ciclo de planificar, hacer, verificar y actuar (Ciclo PDCA, por sus siglas en inglés)

El ciclo de planificar, hacer, verificar y actuar (PDCA, por sus siglas en inglés), conocido también como Ciclo Deming o Shewhart, fue desarrollado por el doctor W. Edwards Deming, considerado como el padre del control de calidad moderno, y por el doctor Walter A. Shewhart. Estas son algunos de los puntos más importantes de la filosofía de Deming:

- Las directrices de la gerencia son las que definen la calidad. Cuando la gerencia puede ofrecer un ambiente conductivo y motivar a sus empleados para que mejore la calidad en forma continua, cada empleado podrá contribuir hacia una calidad superior en el producto. La “Teoría del conocimiento profundo” de Deming aboga por lo que debe hacer la gerencia a fin de crear un ambiente en el cual cada empleado puede realizar contribuciones considerables para mejorar la calidad.

Deming modificó después el nombre de “planificar, hacer, verificar y actuar” a “planificar, hacer, estudiar y actuar” (PDSA, por sus siglas en inglés), ya que consideraba que el término “estudiar” hacía énfasis en el análisis en vez de simplemente la inspección, como lo implica el término “verificar”.

Tanto Scrum como el Ciclo Deming/Shewhart/PDCA son métodos iterativos enfocados en la mejora continua. La figura 5-3 ilustra las etapas del ciclo PDCA y su correlación con los diversos procesos de Scrum.

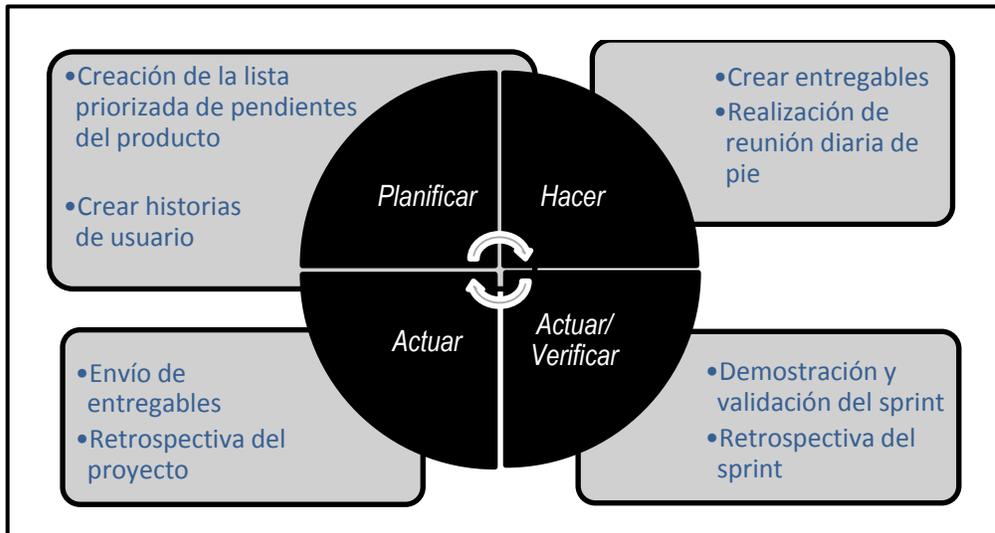


Figura 5-3: Ciclo PDCA en Scrum

## 5.6 Resumen de responsabilidades

Rol	Responsabilidades
Cuerpo de asesoramiento de Scrum	<ul style="list-style-type: none"> <li>• Proporciona la definición de “terminado”.</li> <li>• Proporciona un marco y una guía para el desarrollo de los criterios de aceptación.</li> <li>• Define la gama de herramientas que puede utilizar el equipo Scrum para desarrollar y verificar el producto.</li> </ul>
Propietario del producto de la cartera	<ul style="list-style-type: none"> <li>• Establece criterios mínimos de aceptación para toda la cartera.</li> <li>• Revisa los entregables de la cartera.</li> </ul>
Scrum Master de la cartera	<ul style="list-style-type: none"> <li>• Se asegura de que se mantenga un ritmo sostenible cuyo enfoque sea la calidad de las características y no estrictamente en la velocidad.</li> </ul>
Propietario del producto del programa	<ul style="list-style-type: none"> <li>• Establece criterios mínimos de aceptación para todo el programa.</li> <li>• Revisa los entregables del programa.</li> </ul>
Scrum Master del programa	<ul style="list-style-type: none"> <li>• Se asegura de que se mantenga un ritmo sostenible cuyo enfoque sea la calidad de las características y no estrictamente en la velocidad.</li> </ul>
Socio(s)	<ul style="list-style-type: none"> <li>• Revisa y acepta los entregables y el producto final.</li> </ul>

Propietario del producto	<ul style="list-style-type: none"> <li>• Define los requerimientos del negocio para los productos y define claramente los requerimientos en la lista priorizada de pendientes del producto.</li> <li>• Evalúa la viabilidad y se asegura de que los productos entregables cumplan con los requisitos de calidad.</li> <li>• Establece los criterios mínimos de aceptación para todo el proyecto, incluyendo los criterios de aceptación para el programa respectivo.</li> </ul>
Scrum Master	<ul style="list-style-type: none"> <li>• Facilita una mentalidad de “primero el equipo”, cuando de calidad se trata.</li> <li>• Elimina obstáculos en el entorno que pudieran afectar la calidad de los entregables y de los procesos.</li> <li>• Se asegura de que se mantenga un ritmo sostenible cuyo enfoque sea la calidad de las características y no estrictamente en la velocidad.</li> <li>• Se asegura de que todos los miembros del equipo, incluyendo el propietario del producto, sigan correctamente los procesos de Scrum.</li> </ul>
Equipo Scrum	<ul style="list-style-type: none"> <li>• Desarrolla y mantiene todos los entregables durante los sprints hasta que pasen a manos de los usuarios finales.</li> <li>• Pone en práctica y fomenta la buena comunicación a fin de que los requerimientos queden clarificados y completamente entendidos.</li> <li>• Comparte conocimiento para garantizar que los miembros del equipo se familiaricen con la serie de características y con ello se beneficien de la experiencia de otros.</li> <li>• Hace cambios adecuados con rapidez a los entregables.</li> </ul>

**Tabla 5-1: Resumen de las responsabilidades pertinentes a la calidad**

## 5.7 Scrum vs. Gestión tradicional de proyectos

Aunque hay similitudes entre Scrum y los métodos tradicionales de gestión de proyectos en relación a la definición de “calidad” (la capacidad con la que cuenta el producto para cumplir con los criterios de aceptación acordados y lograr el valor empresarial que espera el cliente), existen diferencias en términos de cómo los métodos abordan la implementación y logro de los niveles exigidos de calidad.

En los métodos tradicionales de gestión de proyectos, los usuarios aclaran sus expectativas; el gerente del proyecto define dichas expectativas en términos cuantificables y obtiene el acuerdo de los usuarios. Después de una planificación detallada, el equipo del proyecto desarrolla el producto durante un período acordado. Si hay necesidad de cambiar alguno de los criterios acordados, los cambios se pueden realizar sólo a través de un sistema formal de gestión de cambio, en el cual se estima el impacto de los cambios y el gerente de proyecto consigue la aprobación de todos los socios relevantes.

En Scrum, sin embargo, el propietario del producto colabora con el equipo Scrum y define los criterios de aceptación para las historias de usuario en relación con el producto que se debe entregar. El equipo Scrum, después desarrolla el producto a partir de una serie de iteraciones cortas denominadas sprints. El propietario del producto puede realizar cambios en los requisitos para mantenerse al ritmo de las necesidades del usuario y estos cambios pueden ser abordados por el equipo Scrum, ya sea al concluir el actual sprint, o al incluir los requisitos ajustados en el próximo sprint, ya que cada uno es de muy corta duración (de una a seis semanas).

Una de las principales ventajas de Scrum es el énfasis en la creación de entregables potencialmente listos para ser enviados al final de cada ciclo de sprint, en vez de esperar al final de todo el proyecto. Así, el propietario del producto y los clientes constantemente inspeccionan, aprueban y aceptan entregables después de cada sprint. Incluso, si un proyecto Scrum se termina antes de tiempo, siempre existe algo de valor que fue creado antes de la terminación como resultado de los entregables creados en sprints individuales.

## 6. Cambio

### 6.1 Introducción

Cada proyecto, independientemente de su método o marco, está expuesto a cambios. Es imperativo que los miembros del equipo del proyecto entiendan que los procesos de desarrollo de Scrum están diseñados para aceptar el cambio. Las organizaciones deben tratar de maximizar los beneficios que se derivan de los cambios, y disminuir los impactos negativos a través de los procesos de gestión de cambio diligente según los principios de Scrum.

El *cambio*, tal como se define en *Una guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria
- Productos, servicios o cualquier otro resultado que se les entregará a los socios
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la Guía SBOK™ puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria—desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo está dividido en las siguientes secciones:

**6.2 Guía de los roles**—En esta sección se proporciona orientación sobre qué secciones son relevantes para cada uno de los roles principales de Scrum: el propietario del producto, el Scrum Master, y el equipo Scrum.

**6.3 Descripción**—Esta sección define el concepto de cambio, específicamente dentro del contexto de los procesos de Scrum. También ilustra la forma en la que las solicitudes de cambios se manejan en los procesos de Scrum.

**6.4 Cambio en Scrum**—En esta sección se detalla la importancia de gestionar con eficacia el cambio en un proyecto Scrum. También se habla de cómo la flexibilidad y la estabilidad se pueden lograr a través del manejo adecuado de las solicitudes de cambio que surgen a lo largo de un proyecto.

**6.5 Integración del cambio**—Esta sección detalla cómo las solicitudes de cambio se evalúan y aprueban (o se rechazan) en la aplicación del marco de Scrum.

**6.6 Cambio a los programas y carteras**—En esta sección se describe el impacto de los cambios en los programas y portafolios.

**6.7 Resumen de las responsabilidades**—Esta sección define las responsabilidades de los miembros de un equipo de proyecto al gestionar cambios.

**6.8 Scrum vs. Proyecto tradicional de gestión**—Esta sección discute los beneficios de gestionar cambios utilizando Scrum en comparación a los modelos tradicionales de gestión de proyectos.

## 6.2 Guía de los roles

4. Propietario del producto—La responsabilidad de iniciar el cambio en un proyecto recae principalmente en el propietario del producto, por lo tanto, todo este capítulo es aplicable a este rol.
5. Scrum Master—El Scrum Master también debe familiarizarse con todo este capítulo, con un enfoque principal en las secciones 6.3, 6.4, 6.5, y 6.7.
6. Equipo Scrum—El Equipo Scrum debe centrarse principalmente en las secciones 6.3, 6.4.2, 6.5 y 6.7.

## 6.3 Descripción

El cambio es inevitable en todos los proyectos. En el mundo excesivamente competitivo de hoy, donde la tecnología, las condiciones del mercado y los patrones de negocio están cambiando de forma continua, el cambio es la única constante.

Un principio fundamental de Scrum es su reconocimiento de que, a) los *socios* (por ejemplo, clientes usuarios y patrocinadores) cambian de opinión acerca de lo que quieren y necesitan en todo el proyecto (a veces a esto se le llama “requerimientos volátiles”), y b) que es muy difícil, si no imposible, para los socios definir todos los requisitos durante el inicio de un proyecto.

Los proyectos de desarrollo de Scrum son receptivos al cambio mediante el uso de pequeños ciclos de desarrollo que incorporan la retroalimentación del cliente de los entregables del proyecto después de cada sprint. Esto permite que el cliente interactúe regularmente con los miembros del equipo Scrum, vea los incrementos del producto a medida que estén listos, y cambie los requisitos en las etapas tempranas en el ciclo de desarrollo. De esa forma, los equipos de gestión de carteras o de programas pueden responder a las solicitudes de cambio pertenecientes a los proyectos Scrum aplicables a su nivel.

Scrum personifica un principio primordial del Manifiesto Ágil (Fowler y Highsmith, 2001): “Responder al cambio, en vez de seguir un plan”. La práctica de Scrum se basa en la aceptación del cambio y de convertirlo en una ventaja competitiva. Por lo tanto, es más importante ser flexible que seguir un plan estricto y predefinido. Esto significa que es esencial abordar la gestión de proyectos de una manera adaptativa que permita el cambio durante el desarrollo del producto o de los ciclos rápidos de servicio.

Ser adaptable al cambio es una ventaja clave del marco de Scrum. Aunque Scrum funciona bien en todo tipo de proyectos e industrias, puede ser muy eficaz cuando el producto o los requisitos del proyecto no se identificaron o no están bien definidos desde el principio. Scrum también es eficaz cuando el mercado del producto es volátil, y/o cuando la atención se centra en hacer que el equipo sea lo suficientemente flexible para incorporar los requisitos cambiantes. Scrum es especialmente útil para proyectos complejos con mucha incertidumbre. La planificación y previsión a largo plazo suele ser ineficaz para este tipo de proyectos e implican grandes cantidades de riesgo. Scrum lleva al equipo a obtener resultados valiosos de negocio utilizando la *transparencia, la inspección y adaptación*.

### 6.3.1 Solicitudes aprobadas y no aprobadas de cambio

La solicitud para hacer cambios se presenta por lo general como solicitudes de cambio. Las solicitudes de cambio no son aprobadas hasta que se obtiene una aprobación formal. El cuerpo de asesoramiento de Scrum por lo general define el proceso de decisión y gestión de cambios en la organización. En ausencia de un proceso formal, se recomienda que los pequeños cambios que no tienen un impacto significativo en el proyecto se aprueben directamente por el propietario del producto. La tolerancia de estos pequeños cambios podría definirse a nivel organizacional o por el patrocinador de un proyecto en particular. En la mayoría de los proyectos, el 90% de las solicitudes de cambio podrían clasificarse como pequeños cambios que deben aprobarse por el propietario del producto. Así que el propietario del producto juega un papel muy importante en la gestión de los cambios en un proyecto Scrum.

Los cambios que están más allá del nivel de tolerancia del propietario del producto pueden necesitar la aprobación de los socios que trabajan con él.

A veces, si un cambio que se ha solicitado puede tener un impacto sustancial en el proyecto u organización, la autorización de la alta gerencia (por ejemplo, el patrocinador ejecutivo, el propietario del producto de la cartera, el propietario del producto del programa, o el jefe propietario del producto), puede ser necesaria.

Las solicitudes de cambio para el proyecto se discuten y aprueban durante los siguientes procesos: desarrollo de épica(s), creación de la lista priorizada de pendientes del producto, y mantenimiento de la lista priorizada de pendientes del producto. Posteriormente, las solicitudes de cambio aprobadas se priorizan junto con otros requisitos del producto y sus respectivas historias de usuario, y luego se incorporan en la lista priorizada de pendientes del producto.

La figura 6-1 resume el proceso de aprobación de cambio, y la figura 6-2 explica cómo se actualiza la lista priorizada de pendientes del producto con los cambios aprobados.

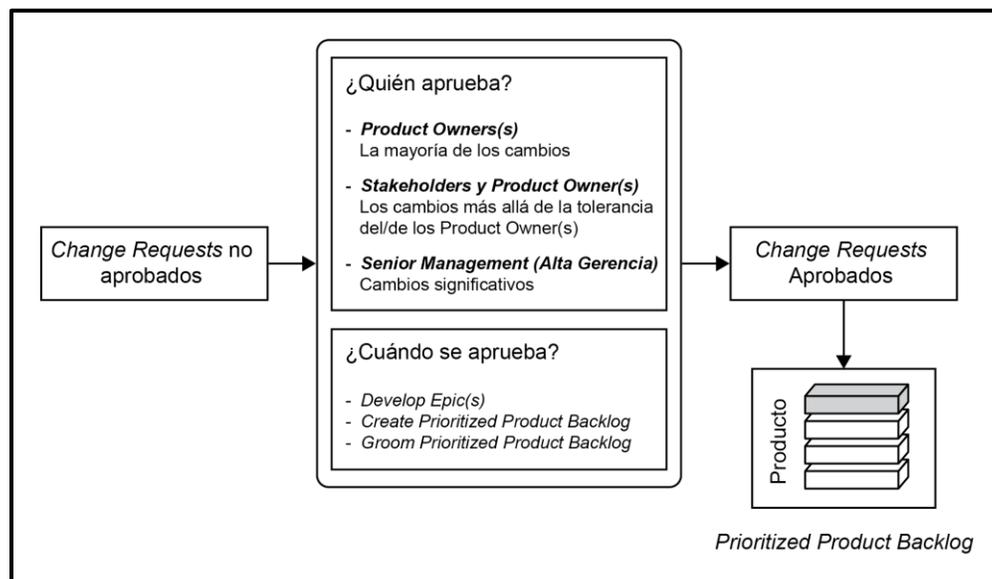


Figura 6-1: Ejemplo del proceso de aprobación de cambios

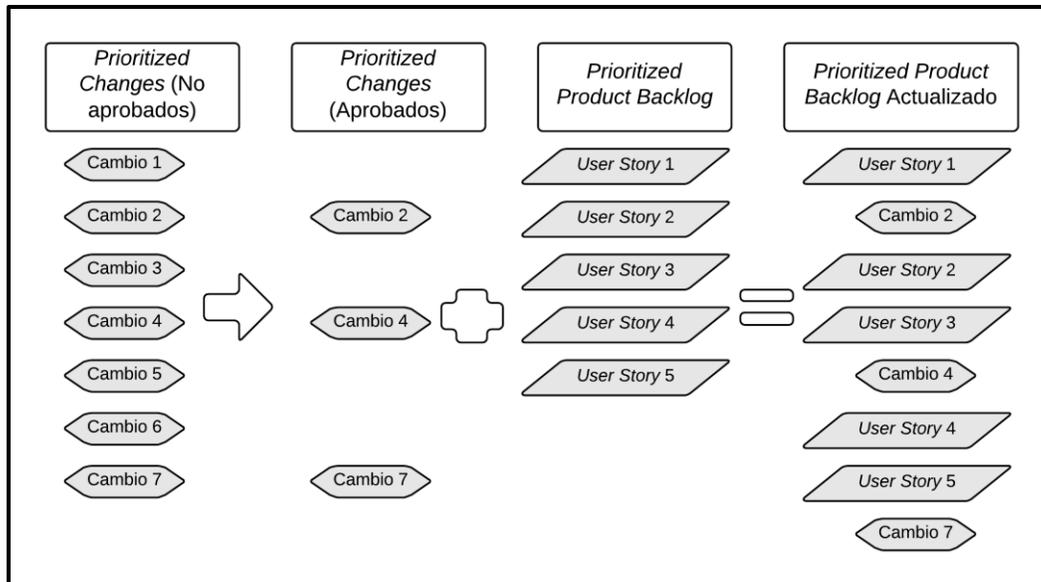


Figura 6-2: Actualización de la lista priorizada de pendientes del producto con los cambios aprobados

## 6.4 Cambio en Scrum

### 6.4.1 Equilibrio entre flexibilidad y estabilidad

Scrum ayuda a las organizaciones a ser más flexibles y receptivas al cambio. Sin embargo, es importante entender que, aunque el marco de Scrum hace hincapié en la flexibilidad, también es de importancia tener estabilidad durante todo el proceso de cambio. De la misma manera que la rigidez extrema es ineficaz, la flexibilidad extrema también es improductiva. La clave es encontrar el equilibrio adecuado entre la flexibilidad y la estabilidad, ya que la estabilidad se necesita a fin de realizar el trabajo. Por lo tanto, Scrum utiliza el desarrollo iterativo y sus otras características y principios para lograr este equilibrio. Scrum mantiene la flexibilidad de que las solicitudes de cambio se pueden crear y aprobar en cualquier momento durante el proyecto; sin embargo, estas se priorizan cuando se crea o se actualiza la lista priorizada de pendientes del producto. Al mismo tiempo, Scrum asegura que la estabilidad permanezca al mantener la lista de pendientes del sprint, y al no permitir la interferencia con el equipo Scrum durante un sprint.

En Scrum, todos los requisitos relacionados con el Sprint en curso se suspenden durante el Sprint. Ningún cambio se introduce hasta que se termina el Sprint, a menos que un cambio se considere lo suficientemente importante como para detener el Sprint. En el caso de un cambio urgente, el Sprint se termina y el equipo se reúne para planificar uno nuevo. Así es como Scrum acepta cambios sin cambiar las fechas de lanzamiento.

## 6.4.2 El logro de la flexibilidad

Scrum facilita la flexibilidad a través de la *transparencia, inspección y adaptación* para lograr los resultados de negocio más valiosos. Scrum proporciona un mecanismo de adaptación para la gestión de proyectos en el que un cambio en los requisitos se puede acomodar sin afectar significativamente el progreso general del proyecto. Es necesario adaptarse a las realidades de los negocios emergentes como parte del ciclo de desarrollo. La flexibilidad en Scrum se logra a través de cinco características claves (véase la Figura 6-3): el desarrollo de productos iterativos, la asignación de un bloque de tiempo, los equipos interfuncionales, la priorización basada en el cliente y la integración continua.

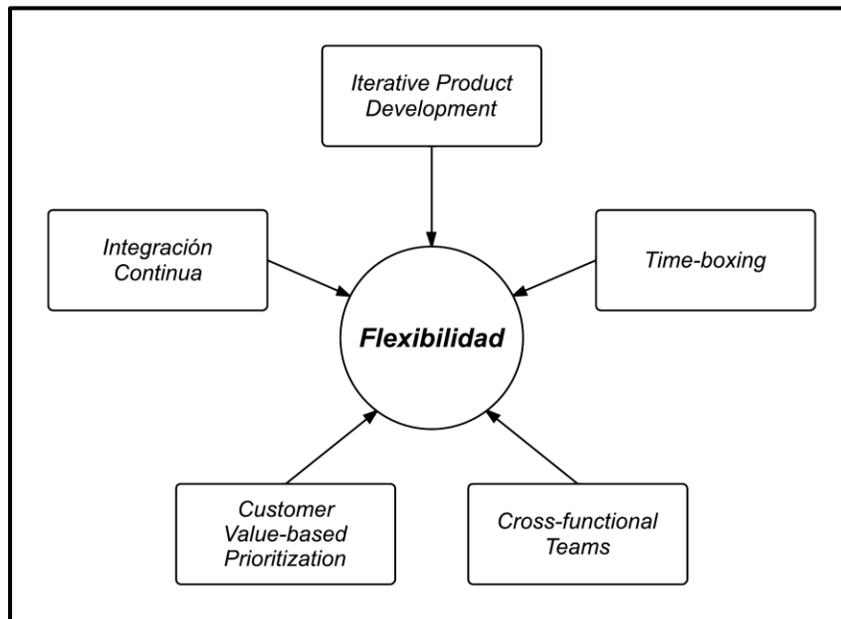


Figura 6-3: Características de Scrum para lograr flexibilidad

### 6.4.2.1 La flexibilidad mediante el desarrollo iterativo del producto

Scrum sigue un enfoque iterativo e incremental de desarrollo de productos y servicios, por lo que es posible la incorporación de cambios en cualquier paso en el proceso de desarrollo. A medida que se desarrolla el producto, una solicitud de cambio para el proyecto puede provenir de múltiples fuentes de la siguiente manera:

#### 1. Los socios

Los socios del proyecto —en particular los patrocinadores, clientes, y los usuarios— podrán presentar solicitudes de cambio en cualquier momento durante todo el proyecto. Las solicitudes podrían deberse a cambios en las condiciones del mercado, la dirección de la organización, asuntos legales o reglamentarios, o a varias otras razones. Por otra parte, los socios pueden presentar dichas solicitudes a medida que van revisando los entregables durante los procesos de

*demonstración y validación del Sprint, retrospectiva del sprint o la retrospectiva del proyecto.* Todas las solicitudes de cambio se añaden a la *lista priorizada de pendientes del producto del proyecto* (también denominada lista priorizada de pendientes del producto o lista de pendientes del producto), una vez que se aprueba. La figura 6-4 muestra algunas de las razones por las que los socios inician el proceso de solicitud de cambio.

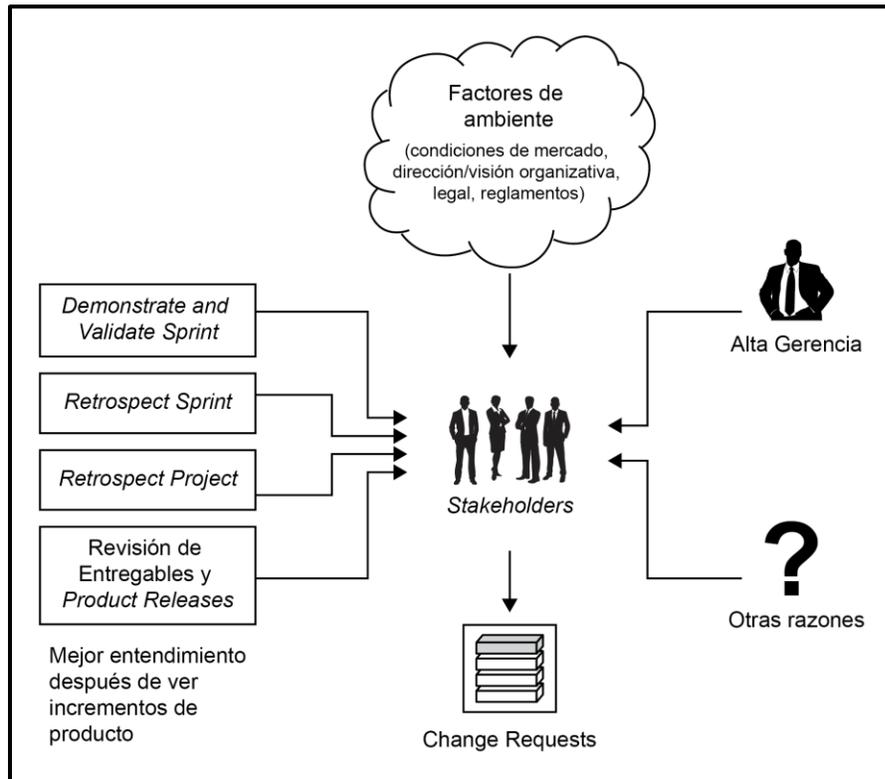


Figura 6-4: Motivación de los socios para la solicitud de cambios

## 2. El equipo principal de Scrum

El equipo principal de Scrum (es decir, el propietario del producto, el Scrum Master y el equipo Scrum) participa en la creación de los entregables del producto. La interacción continua entre los miembros principales de un equipo Scrum y otros (como otros equipos Scrum del proyecto, y los socios internos y externos del proyecto) puede motivarlos a sugerir cambios o mejoras en el producto, servicio, o cualquier otro aspecto del proyecto. Normalmente, estos cambios —al igual que otros— se incluyen en las solicitudes de cambio, y el propietario del producto toma una decisión final acerca de qué sugerencias de cambio por parte del equipo Scrum o Scrum Master se deben considerar como solicitudes de cambio formales.

A veces pueden existir desafíos con la creación de ciertos entregables, lo que puede resultar en solicitudes de cambio. Por ejemplo, el equipo puede añadir o modificar una característica para

mejorar el rendimiento del producto. En la mayoría de los proyectos Scrum, las recomendaciones de cambios por parte del equipo principal de Scrum suceden durante el proceso de *creación de entregables*, o cuando participa en las *reuniones diarias* o las *reuniones de retrospectiva del sprint*. La figura 6-5 muestra algunas de las razones por las que el equipo principal de Scrum puede iniciar solicitudes de cambio.

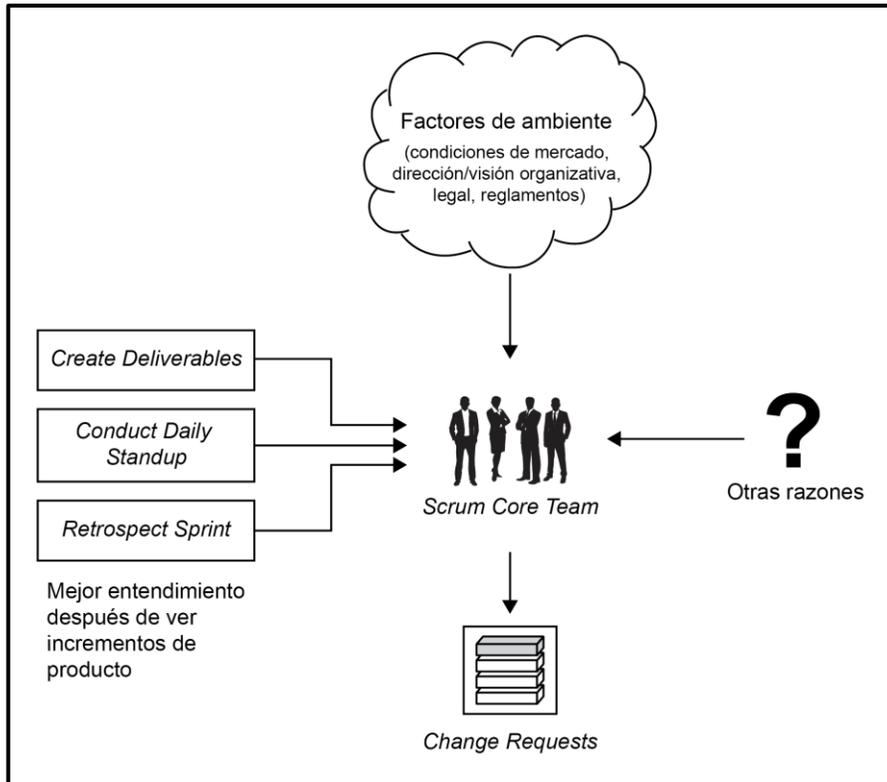


Figura 6-5: Motivación del equipo principal de Scrum para la solicitud de cambios

### 3. La alta gerencia

La gerencia general —incluyendo la gerencia de la cartera y del programa— puede recomendar cambios que afectan al proyecto. Esto puede deberse a cambios estratégicos en la dirección de la empresa, a un entorno competitivo, a cambios de financiación relacionadas con problemas, y así sucesivamente. Es importante tener en cuenta que estos cambios se añaden a la lista priorizada de pendientes del producto y tienen que pasar por el proceso normal de gestión del cambio. Si algunos de estos cambios son urgentes, es posible que cualquier sprint afectado necesite terminarse (véase la sección 6.6 para más detalles).

#### 4. El cuerpo de asesoramiento de Scrum

El cuerpo de asesoramiento de Scrum puede presentar solicitudes de cambio que afectan a todos los proyectos debido a alguno de los siguientes ejemplos:

- Cambios en las regulaciones del gobierno (por ejemplo, la privacidad, las normas de seguridad, o una nueva legislación)
- Directivas corporativas de calidad, seguridad, u otras iniciativas de la organización que deben ponerse en práctica en toda la compañía
- Puntos de referencia o mejores prácticas para alcanzar cierto nivel
- Lecciones aprendidas de proyectos anteriores que se puedan implementar por otros equipos Scrum

El sello distintivo de Scrum es que es tolerante y adaptable a los cambios. Scrum no promueve determinar y establecer planes con mucha firmeza y anticipación, ya que opera en la premisa de que el desarrollo del proyecto es muy propenso al cambio y al riesgo. El resultado es un alto grado de flexibilidad y tolerancia al cambio. El proyecto se lleva a cabo y se gestiona de forma incremental, por lo que generalmente es fácil incorporar cambios a lo largo del proyecto.

6

##### 6.4.2.2 La flexibilidad mediante la asignación de un bloque de tiempo

La asignación de un bloque de tiempo se refiere al establecimiento de periodos cortos de tiempo para que el trabajo sea hecho. Si el trabajo realizado queda incompleto al final del bloque de tiempo, este se asigna a un nuevo bloque. Algunos ejemplos de bloques de tiempo incluyen el limitar las reuniones diarias de pie a 15 minutos y el establecer la duración del sprint de una a seis semanas. El asignar bloques de tiempo fijo proporciona la estructura necesaria para los proyectos Scrum, los cuales tienen un elemento de incertidumbre, son dinámicos por naturaleza y propensos a cambios frecuentes. Estos bloques de tiempo ayudan a medir el progreso del proyecto y permiten que el equipo identifique fácilmente cuándo se necesitará modificar un proceso o método.

Los sprints que se asignan a un bloque de tiempo contribuyen en gran medida al cumplimiento de los plazos, así como al logro de altos niveles de productividad. Los sprints promueven el orden y la consistencia en un ambiente de trabajo volátil, y proporcionan una plataforma para medir resultados, así como para obtener información en un corto espacio de tiempo. Los sprints también permiten una evaluación frecuente de los progresos y los métodos que se utilizan para gestionar el proyecto, incluyendo la gestión del cambio eficaz. Los errores o problemas pueden identificarse en una etapa temprana y rectificarse rápidamente.

Mediante el uso de la asignación de bloques de tiempo en los sprints, el equipo repasa con frecuencia el proceso de estimación del trabajo que se va a realizar, por lo que la proyección de tiempo y esfuerzo que se requieren es más precisa con cada sprint subsecuente a medida que el proyecto avanza. Estos ciclos iterativos también motivan a los miembros del equipo a lograr los objetivos previstos y las metas incrementales, a fin de alcanzar el objetivo mayor.

### 6.4.2.3 La flexibilidad a través de equipos interfuncionales y auto-organizados

Las estructuras interfuncionales y auto-organizadas del equipo Scrum les permiten a los miembros del equipo enfocarse en los resultados deseados del sprint. El equipo tiene un conjunto definido de objetivos durante cada sprint, así como la flexibilidad para dar cuenta de un cambio en los objetivos antes de comenzar el siguiente sprint.

El uso de equipos interfuncionales también se asegura de que todas las habilidades y conocimientos necesarios para llevar a cabo el trabajo del proyecto existan dentro del equipo. Esto proporciona un modelo de trabajo eficiente que da lugar a la creación de entregables listos para mostrarlos al propietario del producto y/u otros socios.

La auto-organización garantiza que los miembros del equipo Scrum decidan por sí mismos la forma de hacer el trabajo del proyecto sin la microgestión de las tareas por un alto directivo.

El tener equipos interfuncionales y auto-organizados le permite al grupo adaptarse y administrar los trabajos en curso y algunos problemas menores o cambios sin tener que obtener el apoyo o la experiencia de miembros fuera del equipo, y en el proceso, crear entregables que estén listos para enviarse en caso de ser necesario.

### 6.4.2.4 Flexibilidad a través de la priorización basada en el valor que se entrega al cliente

La priorización de las necesidades y el trabajo en un proyecto Scrum siempre se determina en base al valor proporcionado al cliente. En primer lugar, al inicio de un proyecto, los requisitos iniciales se priorizan en función del valor que cada requisito proveerá —esto está documentado en la lista priorizada de pendientes del producto. Cuando se realiza una solicitud para un requisito nuevo o un cambio a uno ya existente, esto se evalúa durante el proceso de *mantenimiento de la lista priorizada de pendientes del producto*. Si se considera que el cambio proporcionará más valor que otros requisitos existentes, se añadirá y priorizará de acuerdo a la versión actualizada de la lista priorizada de pendientes del producto. Así, la lista priorizada de pendientes del producto ofrece la posibilidad de incorporar cambios y agregar nuevos requisitos cuando sea necesario.

Es importante tener en cuenta que los nuevos requisitos y cambios añadidos a la lista priorizada de pendientes del producto puede disminuir la prioridad de otras historias de usuarios existentes en la lista de pendientes: por tanto, estas historias de usuario de menos prioridad se pueden implementar más adelante dependiendo de su nueva priorización. Debido a que los clientes están íntimamente involucrados con la priorización de las necesidades y de sus historias de usuario correspondientes en la lista priorizada de pendientes del producto, esta práctica asegura que los requisitos que los clientes consideran de "alto valor" se completen lo antes posible, y que el proyecto inicie la entrega de valor con mayor anticipación.

### 6.4.2.5 Flexibilidad a través de la integración continua

Utilizando técnicas de integración continua, los miembros del equipo Scrum pueden incorporar características nuevas y modificadas en las entregas siempre que sea posible. Esto reduce el riesgo de que varios miembros del equipo hagan cambios en componentes redundantes (por ejemplo, código obsoleto en los productos de software, o diseños antiguos para la fabricación de piezas). Esto asegura que se esté trabajando solamente en la última función de la versión y en evitar problemas de compatibilidad.

## 6.5 Integración del cambio

Dependiendo de la industria y el tipo de proyecto, la prioridad de las características y los requisitos para un proyecto pueden permanecer fijos durante periodos significativos de tiempo, o bien pueden cambiar con frecuencia. Si los requisitos del proyecto son generalmente estables, normalmente sólo hay pequeños cambios realizados en la lista priorizada de pendientes del producto en todo el desarrollo, y los equipos Scrum pueden trabajar secuencialmente en completar los requisitos que le proporcionan el valor máximo al cliente como se priorizó en la lista priorizada de pendientes del producto. En entornos estables, la duración del sprint es generalmente más larga, de 4 a 6 semanas.

Si los requisitos del proyecto cambian durante la duración del mismo debido a, por ejemplo, las modificaciones de los requerimientos de negocio, el mismo método sigue siendo eficaz. Antes de comenzar un sprint —durante los procesos de *creación de la lista priorizada de pendientes del producto* o *mantenimiento de la lista priorizada de pendientes del producto*— los requisitos de mayor prioridad en la lista priorizada de pendientes del producto se seleccionan normalmente para completarse en ese sprint. Dado a que los cambios se han tenido en cuenta en la lista priorizada de pendientes del producto, el equipo sólo tiene que determinar el número de tareas que se pueden realizar en el sprint, basado en el tiempo y los recursos proporcionados. La gestión del cambio se lleva a cabo en los procesos de priorización actuales, y se le agregan tareas a la lista priorizada de pendientes del producto.

### 6.5.1 Los cambios a un sprint

Si hay una solicitud de cambio que puede tener un impacto significativo sobre un sprint en curso, el propietario del producto, después de consultar con socios relevantes, decide si el cambio puede esperar hasta el próximo sprint o si representa una situación urgente que puede requerir finalizar el sprint actual y comenzar uno nuevo.

El marco de Scrum especifica claramente que el alcance de un sprint no puede cambiarse una vez que este comience. Si el cambio requerido es tan importante que los resultados del sprint no tendrían ningún valor sin él, entonces el sprint debe terminarse. De lo contrario, entonces el cambio se incorpora más adelante en un sprint (como se muestra en la Figura 6-6).

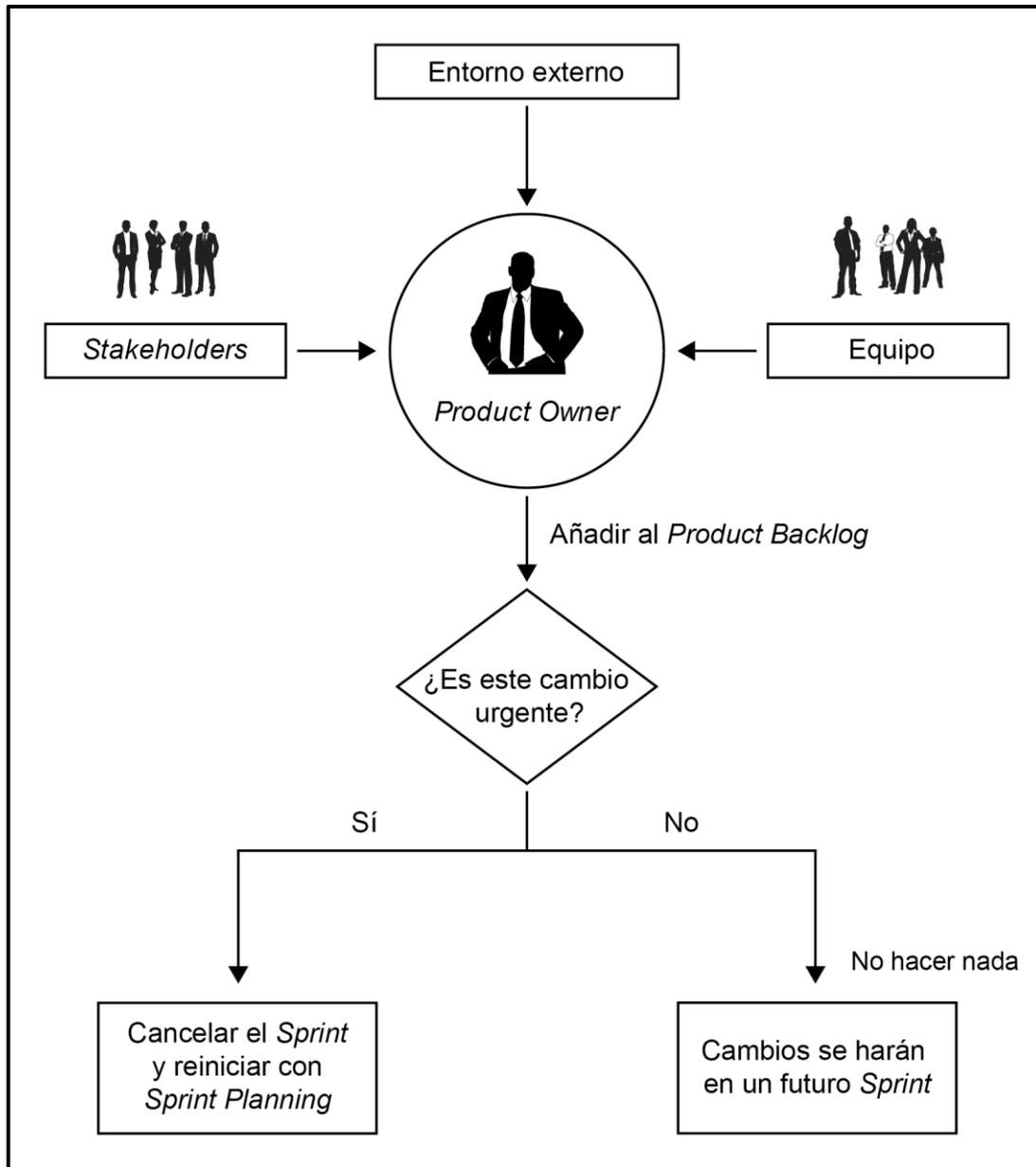


Figura 6-6: Integración del cambio en Scrum

Sólo hay una excepción a esta regla de no modificar el alcance de un sprint una vez que ha comenzado. Si el equipo Scrum determina que se ha sobrestimado en gran medida el esfuerzo durante el sprint y no tiene capacidad para poner en práctica historias de usuario adicionales, el equipo puede preguntarle al propietario del producto cuáles historias de usuario han de incorporarse en el sprint actual.

Al bloquear el alcance de cada sprint, el equipo es capaz de optimizar y administrar con eficiencia su trabajo y esfuerzo. Un beneficio adicional es que el equipo no tiene que preocuparse por la gestión de los cambios una vez que comienzan a trabajar en un sprint. Esta es una gran ventaja del marco de Scrum en comparación con la gestión tradicional de proyectos.

En la gestión tradicional de proyectos, los cambios pueden solicitarse y aprobarse en cualquier momento durante el ciclo de vida del proyecto. Esto a menudo causa confusión entre los miembros del equipo del proyecto, disminuye la motivación del equipo debido a la discontinuidad, da lugar a una falta de concentración, y el equipo tiene la sensación de que "nunca se termina nada". En cambio, en los proyectos Scrum, los cambios no se permiten una vez que se inicia un Sprint. Esto asegura que, en cada Sprint, el equipo complete entregables y que las tareas se lleven a cabo. Por otra parte, el negocio reconoce los beneficios tangibles de los entregables que están potencialmente listos para la entrega al final de cada sprint.

Además, como el propietario del producto y los socios están conscientes de que los cambios no se permiten una vez que el sprint comienza, y un sprint dura entre 1 y 6 semanas, ellos definen y priorizan las necesidades durante los procesos adecuados de creación de épica(s), la creación de la lista priorizada de pendientes del producto, y el mantenimiento de la lista priorizada de pendientes del producto.

### 6.5.1.1 Impacto del cambio esperado en la duración del sprint

Dado que los cambios no están permitidos durante un sprint, el impacto y la frecuencia de los cambios previstos pueden tener un impacto en la decisión relacionada con la longitud del Sprint cuando ésta se determina durante el proceso de *realizar la planificación del lanzamiento*.

Si los requisitos del proyecto son generalmente estables y no se esperan grandes cambios en un futuro próximo, la duración de un sprint se puede ajustar para que sea más larga, de 4 a 6 semanas. Esto les proporciona estabilidad a los miembros del equipo Scrum para trabajar en los requisitos de lista priorizada de pendientes del producto durante largos periodos de tiempo sin tener que pasar por los procesos de *creación de historias de usuario; aprobación, estimación y asignación de historias de usuario; creación de tareas; estimación de tareas*, y otros procesos relacionados que se llevan a cabo para cada sprint.

Sin embargo, si los requisitos del proyecto no están muy bien definidos o si se esperan cambios significativos en el futuro inmediato, la duración del sprint puede ser relativamente corta, de 1 a 3 semanas. Esto les proporciona estabilidad a los miembros del equipo Scrum para trabajar en sprints más cortos y entregar resultados, los que se pueden evaluar por el propietario del producto y los socios al final del sprint. Esto también proporciona la flexibilidad suficiente para que puedan aclarar los requisitos y realizar cambios en la lista priorizada de pendientes del producto al final de cada Sprint.

Para obtener los máximos beneficios de un proyecto Scrum, siempre se recomienda mantener el sprint bajo un bloque de tiempo de cuatro semanas, a menos que existan proyectos con requisitos muy estables, donde los sprints pueden extenderse hasta seis semanas.

La Figura 6-7 muestra el impacto del cambio esperado en la duración del sprint.

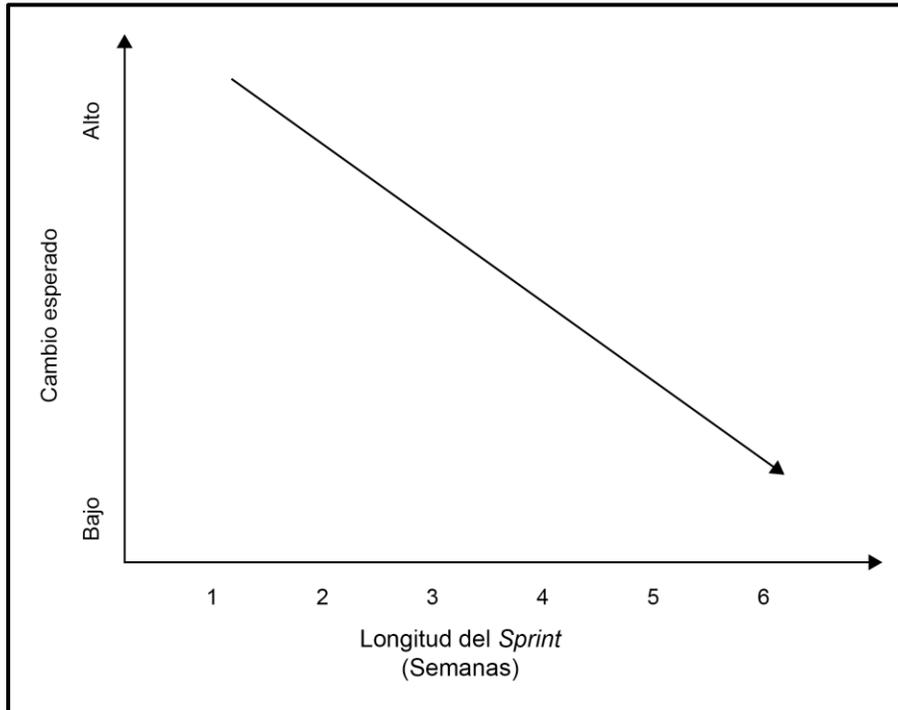


Figura 6-7: Impacto del cambio esperado en la duración del sprint

Sin embargo, es importante tener en cuenta que el cambio esperado no es el único factor que se utiliza para determinar la duración del sprint. Otros factores que también deben tenerse en cuenta son:

- El tiempo real para realizar su trabajo (si el proyecto o entorno corporativo necesita un tiempo específico para realizar tareas de forma, eso podría determinar la duración del sprint)
- La fecha prevista para su lanzamiento (la duración del sprint debe tener en cuenta las fechas de lanzamiento para el producto o el servicio en general)
- Cualquier otro factor que determine el propietario del producto o el Scrum Master que deben tenerse en cuenta al determinar la duración del sprint

Es importante tener en cuenta que el cambio en la duración del sprint no debe decidirse a la ligera o de manera periódica (por ejemplo, no es recomendable tener un sprint de 3 semanas, luego uno de 2 semanas, y el siguiente de 4 semanas, etc.) La duración del sprint debe ser consistente preferiblemente. Uno de los mayores impactos del cambio de la duración del sprint es que causa un restablecimiento en todo el seguimiento a nivel de proyecto. Las velocidades anteriores pueden llegar a ser inútiles para la previsión y la planificación de los futuros sprints. Sin una velocidad precisa (que es una medida primaria en cualquier proyecto Scrum), el equipo Scrum no puede medir la eficacia o elegir adecuadamente el número de historias para asumir la planificación del próximo sprint. Así que, una vez que la duración del sprint se decide, se debe mantener constante preferiblemente durante toda la duración del proyecto o a través de múltiples ciclos de sprint.

### 6.5.1.2 Gestión de cambios mediante el mantenimiento de la lista priorizada de pendientes del producto

Una lista priorizada de pendientes del producto típica tendrá todas las historias de usuarios, sus estimaciones de tiempo (incluyendo las estimaciones revisadas), y el estado de las necesidades de mayor prioridad. También se incorporan historias de usuario nuevas o revisadas que resultaron de cambios en los requerimientos de negocio, pedidos de los clientes, condiciones externas del mercado, y/o lecciones aprendidas en sprints anteriores.

Una de las responsabilidades principales de los propietarios del producto es preparar la lista priorizada de pendientes del producto para garantizar que los requisitos priorizados en dicha lista se incluyan en los próximos dos o tres sprints, y se refinan en acuerdo con las historias de usuario. Se recomienda que el propietario del producto pase una cantidad significativa de tiempo en cada sprint para mantener la lista. El propietario del producto es responsable de añadir y modificar elementos a dicha lista en respuesta a los cambios, así como de proporcionar historias de usuario más detalladas que se utilizarán en el próximo sprint.

Este mantenimiento ayuda a asegurar que la refinación de los requisitos y sus historias de usuario se hagan mucho antes de la reunión de planificación del sprint, a fin de que el equipo tenga un conjunto de historias muy bien analizado y claramente definido que pueda dividirse fácilmente en tareas y, posteriormente, ser estimado. Basado en las lecciones aprendidas del sprint actual, puede haber cambios en los requisitos, o bien una priorización nueva que pueda incorporarse fácilmente en sprints posteriores. Este mantenimiento apoya y mejora la flexibilidad del modelo Scrum mediante la incorporación de los últimos avances técnicos y de negocio en futuros sprints.

La reunión de revisión de la lista priorizada de pendientes del producto es una reunión formal durante el proceso de mantenimiento de la lista priorizada de pendientes del producto, que ayuda al equipo Scrum a repasar y alcanzar consenso sobre el mantenimiento de la lista priorizada de pendientes del producto. Sin embargo, aparte de la reunión de revisión de la lista priorizada de pendientes del producto, la lista priorizada de pendientes del producto, el mantenimiento debería ocurrir durante todo el proyecto y puede incluir situaciones en las que el propietario del producto escribe nuevas historias de usuarios o vuelve a priorizar las historias de usuario en la lista priorizada de pendientes del producto vigente, y los miembros del equipo Scrum o socios ofrecen sus sugerencias sobre las nuevas historias al propietario del producto, y así sucesivamente.

Es importante tener en cuenta que cualquier elemento de la lista priorizada de pendientes del producto está siempre abierto para la re-estimación hasta que la lista de pendientes del sprint sea finalizada en el proceso de creación de la lista de pendientes del sprint. Después de eso, los cambios se podrán seguir haciendo inclusive hasta momentos antes de la reunión de planificación del sprint, si es necesario.

#### **6.5.1.2.1 Reunión eficaz de revisión de la lista de pendientes del producto (o sesión de mantenimiento de la lista priorizada de pendientes del producto)**

El propietario del producto es quien está encargado de que se lleve a cabo una reunión de revisión de la lista de pendientes del producto durante el proceso de *mantenimiento de la lista priorizada de pendientes del producto*. Es importante que el propietario del producto establezca los objetivos, y lo ideal sería desarrollar una agenda antes de comenzar esta reunión. Sin esto, la sesión no tendría estructura y podría resultar improductiva. También es importante limitar el número de socios que participan en la reunión. El tener demasiados participantes tiende a disminuir la eficiencia general de la reunión. El propietario del producto debe invitar sólo a los socios cuyas votaciones se requieren para la sesión de la preparación. Se deben incluir todos los miembros del equipo Scrum debido a que su opinión es valiosa para el trabajo que se realiza y los problemas que se encontraron. Si los resultados de la sesión de preparación o cuidado resultan en nueva priorización o cambio en la lista priorizada de pendientes del producto, es importante que el equipo esté de acuerdo con esos cambios.

Una sesión de preparación eficaz debe dar lugar a los elementos de la lista priorizada de pendientes del producto claramente definidos para que el equipo Scrum entienda los requisitos del cliente. Esto también ayuda a que el equipo se familiarice con todas las historias de usuario en caso de que una o más de ellas sean incluidas en un sprint a corto plazo. También pueden tratarse los criterios de aceptación y de terminado durante las sesiones de preparación.

Los ejercicios de mantenimiento no se asignan a un bloque de tiempo en Scrum. El mantenimiento de la lista priorizada de pendientes del producto es una actividad continua para el propietario del producto.

#### **6.5.1.3 Gestión de cambios durante la demostración y validación del Sprint**

Aunque el propietario del producto tiene la última palabra sobre los elementos de la lista priorizada de pendientes del producto, y si se deben aceptar o rechazar las historias de usuario (correspondientes a las solicitudes de cambio aprobadas) presentadas durante el proceso de demostración y validación del Sprint, es la responsabilidad del Scrum Master garantizar que los requisitos y criterios de aceptación no se alteren durante la reunión de revisión del sprint de las historias de usuario completadas en el sprint actual. Esto evita el rechazo de futuras historias de usuario basado en el hecho de que no cumplen los requisitos recién cambiados. Si los requisitos se deben cambiar, cualquier elemento de la lista priorizada de pendientes del producto correspondiente debe revisarse para adaptarse a los requisitos modificados en un sprint futuro.

## 6.6 Cambio en carteras y programas

Cualquier cambio que se produce cualquiera de los programas o carteras puede tener un efecto en cascada en todos los proyectos dependientes y los sprints. Por lo tanto, es aconsejable minimizar los cambios en estos niveles más altos. Si se requiere un cambio y todos los socios están de acuerdo en hacer el cambio a estos niveles, lo siguiente deberá tenerse en cuenta.

### 6.6.1 En carteras

1. No se recomienda hacer cambios entre dos reuniones de la lista de pendientes de la cartera.
2. Si el cambio es menor, el propietario del producto de la cartera debe asegurar la aprobación de los socios correspondientes (por ejemplo, el patrocinador, el cliente, y el usuario meta), y luego añadir los requisitos a la lista de pendientes de la cartera. Los propietarios del producto del programa y del proyecto tendrán en cuenta los requisitos para su inclusión en futuros sprints.
3. Si el cambio es importante, los esfuerzos de la cartera, junto con los programas asociados, los proyectos y los sprints tienen que detenerse, y una reunión de la lista de pendientes de la cartera debe llevarse a cabo para determinar los siguientes pasos.
4. Las reuniones de la lista priorizada de pendientes de la cartera (también llamadas reuniones de la lista de pendientes de la cartera), deben llevarse a cabo en intervalos de 4 a 12 meses. La frecuencia y el impacto de los cambios en una cartera determinan en gran medida la duración de tiempo entre dos reuniones de la lista de pendientes de la cartera. Si son varios los cambios esperados en la cartera, es preferible llevar a cabo este tipo de reuniones, en intervalos más regulares (por ejemplo, 4 a 6 meses); pero si hay menos cambios esperados y si los requisitos son estables, la duración entre dos reuniones podría incrementarse (por ejemplo, de 9 a 12 meses).

### 6.6.2 En programas

1. No se recomienda hacer cambios entre dos reuniones de la lista de pendientes del programa.
2. Si el cambio es menor, el propietario del producto del programa debe obtener la aprobación de los socios relevantes (por ejemplo, el patrocinador, el cliente, y el usuario meta), y el propietario del producto de la cartera, y luego agregarle los requisitos a la lista de pendientes del programa. Los propietarios del producto del proyecto tendrán en cuenta los requisitos para la inclusión en futuros sprints.
3. Si el cambio es importante, los esfuerzos de los programas, junto con los proyectos asociados y sprints tienen que detenerse, y la reunión de lista priorizada de pendientes del producto debe llevarse a cabo para determinar los siguientes pasos.
4. Las reuniones de la lista priorizada de pendientes del producto del programa (también conocidas como reuniones de la lista de pendientes del programa), deben llevarse a cabo preferentemente en intervalos de 2 a 6 meses. La frecuencia y el impacto de los cambios en un programa determinan en gran medida la duración de tiempo entre dos reuniones de la lista de pendientes del programa.

Si hay varios cambios previstos en el programa, es preferible llevar a cabo este tipo de reuniones en intervalos más regulares (por ejemplo, de 2 a 3 meses); pero si hay menos cambios esperados y si los requisitos son estables, la duración entre dos reuniones podría aumentarse (por ejemplo, de 5 a 6 meses).

La figura 6-8 muestra cómo se pueden administrar los cambios dentro del flujo de Scrum tanto para las carteras como para los programas.

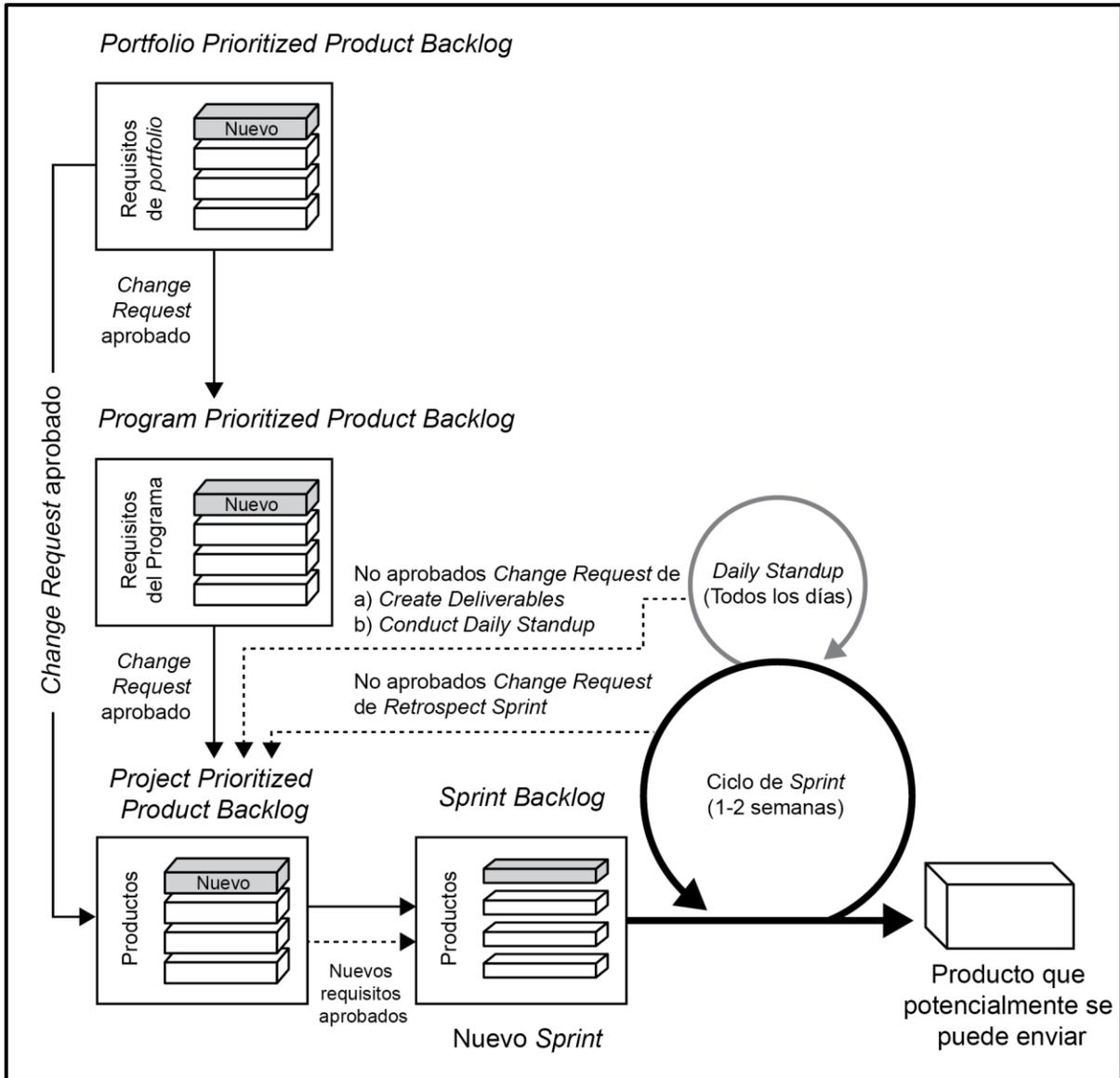


Figura 6-8: La incorporación de cambios en carteras y programas

## 6.7 Resumen de responsabilidades

Rol	Responsabilidades
Cuerpo de asesoramiento de Scrum	<ul style="list-style-type: none"> <li>• Proporcionar una guía general para los procedimientos de gestión de cambios que deben seguirse durante todo el proyecto</li> </ul>
Propietario del producto de la cartera	<ul style="list-style-type: none"> <li>• Proporcionar solicitudes de cambio para las carteras</li> <li>• Aprobar los productos que son modificados, eliminados o añadidos de acuerdo con los requisitos de la cartera</li> </ul>
Scrum Master de la cartera	<ul style="list-style-type: none"> <li>• Facilitar la identificación, evaluación y gestión de las solicitudes de cambio para las carteras</li> </ul>
Propietario del producto del programa	<ul style="list-style-type: none"> <li>• Proporcionar solicitudes de cambio para los programas</li> <li>• Aprobar los productos que son modificados, eliminados o añadidos de acuerdo con los requisitos del programa</li> </ul>
Scrum Master del programa	<ul style="list-style-type: none"> <li>• Facilitar la identificación, evaluación y gestión de las solicitudes de cambio para los programas</li> </ul>
Socios(s)	<ul style="list-style-type: none"> <li>• Proporcionar la solicitud de cambios</li> <li>• Participar en la aprobación y priorización de las solicitudes de cambio</li> </ul>
Propietario del producto	<ul style="list-style-type: none"> <li>• Proporcionar solicitudes de cambio de un proyecto</li> <li>• Evaluar el impacto de las solicitudes de cambio planteadas por la cartera, el programa o el proyecto</li> <li>• Priorizar las historias de usuario en la lista priorizada de pendientes del producto del proyecto</li> <li>• Evaluar el impacto de los problemas sobre los objetivos del proyecto identificados por el equipo Scrum</li> <li>• Proporcionar una comunicación clara a los socios sobre los elementos de la lista de pendientes del producto que se han vuelto a priorizar</li> </ul>
Scrum Master	<ul style="list-style-type: none"> <li>• Facilitar la identificación y evaluación de los problemas y solicitudes de cambio por el equipo Scrum</li> </ul>
Equipo Scrum	<ul style="list-style-type: none"> <li>• Sugerir mejoras o cambios durante los procesos de creación de entregables y realización de la reunión diaria de pie</li> </ul>

Tabla 6-1: Resumen de las responsabilidades relacionadas con el cambio

## 6.8 Scrum vs. Gestión de proyectos tradicional

La gestión del cambio en los proyectos que se gestionan tradicionalmente está estrechamente relacionada con la gestión de la configuración. Todos los cambios se basan en la magnitud de la variación de la base de referencia del valor. Al director del proyecto se le permite gestionar las actividades y decisiones diarias del proyecto. Cuando una solicitud de cambio supera las tolerancias definidas, el director de proyecto debe escalar la propuesta de cambio a niveles superiores de gestión, y esperar la decisión antes de hacerla efectiva. El gerente del proyecto registra primero la petición de cambio en un registro de problemas o cambios, y luego se les entrega el cambio a las autoridades superiores. Estas podrían incluir al patrocinador del proyecto, así como a otros socios relevantes y a aquellos que toman decisiones sobre el caso. En algún momento, se llevará a cabo una evaluación del impacto. Con base al impacto estimado del cambio, se tomará una decisión respecto a si el cambio debe aplicarse o no. El director del proyecto también podrá proponer posibles soluciones a los problemas planteados por el cambio. Si las autoridades superiores deciden proceder con el cambio, el director del proyecto es responsable de asegurar que el cambio se implemente correctamente.

El cambio en Scrum funciona de manera muy diferente en comparación con la gestión tradicional de proyectos. El marco de Scrum está muy sintonizado con la gestión de cambios de manera eficaz y eficiente. Cada vez que el propietario del producto o el equipo Scrum reconoce un problema o defecto o identifica un elemento de lista priorizada de pendientes del producto que necesita modificarse, sustituirse o añadirse, el cambio se realiza en la lista priorizada de pendientes del producto. Del mismo modo, la alta gerencia, el propietario del producto, o el(los) socio(s) puede(n) añadir solicitudes de cambio a la lista priorizada de pendientes del producto. El propietario del producto y el(los) socios(s) aprueba(n) las solicitudes de cambio y las nuevas prioridades de la cartera en consecuencia. Siempre que hay un problema o una nueva exigencia que se debe atender, la cual resulta en cambios inmediatos que afectan el sprint actual, el propietario del producto debe terminar el Sprint con la aprobación de los socios relevantes. Una vez terminado, el Sprint se vuelve a planificar y a reiniciar para incorporar los nuevos requisitos.

Sin embargo, si el problema o cambio no es importante y no garantiza un cambio dentro del sprint actual, el cambio se añadirá a la lista priorizada de pendientes del producto, y se incorporará en la planificación para un futuro sprint. Esto les da a los socios la capacidad de responder a los cambios en el ambiente externo, mientras se mantiene un cierto grado de control sobre las actividades en curso dentro del proyecto. Además, al final de cada Sprint, el equipo Scrum muestra los entregables clasificados como terminados. Estos entregables que potencialmente se pueden enviar, pueden revisarse por el propietario del producto y otros socios.

## 7. RIESGO

### 7.1 Introducción

El propósito de este capítulo es definir los riesgos, analizar la gestión de riesgos en un entorno de Scrum y considerar las herramientas que faciliten la gestión de los riesgos. Para garantizar la viabilidad del negocio, reducir la probabilidad de fracaso de los proyectos, y tomar decisiones de negocio más informadas, es importante que los riesgos se gestionen con eficacia mediante un enfoque metódico y bien organizado.

En un entorno de Scrum, los riesgos generalmente se minimizan, en gran parte debido al trabajo que se realiza en los sprints, donde se produce una serie continua de entregables en ciclos muy cortos; los entregables se comparan con las expectativas, y el propietario del producto participa activamente en el proyecto. Sin embargo, hasta en el más simple de los proyectos, las cosas pueden salir mal, por lo que es importante contar con una estrategia para identificar y abordar los riesgos.

El *riesgo*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se les entregará a los socios;
- Proyectos de cualquier tamaño o complejidad.

El término “producto” en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Este capítulo se divide en las siguientes secciones:

**7.2 Guía de roles**—Esta sección ofrece orientación sobre cuáles secciones son relevantes para cada uno de los roles centrales de Scrum: El propietario del producto, el Scrum Master y el equipo Scrum.

**7.3 ¿Qué es un riesgo?** —En esta sección se define el riesgo y se explica cómo puede afectar los objetivos de un proyecto y contribuir al éxito o al fracaso del mismo.

**7.4 Procedimiento de gestión de riesgos**—Esta sección presenta las técnicas claves de la gestión de riesgos y profundiza en el desarrollo de estrategias para identificar, evaluar y gestionar riesgos.

**7.5 Minimizar el riesgo mediante el uso de Scrum**—Esta sección explica los aspectos claves de Scrum que lo convierten en un marco de gestión ideal para manejar con eficacia los riesgos en varios niveles: cartera, programa y proyecto.

**7.6 Resumen de responsabilidades**—Esta sección describe las responsabilidades pertinentes a la calidad para cada persona o rol en un proyecto.

**7.7 Scrum vs. Gestión tradicional de proyectos**—Esta sección destaca los beneficios de la gestión de calidad en el método Scrum en comparación a los modelos tradicionales de gestión de proyectos.

## 7.2 Guía de roles

4. Propietario del producto—La principal responsabilidad en la gestión de riesgos en un proyecto recae sobre el propietario del producto; por ello, todo este capítulo aplica principalmente a este rol.
5. Scrum Master—El Scrum Master debe estar familiarizado con todo este capítulo, con un enfoque principal en las secciones 7.3, 7.4 y 7.7.
6. Equipo Scrum—El equipo Scrum debe enfocarse principalmente en las secciones 7.3 y 7.7.

7

## 7.3 ¿Qué es un riesgo?

El riesgo se define como un evento incierto o un conjunto de eventos que pueden afectar los objetivos de un proyecto y pudieran contribuir a su éxito o fracaso. Los riesgos con un potencial de impacto positivo en el proyecto se denominan “oportunidades”; mientras que las amenazas son riesgos que pudieran afectar negativamente a un proyecto. La gestión de riesgos debe hacerse con proactividad, y es un proceso iterativo que debería empezar al inicio del proyecto y continuar durante toda la vida del mismo. El proceso de gestión de riesgos debe seguir algunos pasos estandarizados para garantizar que los riesgos sean identificados, evaluados y se determine un curso de acción para actuar en consecuencia.

Es necesario identificar, evaluar y responder a los riesgos basándose principalmente en dos factores: la probabilidad de ocurrencia y el impacto probable en caso de que ocurra. Los riesgos de alta probabilidad y con un alto índice de impacto deben ser abordados antes de aquellos con una calificación más baja. En general, una vez que se detecte un riesgo, es importante comprender los aspectos básicos del riesgo respecto a las posibles causas, el área de la incertidumbre y los efectos potenciales si se produce el riesgo.

### 7.3.1 Diferencia entre riesgos y problemas

Los riesgos son las incertidumbres relacionadas con un proyecto que podrían alterar considerablemente el resultado del proyecto de manera positiva o negativa. Debido a que los riesgos son incertidumbres a futuro, no tienen ningún impacto actual en el proyecto, pero podrían tener un impacto potencial en el futuro. Los siguientes son algunos ejemplos de riesgos:

- Incluso después de una amplia capacitación, es posible que los representantes de servicio al cliente no estén listos para tomar pedidos el día oficial del lanzamiento.

- Es posible que una cuadrilla de pintores se retrase debido a las fuertes lluvias, lo cual pudiera influir negativamente en el cronograma del proyecto.

Los problemas generalmente son certezas que se están suscitando en el proyecto, por lo que no hay necesidad de realizar una evaluación de la probabilidad como lo haríamos con un riesgo. Los problemas deben atenderse. Los siguientes son algunos ejemplos de problemas:

- No se autoriza el financiamiento.
- Los requisitos no son claros.

Si no se atienden a tiempo los riesgos, estos podrían convertirse en problemas. El objetivo de la gestión de riesgos es estar preparados con planes para poder abordar cualquier riesgo que pudiera presentarse.

### 7.3.2 Actitud de riesgo

Entre los socios se incluyen a todas aquellas personas u organizaciones afectadas por el proyecto, así como aquellas que cuentan con la capacidad para afectarlo. Es importante entender la actitud de riesgo de los socios. La actitud de riesgo (conocida en inglés como *Risk Attitude*) se ve influenciada por los siguientes tres factores:

1. **Apetito de riesgo:** Es la cantidad de incertidumbre que está dispuesta a asumir un socio o una organización.
2. **Tolerancia al riesgo:** Indica el grado, cantidad o volumen de riesgo que resistirán los socios.
3. **Umbral de riesgo:** Es el nivel al cual es riesgo es aceptable para la organización del socio. Un riesgo caerá por encima por debajo del umbral de riesgo. Si está por debajo, el socio o la organización es más probable que acepte el riesgo.

En esencia, la actitud de riesgo de los socios determina cuánto riesgo considera aceptable el socio(s). Es un factor determinante cuando deciden tomar acciones para mitigar posibles riesgos adversos. Por lo tanto, es importante entender los niveles de tolerancia de los socios en relación a diversos factores como el costo, la calidad, el alcance y los plazos.

La **función de utilidad** (del inglés: *Utility Function*) es un modelo utilizado para medir la preferencia del socio por el riesgo o su actitud hacia el riesgo. Esto define el nivel del socio para aceptar riesgos. Las tres categorías de la función de utilidad son las siguientes:

1. **Aversión al riesgo:** Describe al socio que no está dispuesto a aceptar un riesgo sin importar el beneficio o la oportunidad anticipada.
2. **Neutral al riesgo:** Describe al socio que ni tiene aversión al riesgo, ni busca riesgos; cualquier decisión que no se ve afectada por el nivel de incertidumbre de los resultados. Cuando dos posibles escenarios llevan el mismo nivel de beneficio, el socio neutral al riesgo no se preocupará si uno de dichos casos es más riesgoso que el otro.

3. **Búsqueda de riesgos:** Se refiere a que un socio esté dispuesto a aceptar el riesgo incluso si ofrece un aumento marginal de retorno o beneficio al proyecto.

## 7.4 Procedimiento de gestión de riesgos

La gestión de riesgos se compone de cinco pasos:

1. **Identificación de riesgos:** Utilizar diversas técnicas para identificar todos los riesgos potenciales.
2. **Evaluación de riesgos:** Evaluar y estimar los riesgos identificados.
3. **Priorización de riesgos:** Dar prioridad al riesgo que habrá de incluirse en la lista priorizada de pendientes del producto.
4. **Mitigación de riesgos:** Desarrollar de una estrategia adecuada para hacer frente a un riesgo.
5. **Comunicación de riesgos:** Comunicar a los socios apropiados los resultados de los primeros cuatro pasos de la gestión de riesgos y determinar su percepción respecto a eventos inciertos.

### 7.4.1 Identificación de riesgos

7

Los miembros del equipo Scrum deben hacer un intento por identificar todos los riesgos que pudieran afectar el proyecto. Con tal solo observar el proyecto desde una perspectiva diferente —y con el uso de una variedad de técnicas—, pueden lograrlo a fondo. La identificación de riesgos se lleva a cabo a lo largo del proyecto y los riesgos identificados se convierten en entradas en varios procesos de Scrum, incluyendo la *Creación de la lista priorizada de pendientes del producto*, el proceso de *Dar mantenimiento a la lista priorizada de pendientes del producto* y la *Demostración y validación del sprint*.

Las siguientes técnicas se utilizan comúnmente para identificar riesgos:

#### 7.4.1.1 Técnicas de identificación de riesgos

1. **Revisar las lecciones aprendidas de los procesos de retrospectiva del sprint o retrospectiva del proyecto.**

Aprender de proyectos similares y de sprints anteriores en el mismo proyecto, al igual que explorar las incertidumbres que afectan a dichos proyectos y sprints, puede ser una forma útil de identificar riesgos.

2. **Listas de verificación de riesgos**

Las listas de verificación de riesgos (conocidas en inglés como: *Risk Checklists*) pueden incluir puntos clave a considerarse cuando se identifican los riesgos, riesgos comunes encontrados en un proyecto Scrum, o incluso categorías de riesgo que el equipo debe atender. Las listas de verificación son una valiosa herramienta que ayuda a garantizar una identificación integral del riesgo.

### 3. Lista corta de riesgos

Conocidas en inglés como *Risk Prompt List*, estas listas se utilizan para estimular el pensamiento respecto a la fuente de donde se pudieran originar los riesgos. Dichas listas para distintas industrias y tipos de proyectos están disponibles al público.

### 4. Lluvia de ideas

Son sesiones donde los socios y los miembros del equipo principal de Scrum comparten abiertamente ideas por medio de discusiones y sesiones de intercambio de conocimientos, generalmente dirigidas por un facilitador.

### 5. Estructura de distribución de riesgos

Una de las herramientas clave que se utilizan para identificar riesgos es la estructura de distribución de riesgos, conocida en inglés como: *Risk Breakdown Structure*. En esta estructura, se agrupan los riesgos con base en sus categorías o modalidades. Por ejemplo, los riesgos se pueden clasificar como financieros, técnicos o relacionados a la seguridad.

7

#### 7.4.1.2 Despunte basado en riesgo

Los despuntes basados en riesgo son básicamente experimentos que implican una investigación o hacer un prototipo para entender mejor los riesgos potenciales. En un *spike*, se lleva a cabo un intenso ejercicio de dos a tres días (de preferencia al inicio de un proyecto, antes del proceso de desarrollo de épicas o de la creación de la lista priorizada de pendientes del producto) para ayudar al equipo a determinar las incertidumbres que pudieran afectar al proyecto. Los despuntes basados en riesgos son útiles cuando el equipo Scrum trabaja con el cliente trabaja y se acostumbra a las nuevas tecnologías y herramientas, o bien, cuando las historias de usuario son muy extensas. Ayuda también a estimar con más precisión el tiempo y el esfuerzo.

#### 7.4.2 Evaluación de riesgos

La evaluación de riesgos ayuda a: entender el impacto potencial de un riesgo; qué posibilidades hay de que suceda y cuándo pudiera materializarse. Se debe estimar el efecto generalizado en el valor empresarial. Si dicho impacto es lo suficientemente considerable como para superar la justificación del negocio, se debe de tomar la decisión si se le da continuidad al proyecto.

La evaluación de riesgo se lleva a cabo en relación a la probabilidad, proximidad e impacto. La probabilidad de riesgo es la probabilidad de su ocurrencia, mientras que la proximidad se refiere a cuándo pudiera suscitarse un riesgo. El impacto es el efecto probable del riesgo sobre un proyecto u organización.

Para estimar la probabilidad de un riesgo, se pueden utilizar varias técnicas, incluyendo árboles de probabilidad, análisis de Pareto y matriz de impacto.

Además de la probabilidad, en la evaluación de riesgos se evalúa también el efecto potencial neto de los riesgos sobre el proyecto o la organización. Dichos efectos se pueden estimar utilizando técnicas tales como modelos de riesgo y valor monetario esperado.

### 7.4.2.1 Técnicas de evaluación de riesgos

#### 1. Reunión de riesgos

Los riesgos se pueden priorizar con facilidad por el propietario del producto convocando a una reunión del equipo principal de Scrum con la opción de invitar a los socios relativos a dicha reunión. El equipo pudiera reunirse y dar prioridad a distintos riesgos con base a su evaluación subjetiva del impacto del riesgo sobre los objetivos del proyecto.

#### 2. Árboles de probabilidad

Los eventos potenciales se representan en un diagrama con una rama para cada resultado posible de los acontecimientos. La probabilidad de cada resultado se indica en la rama apropiada, y estos valores se pueden utilizar para calcular el impacto general de la ocurrencia de riesgos en un proyecto. Los valores resultantes se suman para calcular el impacto general esperado de un riesgo en un proyecto (véase la figura 7-1).

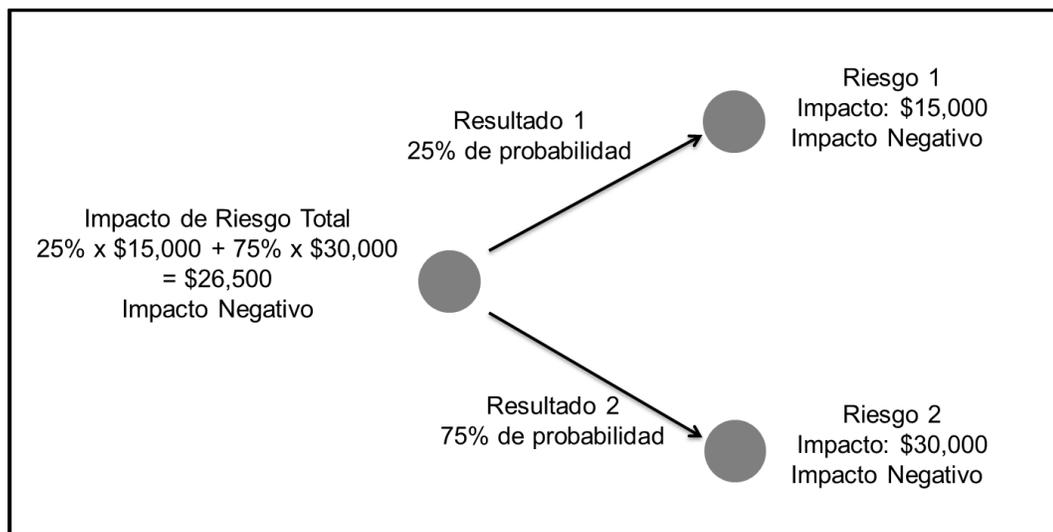


Figura 7-1: Ejemplo de árbol de probabilidad

### 3. Análisis de Pareto

Esta técnica de evaluación de riesgos implica la clasificación de riesgos por magnitud. Ayuda al equipo Scrum a atender los riesgos por orden de impacto probable en un proyecto. Por ejemplo, en la figura 7-2, el Riesgo 1 tiene el mayor impacto y de preferencia debe atenderse primero.

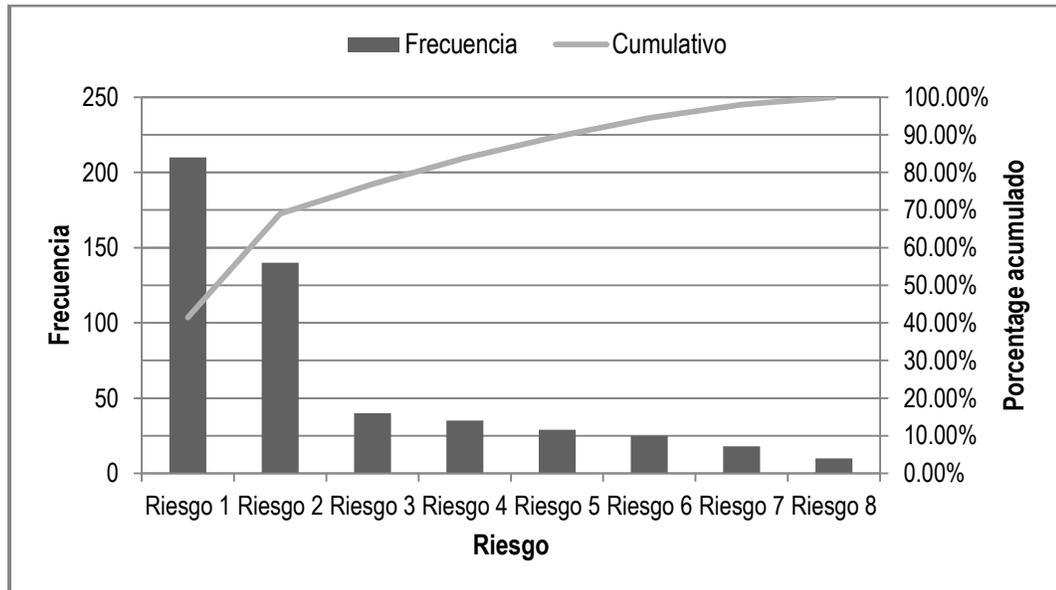


Figura 7-2: Ejemplo de un diagrama de Pareto

### 4. Cuadrícula de probabilidad e impacto (*Probability Impact Grid*)

Es una cuadrícula donde se evalúan los riesgos para establecer la probabilidad de ocurrencia y del impacto potencia en los objetivos del proyecto. Generalmente, se asigna una clasificación numérica tanto para la probabilidad como para el impacto en forma independiente. Ambos valores se multiplican para obtener una puntuación de la gravedad del riesgo, lo cual se puede utilizar para priorizar los riesgos.

Por ejemplo, la puntuación de la gravedad del riesgo para un riesgo con una probabilidad del 50 % y un índice de impacto de 0.6 se calcularía de la siguiente forma:

$$0.5 \text{ (Probabilidad)} \times 0.6 \text{ (Impacto)} = 0.3$$

Los esquemas de calificación implementados se determinan dentro de la organización o para el proyecto. Por lo general, se utiliza una escala decimal, del cero al uno, donde un índice de probabilidad de 0.5 indicaría un 50 % de posibilidad. Otras opciones incluyen una escala del uno al diez, o alta (3), mediana (2) y baja (1).

La figura 7-3 muestra el uso de la escala decimal. Cada riesgo se califica con base a su probabilidad de ocurrencia e impacto en una escala objetiva.

Matriz de probabilidad e impacto							
		Amenazas			Oportunidades		
Probabilidad	0.90	0.09	0.27	0.72	0.72	0.27	0.09
	0.75	0.075	0.225	0.60	0.60	0.225	0.075
	0.50	0.05	0.15	0.40	0.40	0.15	0.05
	0.30	0.03	0.09	0.24	0.24	0.09	0.03
	0.10	0.01	0.03	0.08	0.08	0.03	0.01
		Bajo 0.1	Mediano 0.3	Alto 0.8	Bajo 0.8	Mediano 0.3	Alto 0.1

**Impacto**

Valor bajo de PI    
  Valor moderado de PI    
  Valor alto de PI

Figura 7-3: Muestra de matriz de probabilidad e impacto

El método de asignar valores de probabilidad e impacto a los riesgos varía dependiendo del proyecto y de la cantidad de riesgos que se evalúan, así como los procesos y procedimientos organizacionales existentes. Sin embargo, si se aplica la sencilla fórmula:  $P \times I$ , la gravedad del riesgo se puede calcular en una escala numérica o categórica.

## 5. Valor monetario esperado

El valor monetario del riesgo está basado en su valor monetario esperado (EMV, por sus siglas en inglés). Dicho valor se calcula multiplicando el impacto monetario por la probabilidad de riesgo, según la aproximación del cliente.

Valor monetario esperado = Impacto del riesgo (en dólares) x Probabilidad del riesgo (porcentaje)

Por ejemplo, un riesgo con un pacto negativo estimado de 1,000 dólares y un 50 % de probabilidad de ocurrencia resultaría en el siguiente valor monetario esperado:

$$\text{Valor monetario esperado} = \$1,000 \times 0.50 = \$500$$

### 7.4.3 Priorización de riesgos

Scrum permite una rápida identificación y evaluación de riesgos. Los riesgos identificados se toman en cuenta al momento de crear la lista priorizada de pendientes del producto durante el proceso de su creación, o bien, cuando se actualiza dicha lista en el proceso de su mantenimiento; de tal forma que una lista priorizada de pendientes del producto pudiera también conocerse como: lista priorizada de pendientes del riesgo ajustado.

Los riesgos se pueden identificar y evaluar con base en cualquier técnica de identificación y evaluación de riesgos que se mencionan anteriormente.

En los procesos de *Creación de la lista priorizada de pendientes del producto* y *Mantenimiento de la lista priorizada de pendientes del producto*, las historias de usuario priorizadas de las listas priorizadas, así como las listas de riesgos priorizadas se combinan para crear una lista actualizada de pendientes del producto, misma que incluye los riesgos que fueron identificados.

Pasos para actualizar la lista priorizada de pendientes del producto con riesgos identificados:

1. Crear una lista de riesgos priorizados (por ejemplo: los riesgos se pueden priorizar por valor utilizando la técnica de valor monetario esperado).
2. Seleccionar aquellos riesgos identificados que pudieran mitigarse; y para cuáles el equipo decide tomar acción específica de riesgos durante el sprint a fin de mitigar tales riesgos.
3. Crear una lista de historias de usuarios en la lista priorizada de pendientes del producto, mismas que se priorizan por valor (por ejemplo: el valor de cada historia de usuario se puede evaluar con base en su retorno sobre la inversión esperado).
4. Combinar las listas de los pasos 2 y 3 y priorizarlos por valor para llegar a la lista priorizada de pendientes del producto.

La figura 7-4 muestra el proceso de priorización de riesgos:

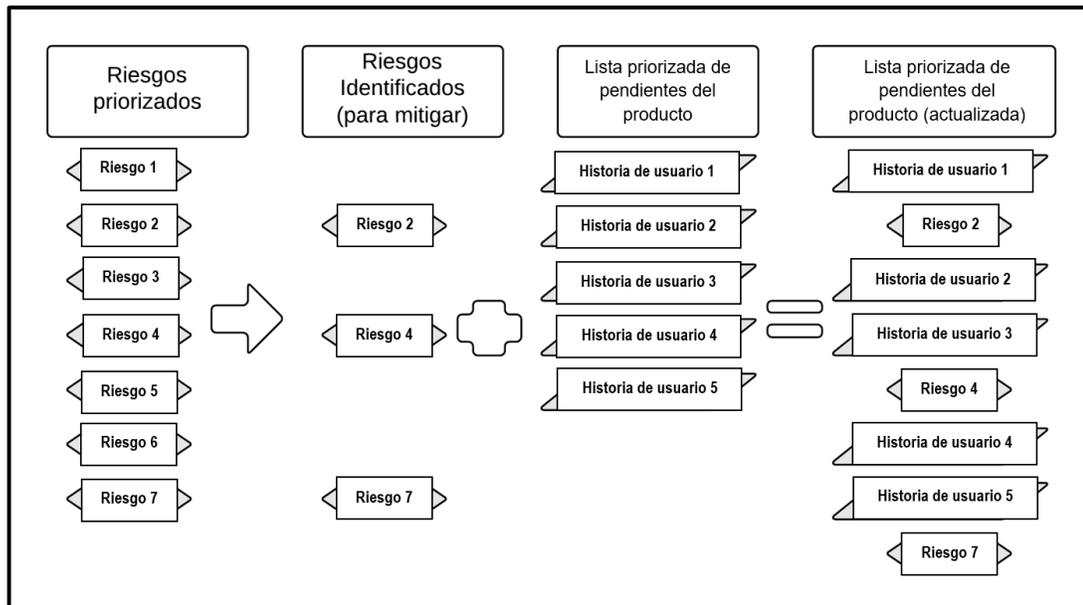


Figura 7-4: Proceso de priorización de riesgos

#### 7.4.4 Mitigación de riesgos

La respuesta a cada riesgo dependerá de la probabilidad y el impacto del mismo. Sin embargo, la naturaleza iterativa de Scrum —con sus ciclos rápidos de respuesta y retroalimentación—, permite que las fallas se detecten de forma temprana; por lo tanto, hablando en términos prácticos, tiene una función de mitigación natural construida adentro del sistema.

Un riesgo puede ser mitigado mediante la implementación de una serie de respuestas. En la mayoría de los casos, las respuestas son proactivas o reactivas. En el caso de un riesgo, se puede formular un plan B, que se puede utilizar como una alternativa en caso de que el riesgo se materialice; en este caso, el plan B es una respuesta reactiva. En ocasiones, los riesgos se aceptan y son un ejemplo de una respuesta al riesgo que no es ni preventiva ni reactiva. Los riesgos se aceptan debido a varias razones, como en una situación en la que la probabilidad o el impacto de riesgo son muy bajos para una respuesta. La aceptación también puede ser el caso en una situación en la que la aprehensión de riesgos secundarios puede disuadir al propietario del producto de tomar cualquier acción. El esfuerzo realizado por el propietario del producto para reducir la probabilidad del riesgo o del impacto (o ambos), es un ejemplo de una respuesta proactiva a la mitigación de riesgo.

Una vez que los riesgos identificados se incluyen como parte de la lista priorizada de pendientes del producto (véase la figura 7-4), varios riesgos se mitigan durante el proceso de creación de entregables donde se completan las tareas relacionadas a las historias de usuario definidas en el proceso de la lista priorizada de pendientes del producto.

En Scrum, la responsabilidad del riesgo es claramente del propietario del producto para la gestión de los riesgos relacionados a los aspectos empresariales; la responsabilidad es también del equipo Scrum para la implementación de respuestas al riesgo durante el curso de un sprint. El cuerpo de asesoramiento de Scrum puede consultarse para pedir orientación sobre la forma de implementar la respuesta a los riesgos y ver si las acciones se alinean con las directrices de la organización en su conjunto. El Scrum Master mantiene una estrecha vigilancia sobre los riesgos potenciales que pudieran afectar el proyecto y mantiene informado al propietario del producto y al equipo Scrum.

## 7.4.5 Comunicación de riesgo

Debido a que los socios tienen un interés en el proyecto, es importante comunicarse con ellos sobre asuntos relacionados a los riesgos. La información proporcionada a los socios relacionada con el riesgo debe incluir el impacto potencial, así como los planes para hacerle frente a cada riesgo. Esta comunicación siempre está en curso y debe ocurrir a la par de los cuatro pasos secuenciales discutidos hasta ahora: identificación, evaluación, priorización y mitigación de riesgos. El equipo Scrum pudiera también discutir con el Scrum Master los riesgos específicos relacionados a sus tareas durante las reuniones diarias de pie. El propietario del producto es responsable de la priorización de riesgos y de la comunicación de la lista priorizada de pendientes del producto al equipo Scrum.

Una herramienta importante que se puede utilizar para comunicar la información relacionada a los riesgos es la gráfica de trabajo pendiente del riesgo, conocida en inglés como: *Risk Burndown Chart*.

### 7.4.5.1 Gráfica de trabajo pendiente el riesgo

La gestión de riesgos es esencial para garantizar la creación de valor; por lo tanto, las actividades de gestión de riesgos se llevan a cabo durante todo el ciclo de vida del proyecto y no sólo durante el inicio del mismo.

Cada riesgo se puede evaluar usando diferentes herramientas de evaluación de riesgos. Sin embargo, la herramienta preferida para evaluación riesgos y crear una gráfica de trabajo pendiente del riesgo (del inglés *Risk Burndown Chart*) es el Valor Monetario Esperado (VME), tal como se describe en la sección 7.4.2.1.

La información recopilada durante la evaluación de riesgos se puede utilizar para crear una gráfica de trabajo pendiente del riesgo. Esto representa la severidad del riesgo del proyecto acumulativo en el tiempo. Las probabilidades de los diversos riesgos se trazan una sobre otra para mostrar el riesgo acumulativo en el eje vertical. La identificación y evaluación inicial de los riesgos en el proyecto, así como la creación de la gráfica de pendientes del riesgo se realizan al inicio. Después, en intervalos de tiempo predeterminados, los nuevos riesgos pueden ser identificados y evaluados mientras que los riesgos restantes deben ser reevaluados en la gráfica según sea el caso. Un momento apropiado para hacerlo es durante la reunión de planificación del sprint. El seguimiento de los riesgos de esta forma le permite al equipo reconocer tendencias en exposición de riesgo y tomar acciones adecuadas conforme sea necesario.

La figura 7-5 muestra un ejemplo de una gráfica de trabajo pendiente del riesgo (*Risk Burndown Chart*)

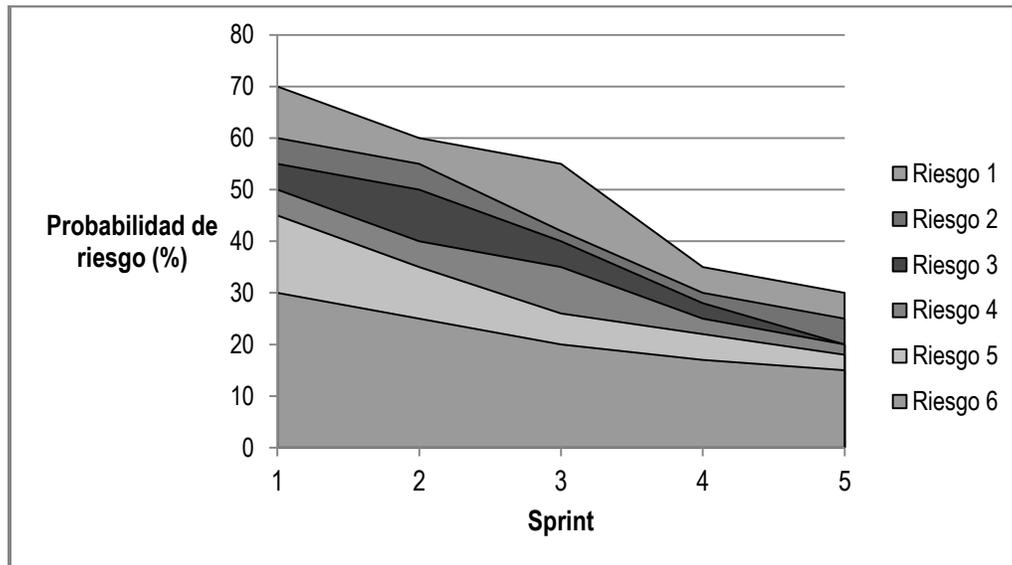


Figura 7-5: Ejemplo de una gráfica de trabajo pendiente del riesgo (*Risk Burndown Chart*)

## 7.5 Minimizar riesgos por medio de Scrum

Al ser un proceso ágil e iterativo, el marco de Scrum minimiza inherentemente el riesgo. Las siguientes prácticas de Scrum facilitan la gestión efectiva del riesgo:

### 1. La flexibilidad reduce el riesgo relacionado al entorno empresarial

El riesgo se reduce en gran medida en Scrum debido a la flexibilidad en la adición o modificación de los requisitos en cualquier momento del ciclo de vida del proyecto. Esto le permite a la organización responder a las amenazas u oportunidades en el entorno empresarial, así como a las necesidades imprevistas cada vez que surjan, por lo general con un bajo costo de la gestión de tales riesgos.

### 2. La retroalimentación constante reduce el riesgo relacionado a las expectativas

Al ser iterativo, el marco de Scrum proporciona amplias oportunidades para obtener información y establecer expectativas en todo el ciclo de vida del proyecto. Esto asegura que los socios del proyecto, así como el equipo, no sean tomados por sorpresa debido a requisitos mal comunicados.

### 3. La propiedad del equipo reduce la estimación de riesgo

El equipo Scrum hace estimaciones y se hace responsable de los elementos de la lista de pendientes del Sprint, lo cual conduce a una estimación más precisa y a la entrega oportuna de los incrementos del producto.

#### 4. La transparencia reduce el riesgo de no detección

El principio de la transparencia en Scrum, en torno al cual se construye el marco, asegura que los riesgos se detecten y se comuniquen oportunamente, lo cual conduce a un mejor manejo y mitigación de riesgos. Por otra parte, al llevar a cabo reuniones de Scrum de Scrum, los impedimentos que un equipo enfrenta en la actualidad pueden considerarse como riesgos para otros equipos Scrum a futuro. Esto debe reconocerse en el registro actualizado de impedimentos (*Updated Impediments Log*).

#### 5. La entrega iterativa reduce el riesgo de inversión

La entrega continua de valor a lo largo del ciclo de vida del proyecto Scrum, como entregables potencialmente listos para la entrega, se crean después de cada sprint, reduciendo así el riesgo de la inversión para el cliente.

## 7.6 Riesgos en carteras y programas

Mientras que algunos riesgos están específicamente relacionados con proyectos individuales, otros pueden tener su origen en los programas y carteras, y generalmente serán administrados allí mismo. Sin embargo, los riesgos relacionados con una cartera o programa también tendrán un impacto en los proyectos que forman parte de la respectiva cartera o programa. Durante la evaluación de riesgos en carteras y programas —si se determina que el riesgo puede afectar un proyecto individual—, la información relevante sobre el riesgo debe ser comunicada al propietario del producto y al equipo Scrum.

Dependiendo de la gravedad o de la prioridad, cuando el equipo del programa o la cartera comunica el riesgo que habrá de tener un impacto en un proyecto individual, el equipo Scrum tal vez tenga que detenerse y volver a planificar el actual Sprint para atender el riesgo. Para los riesgos de menor urgencia, el equipo puede continuar con el actual Sprint y atender el riesgo en un sprint a futuro.

### 7.6.1 En la cartera

1. Cuando se identifican los riesgos en la cartera, el propietario del producto de la cartera tenderá que capturarlos y evaluar la proximidad, probabilidad e impacto de cada riesgo identificado a fin de priorizarlo y determinar la respuesta adecuada para la cartera.
2. El propietario del producto de la cartera tendrá también que comunicar los riesgos a los socios relevantes, a los equipos del programa y del proyecto. En algunos casos, el equipo de la cartera pudiera tener que asumir la responsabilidad de riesgos específicos.

## 7.6.2 En los programas

1. Cuando se identifican los riesgos en el programa, el propietario del producto del programa debe ingresarlo en lista de pendientes del producto del riesgo ajustado, evaluar la proximidad, probabilidad e impacto de cada riesgo identificado a fin de priorizarlo y determinar la respuesta adecuada para los programas.
2. El propietario del producto del programa tendrá también que comunicar los riesgos a los socios relevantes y a los equipos del proyecto. En algunos casos, el equipo de la cartera pudiera tener que asumir la responsabilidad de riesgos específicos.

La figura 7-6 muestra cómo los riesgos se pueden administrar dentro del flujo de Scrum tanto en las carteras como en los programas.

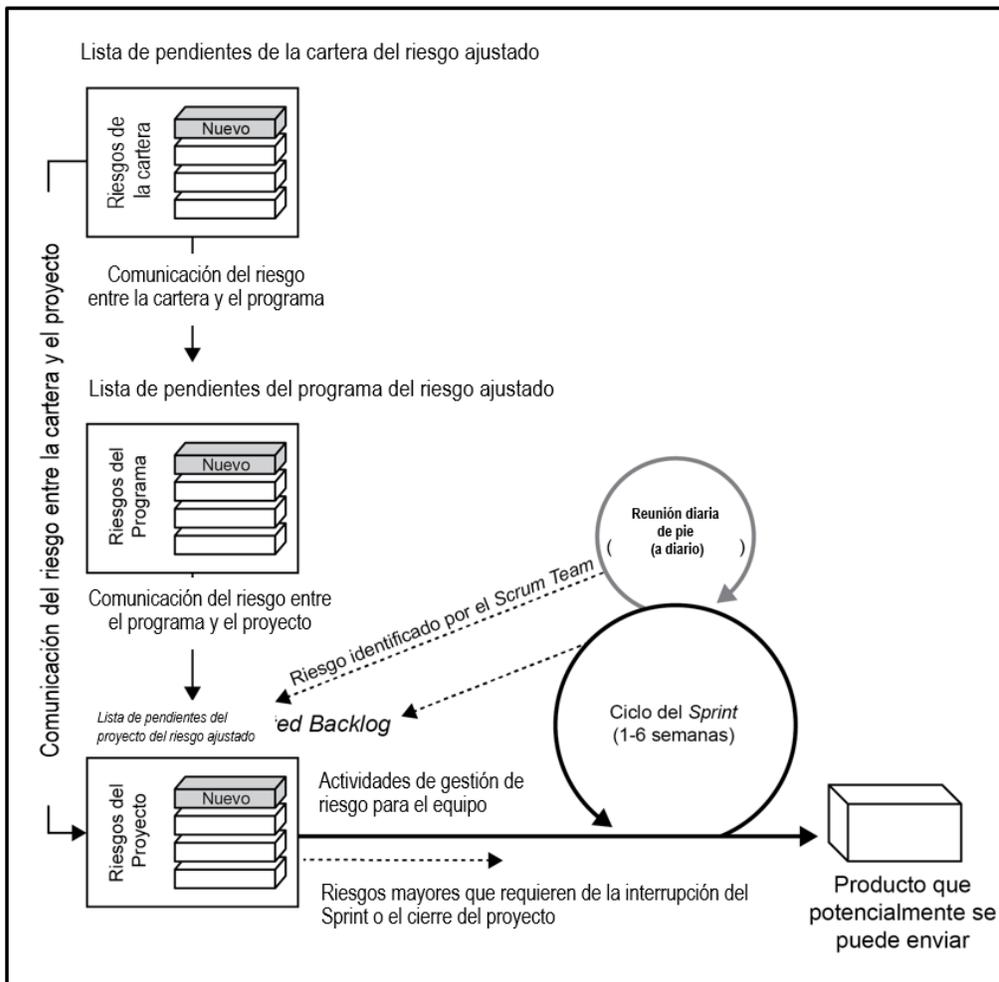


Figura 7-6: Manejo de riesgos en carteras y programas

## 7.7 Resumen de responsabilidades

En Scrum, las actividades de gestión de riesgos se dividen entre varios roles donde algunas de las responsabilidades recaen sobre todos dentro del equipo Scrum y donde el Scrum Master facilita el proceso.

Rol	Responsabilidades
Cuerpo de asesoramiento de Scrum	<ul style="list-style-type: none"> <li>• Brinda una guía general para el procedimiento de gestión de riesgos para que se siga durante todo el proyecto.</li> </ul>
Propietario del producto de la cartera	<ul style="list-style-type: none"> <li>• Captura y evalúa los riesgos de la cartera</li> <li>• Prioriza y comunica los riesgos a los socios relevantes, así como a los equipos del programa y del proyecto.</li> </ul>
Scrum master de la cartera	<ul style="list-style-type: none"> <li>• Facilita la identificación, evaluación y comunicación de los riesgos de la cartera.</li> </ul>
Propietario del producto del programa	<ul style="list-style-type: none"> <li>• Captura y evalúa riesgos de los programas</li> <li>• Prioriza y comunica los riesgos a los socios relevantes y equipos del proyecto.</li> </ul>
Scrum Master del programa	<ul style="list-style-type: none"> <li>• Facilita la identificación, evaluación y comunicación de los riesgos de los programas</li> </ul>
Socio(s)	<ul style="list-style-type: none"> <li>• Interactúa con el equipo principal de Scrum para brindarle información sobre la gestión de riesgos que afectan el logro de los resultados esperados, así como los beneficios del proyecto.</li> </ul>
Propietario del producto	<ul style="list-style-type: none"> <li>• Captura y evalúa riesgos del proyecto</li> <li>• Prioriza y comunica los riesgos a los socios relevantes y equipos de la cartera.</li> <li>• Se asegura de que los niveles de riesgo del proyecto estén dentro de los límites aceptables.</li> </ul>
Scrum Master	<ul style="list-style-type: none"> <li>• Le facilita al equipo Scrum la identificación y escalamiento de riesgos.</li> </ul>
Equipo Scrum	<ul style="list-style-type: none"> <li>• Identifica los riesgos durante el desarrollo del producto durante el proceso de creación de entregables.</li> <li>• Implementa actividades de gestión de riesgo tal como lo recomiende el propietario del producto.</li> </ul>

Tabla 7-1: Resumen de las responsabilidades pertinentes a los riesgos

## 7.8 Scrum vs Gestión tradicional de proyectos

Scrum y la mayoría de los métodos tradicionales de gestión de proyecto definen los “riesgos” como ‘evento(s) inciertos que pudieran afectar de forma positiva o negativa los logros de los objetivos del proyecto’. Asimismo, los riesgos se identifican, se atienden, se planifica y se comunican continuamente.

En los modelos tradicionales de gestión de proyectos, se hace énfasis en la planificación inicial detallada para identificar, evaluar y determinar las respuestas de riesgo para todos los riesgos del proyecto. Durante la ejecución del proyecto, cualquier miembro del equipo del proyecto puede identificar los riesgos y el gerente del proyecto o la dependencia encargada de gestión de proyectos o el personal de apoyo a los proyectos puede actualizarlos en la lista de riesgos o en el registro de riesgos. El gerente del proyecto trata de controlar todos los riesgos y por lo general identifica a individuos específicos en el equipo para que se hagan cargo de diferentes aspectos de riesgos.

En Scrum, cualquier miembro del equipo puede identificar riesgos y el propietario del producto puede actualizar los riesgos identificados en la lista priorizada de pendientes del producto del riesgo ajustado. Los principios de Scrum sobre el control de proceso empírico y de desarrollo iterativo permiten que el equipo Scrum pueda identificar constantemente los riesgos y sumarlos a la lista priorizada de pendientes del producto donde tales riesgos se pueden priorizar con otras historias de usuarios existentes en la lista antes mencionada, a efectos de mitigarlos en futuros sprints. El equipo Scrum tiene responsabilidades colectivas para la gestión de todos los riesgos para el Sprint.

## 8. INICIO

Este capítulo incluye los procesos relacionados con el inicio de un proyecto: *creación de la visión del proyecto, identificación del Scrum Master y el(los) socio(s), formación de un equipo Scrum, desarrollo de épica(s), creación de la lista priorizada de pendientes del producto, y realizar el plan de lanzamiento.*

El inicio, tal como se define en *Una guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Cartera, programas y/o proyectos de cualquier industria
- Productos, servicios o cualquier otro resultado que se entregará a los socios
- Proyectos de cualquier tamaño y complejidad

El término "producto" en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria —desde pequeños proyectos o equipos con tan sólo seis miembros del equipo, hasta proyectos grandes y complejos que cuentan con cientos de miembros por equipo.

A fin de facilitar la mejor aplicación del marco de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como "obligatorias" u "opcionales". Las entradas, herramientas y salidas indicadas por asteriscos (\*) son obligatorias, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el equipo Scrum y aquellas personas que recién inician el aprendizaje sobre el marco Scrum y sus procesos se centren principalmente en las entradas, herramientas y salidas obligatorias; mientras que el propietario del producto, los Scrum Masters, y otros practicantes de Scrum con más experiencia sobre éste, se esfuerzan por alcanzar un conocimiento más profundo de la información de todo el capítulo. También es importante darse cuenta de que, aunque todos los procesos se definen de forma única en la *Guía SBOK™*, no se llevan a cabo de forma secuencial o por separado necesariamente. A veces, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está redactado desde la perspectiva de un equipo Scrum que está trabajando en un Sprint para producir entregables como parte de un proyecto más amplio. Sin embargo, la información que se describe es igualmente aplicable a proyectos completos, programas y carteras. Información adicional relacionada con el uso de Scrum para proyectos, programas y carteras está disponible desde el capítulo 2 hasta el 7, los cuales cubren los principios y aspectos de Scrum.

La figura 8-1 proporciona una visión general de los procesos de la fase de Inicio, que son los siguientes:

**8.1 Creación de la visión del proyecto**—En este proceso, el caso de negocio del proyecto se revisa para crear una declaración de la visión del proyecto que servirá de inspiración y proporcionará un enfoque de todo el proyecto. En este proceso se identifica al propietario del producto.

**8.2 Identificación del Scrum Master y el(los) socio(s)**—En este proceso, se identifica al Scrum Master y a los socios utilizando criterios de selección específicos.

**8.3 Formación de un equipo Scrum**—En este proceso, se seleccionan a los miembros del equipo Scrum. Normalmente, el propietario del producto es el responsable principal de la selección de los miembros del equipo, pero a menudo lo hace en colaboración con el Scrum Master.

**8.4 Desarrollo de épica(s)**—En este proceso, la declaración de la visión del proyecto sirve como la base para el desarrollo de épicas. Las reuniones de grupo de usuarios pueden llevarse a cabo para discutir la(las) épica(s) apropiada(s).

**8.5 Creación de la lista priorizada de pendientes del producto**—En este proceso, la épica(s) es refinada, detallada, y luego priorizada para crear la lista priorizada de pendientes del producto del proyecto. Lo que se conoce como criterios de terminado también se establecen en este punto.

**8.6 Realizar el plan de lanzamiento**—En este proceso, el equipo principal de Scrum revisa las historias de usuario en el lista priorizada de pendientes del producto para desarrollar un cronograma de planificación del lanzamiento, que es esencialmente un programa de implementación por fases que se puede compartir con los socios del proyecto. La duración de los sprints también se determina en este proceso.

8.1 Creación de la visión del proyecto	8.2 Identificación del Scrum Master y el(los) socio(s)	8.3 Formación del equipo Scrum
<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Caso de negocio del proyecto*</li> <li>2. Propietario del producto del programa</li> <li>3. Scrum Master del programa</li> <li>4. Socio(s) del programa</li> <li>5. Jefe propietario del producto</li> <li>6. Lista de pendientes del producto del programa</li> <li>7. Proyecto de prueba</li> <li>8. Prueba del concepto</li> <li>9. Visión de la empresa</li> <li>10. Misión de la empresa</li> <li>11. Estudio del mercado</li> <li>12. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reuniones de la visión del proyecto*</li> <li>2. Sesiones de diseño de aplicación conjunta (JAD)</li> <li>3. Análisis FODA (SWOT)</li> <li>4. Análisis de brecha</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Propietario del producto identificado*</li> <li>2. Declaración de la visión del proyecto*</li> <li>3. Acta constitutiva del proyecto</li> <li>4. Presupuesto del proyecto</li> </ol>	<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Propietario del producto*</li> <li>2. Declaración de la visión del proyecto*</li> <li>3. Propietario del producto del programa</li> <li>4. Scrum Master del programa</li> <li>5. Jefe propietario del producto</li> <li>6. Jefe Scrum Master</li> <li>7. Socio(s) del programa</li> <li>8. Requerimientos de las personas</li> <li>9. Disponibilidad y compromiso de prototipos</li> <li>10. Matriz de recursos organizacionales</li> <li>11. Matriz de las destrezas requeridas</li> <li>12. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Criterios de selección*</li> <li>2. Asesoramiento de expertos en recursos humanos</li> <li>3. Capacitación y costos de capacitación</li> <li>4. Costos de recursos</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Scrum Master Identificado*</li> <li>2. Socio(s) Identificado*</li> </ol>	<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Propietario del producto*</li> <li>2. Scrum Master*</li> <li>3. Declaración de la visión del proyecto*</li> <li>4. Jefe propietario del producto</li> <li>5. Requisitos de personal</li> <li>6. Disponibilidad y compromiso de los prototipos</li> <li>7. Matriz de recurso organizacional</li> <li>8. Matriz de las destrezas requeridas</li> <li>9. Requerimientos de recursos</li> <li>10. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Selección del equipo*</li> <li>2. Asesoramiento de expertos en recursos humanos</li> <li>3. Costos del personal</li> <li>4. Capacitación y costos de capacitación</li> <li>5. Costos de recursos</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Equipo Scrum identificado*</li> <li>2. Substitutos</li> <li>3. Plan de colaboración</li> <li>4. Plan para la formación del equipo</li> </ol>
<p><b>8.4 Desarrollo de épica(s)</b></p> <p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Equipo Principal de Scrum*</li> <li>2. Declaración de la visión del proyecto*</li> <li>3. Socio(s)</li> <li>4. Lista de pendientes del producto del programa</li> <li>5. Solicitudes de cambios aprobadas</li> <li>6. Solicitudes de cambios no aprobadas</li> <li>7. Riesgos de la cartera y el programa</li> <li>8. Leyes y regulaciones</li> <li>9. Contratos aplicables</li> <li>10. Información de proyectos previos</li> <li>11. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reuniones del grupo de usuarios*</li> <li>2. Talleres de historias de usuario</li> <li>3. Reuniones del grupo de enfoque</li> <li>4. Entrevistas al usuario o cliente</li> <li>5. Cuestionarios</li> <li>6. Técnicas de identificación de riesgos</li> <li>7. Experiencia del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Épica(s)*</li> <li>2. Prototipos*</li> <li>3. Cambios aprobados</li> <li>4. Riesgos identificados</li> </ol>	<p><b>8.5 Creación de la lista priorizada de pendientes del producto</b></p> <p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Equipo principal de Scrum*</li> <li>2. Épica(s)*</li> <li>3. Prototipos*</li> <li>4. Socio(s)</li> <li>5. Declaración de la visión del proyecto</li> <li>6. Lista de pendientes del producto del programa</li> <li>7. Requerimientos del negocio</li> <li>8. Solicitudes de cambios aprobadas</li> <li>9. Riesgos identificados</li> <li>10. Contratos Aplicables</li> <li>11. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Métodos de priorización de historias del usuario*</li> <li>2. Talleres de historias del usuario</li> <li>3. Planificación de valor</li> <li>4. Técnicas de evaluación del riesgo</li> <li>5. Estimación del valor del proyecto</li> <li>6. Métodos de estimación de historias de usuario</li> <li>7. Experiencia del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Lista priorizada de pendientes del producto*</li> <li>2. Criterios de terminado*</li> </ol>	<p><b>8.6 Realizar la planificación del lanzamiento</b></p> <p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Equipo principal de Scrum*</li> <li>2. Socio(s)*</li> <li>3. Declaración de la visión del proyecto*</li> <li>4. Lista priorizada de pendientes del producto*</li> <li>5. Criterios de terminado*</li> <li>6. Propietario del producto del programa</li> <li>7. Scrum Master del programa</li> <li>8. Jefe propietario del producto</li> <li>9. Lista de pendientes del producto del programa</li> <li>10. Requerimientos del negocio</li> <li>11. Calendario de días festivos</li> <li>12. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Sesiones de planificación del lanzamiento*</li> <li>2. Métodos de priorización del lanzamiento*</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Cronograma de planificación del lanzamiento*</li> <li>2. Duración del Sprint*</li> <li>3. Clientes meta para el lanzamiento</li> <li>4. Lista priorizada de pendientes del producto refinada</li> </ol>

Figura 8-1: Información general sobre el inicio

Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

La figura 8-2 muestra las entradas obligatorias, herramientas y salidas para los procesos en la fase de inicio.

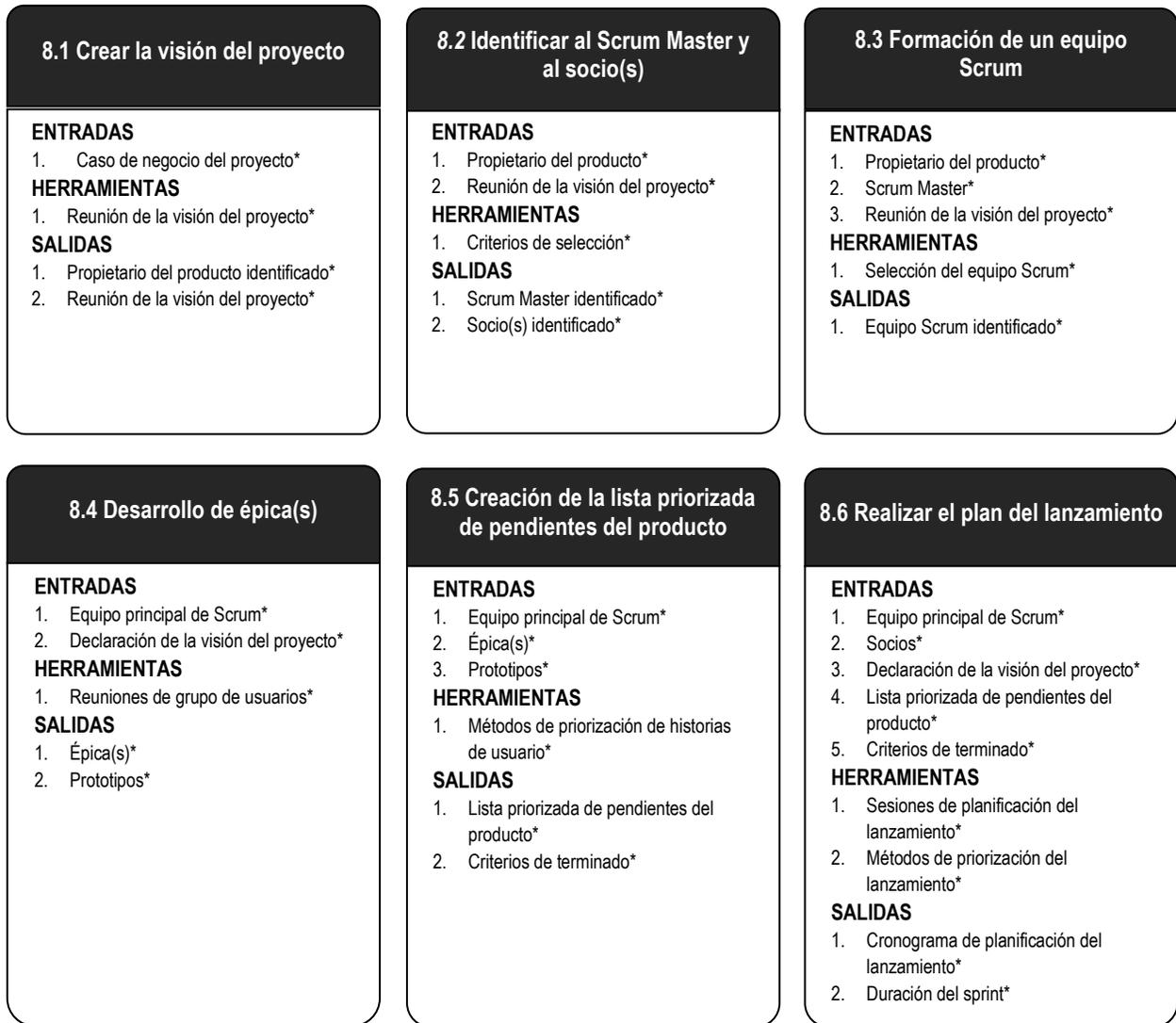
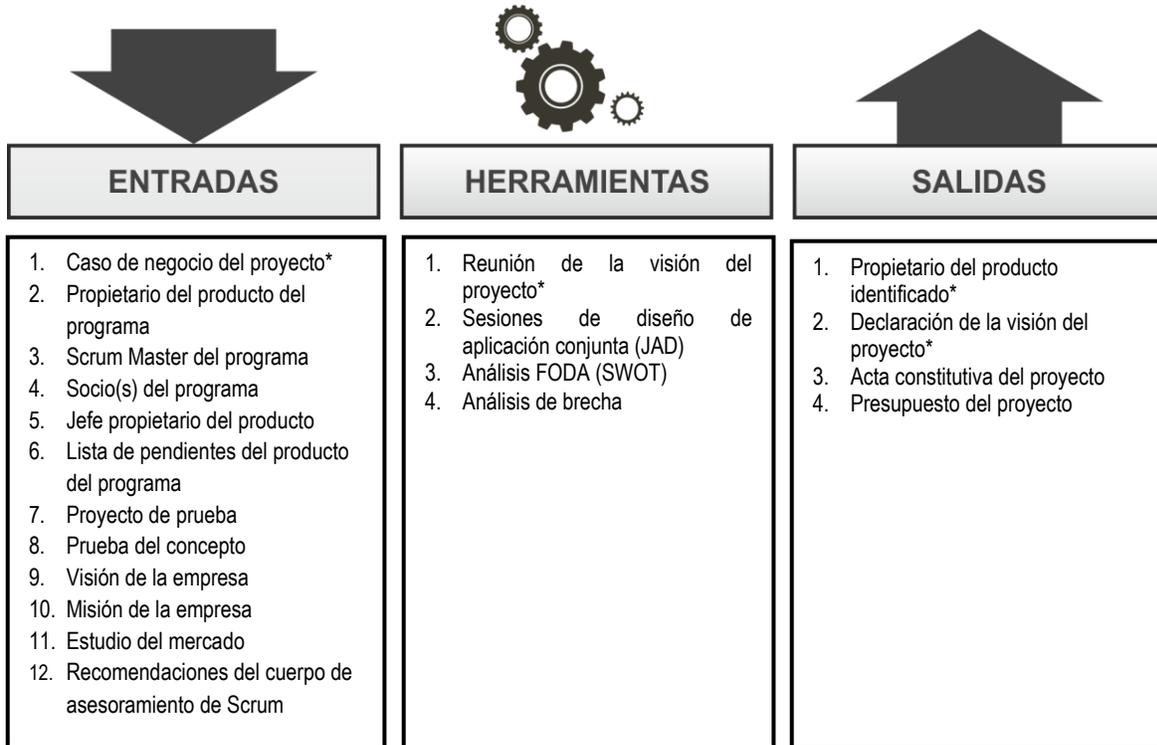


Figura 8-2: Resumen sobre el inicio (esenciales)

## 8.1 Creación de la visión del proyecto

La Figura 8-3 muestra todas las entradas, herramientas y salidas para el proceso de *creación de la visión del proyecto*.



**Figura 8-3: Creación de la visión del proyecto – Entradas, herramientas y salidas**

*Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.*

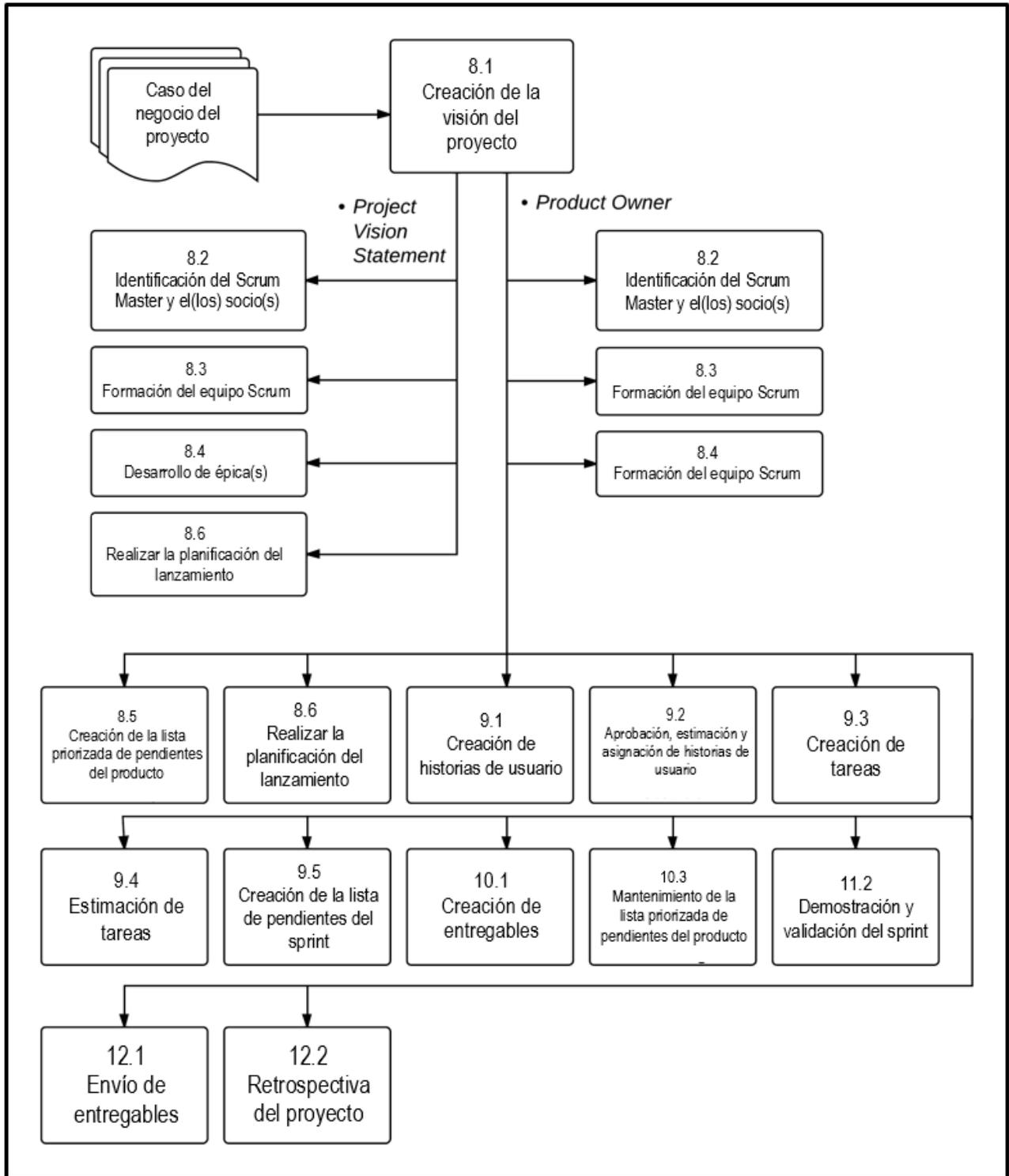


Figura 8-4: Diagrama del flujo de datos de la creación de la visión del proyecto

## 8.1.1 Entradas

### 8.1.1.1 Caso de negocio del proyecto\*

Un caso de negocio puede ser un documento bien estructurado o simplemente una declaración verbal que expresa la razón para iniciar un proyecto. Puede ser formal y detallado, o informal y breve. Independientemente del formato, a menudo este incluye información sustancial sobre los antecedentes del proyecto, los objetivos del negocio y los resultados deseados, un reporte de análisis FODA y de brecha, una lista de los riesgos identificados, y las estimaciones de tiempo, esfuerzo y costo.

El proyecto se inicia con la presentación del caso de negocio del proyecto. Un caso de negocio se le presenta a los socios y patrocinadores. De esta manera, los socios comprenden los beneficios de negocio esperados de tal proyecto, y los patrocinadores confirman que van a proporcionar los recursos financieros para el proyecto.

### 8.1.1.2 Propietario del producto del programa

El propietario del producto del programa es la persona responsable de maximizar el valor del negocio para un programa. Está a cargo de la articulación de los requisitos de los clientes y de mantener la justificación del negocio para el programa, puede proporcionar aportaciones valiosas sobre qué proyectos se deben implementar, y de qué manera, en un programa. El propietario del producto del programa también administra la lista de pendientes del producto del programa.

El propietario del producto del programa interactúa con el propietario del producto de la cartera para asegurar la alineación del programa con las metas y objetivos de la cartera. También tiene que ver con el nombramiento de los propietarios del producto para los proyectos individuales, y debe asegurar que la visión, los objetivos, los resultados y los lanzamientos de los proyectos individuales en el programa estén alineados con los del programa.

### 8.1.1.3 Scrum Master del programa

El Scrum Master del programa es un facilitador que asegura que todos los equipos del proyecto en el programa dispongan de un ambiente propicio para completar con éxito sus proyectos. El Scrum Master del programa guía, facilita y les enseña las prácticas de Scrum a todos los que participan en el programa; también sirve como guía para los Scrum Masters de los distintos proyectos; elimina los impedimentos de los diferentes equipos del proyecto; coordina con el cuerpo de asesoramiento de Scrum para definir los objetivos relacionados con la normativa de calidad, gobierno, seguridad y otros parámetros claves de la organización; y, asegura que los procesos de Scrum se estén siguiendo de manera efectiva a lo largo del programa.

El Scrum Master del programa debe interactuar con el Scrum Master de la cartera para asegurar la alineación del programa con las metas y objetivos de la cartera. También tiene que ver con el nombramiento de Scrum Masters para proyectos individuales y debe asegurar que la visión, los objetivos, los resultados, y los lanzamientos de los proyectos individuales en el programa estén alineados con los del programa.

#### **8.1.1.4 Socio(s) del programa**

El(los) socio(s) del programa es un término colectivo que incluye a los clientes, usuarios y patrocinadores para un programa. Estos influyen en todos los proyectos del programa durante todo el desarrollo del proyecto. El(los) socio(s) del programa también puede(n) ayudar a definir la visión del proyecto y a proporcionar orientación en relación con el valor del negocio.

El(los) socio(s) del programa debe(n) interactuar con los socios de la cartera para asegurar la alineación del programa con las metas y objetivos de la cartera. Ellos también están involucrados con el nombramiento del (los) socios(s) para proyectos individuales y se aseguran de que la visión, los objetivos, los resultados, y los lanzamientos de los proyectos individuales en el programa estén alineados con los del programa.

8

#### **8.1.1.5 Jefe propietario del producto**

En el caso de proyectos grandes con numerosos equipos Scrum, tener un jefe propietario del producto puede ser necesario. Este rol se encarga de coordinar el trabajo de los múltiples propietarios del producto. El jefe propietario del producto prepara y mantiene la lista priorizada de pendientes del producto para los proyectos grandes, usándolo para coordinar el trabajo a través de los propietarios del producto de los equipos Scrum. Los propietarios del producto, a su vez, gestionan sus respectivas partes de la lista priorizada de pendientes del producto.

El jefe propietario del producto también se comunica con el propietario del producto del programa para asegurar la alineación de los proyectos grandes con las metas y objetivos del programa.

#### **8.1.1.6 Lista de pendientes del producto del programa**

El propietario del producto del programa desarrolla la lista de pendientes del producto del programa que contiene una lista de prioridades de negocios de alto nivel, y los requisitos del proyecto escritos preferiblemente en forma de grandes elementos de la lista de pendientes del programa. Luego, estos son refinados por los propietarios del producto de los proyectos individuales, ya que crean y dan prioridad a pedidos pendientes del producto para sus proyectos. Esta lista priorizada de pendientes del producto tiene historias de usuario más pequeñas pero detalladas que pueden ser aprobadas, estimadas y asignadas por equipos Scrum.

El propietario del producto del programa mantiene de forma continua la lista de pendientes del producto del programa para garantizar que los nuevos requerimientos de negocio se añadan, y que los requisitos existentes estén debidamente documentados y priorizados.

Esto asegura que a los requisitos más valiosos en el cumplimiento de los objetivos del programa se les esté dando alta prioridad, y que los restantes reciban una prioridad más baja. La lista de pendientes del producto del programa creada presenta una imagen más grande de todos los proyectos que forman parte del programa. Por lo tanto, puede servir de orientación en relación con las metas del proyecto, el alcance, los objetivos y los beneficios esperados del negocio.

#### **8.1.1.7 Proyecto de prueba**

Si es factible, una demostración o prueba del proyecto a pequeña escala se pudiera llevar a cabo como un experimento para predecir y evaluar la viabilidad, tiempo, costo, riesgos, y posibles efectos del proyecto. Esto ayuda a evaluar el ambiente de la práctica y guía el diseño real del proyecto antes del inicio del proyecto a gran escala.

#### **8.1.1.8 Prueba de concepto**

La prueba del concepto demuestra y verifica que la idea alrededor del proyecto actual es potencialmente viable en la vida real. A menudo esto se hace en la forma de un prototipo diseñado para determinar la viabilidad técnica y financiera, ayudar a comprender los requerimientos, y asistir en la evaluación de las decisiones de diseño al principio del proceso. Sin embargo, la prueba del concepto no necesita representar necesariamente los verdaderos entregables del proyecto.

#### **8.1.1.9 Visión de la compañía**

La comprensión de la visión de la compañía ayuda a que el proyecto mantenga su enfoque en los objetivos de la organización y el futuro probable de la empresa. El propietario del producto puede guiarse por la visión de la compañía para crear la declaración de la visión del proyecto.

#### **8.1.1.10 Misión de la compañía**

La misión de la compañía ofrece un marco para la formulación de las estrategias de la empresa y orienta la toma de decisiones en general en la compañía. Dicha misión debe enmarcarse de tal manera que su cumplimiento ayude a la organización a llevar a cabo su misión.

#### **8.1.1.11 Estudio de mercado**

El estudio de mercado se refiere a la investigación organizada, la recopilación, la comparación y el análisis de datos relacionados con las preferencias de los clientes sobre los productos. Con frecuencia, esta incluye numerosos datos sobre las tendencias y la segmentación del mercado, así como los procesos de comercialización.

El estudio de mercado podría incluir también un estudio analítico de los competidores, lo que proporciona una mejor comprensión de las fortalezas y debilidades de los competidores, y puede ayudar a quienes toman decisiones a formular productos mejor posicionados.

### 8.1.1.12 Recomendaciones del cuerpo de asesoramiento de Scrum

El cuerpo de asesoramiento de Scrum es un rol opcional. Por lo general, se compone de un grupo de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de objetivos relacionados con la calidad, las regulaciones gubernamentales, la seguridad y otros parámetros claves de la organización. Estos objetivos guían la labor llevada a cabo por el propietario del producto, el Scrum Master, y el equipo Scrum. El cuerpo de asesoramiento de Scrum también ayuda a captar las mejores prácticas que deben utilizarse en todos los proyectos Scrum en la organización.

El cuerpo de asesoramiento de Scrum no toma decisiones relacionadas con el proyecto. Este actúa como una consultoría o una estructura de orientación para todos los niveles de la jerarquía la organización del proyecto—la cartera, programa y proyecto. Los equipos Scrum tienen la opción de pedirle recomendaciones al cuerpo de asesoramiento de Scrum.

Es importante asegurarse de que la visión del proyecto esté alineada con las recomendaciones proporcionadas por el cuerpo de asesoramiento de Scrum, y que los procesos cumplan con las normas y directrices establecidas por el cuerpo.

## 8.1.2 Herramientas

### 8.1.2.1 Reunión de la visión del proyecto\*

La reunión de la visión del proyecto es una reunión con el(los) socio(s) del programa, el propietario del producto del programa, el Scrum Master del programa, y el jefe propietario del producto. Esta ayuda a identificar el contexto empresarial, los requerimientos de negocio, y las expectativas de los socios a fin de desarrollar una declaración de la visión del proyecto eficaz. Scrum cree en la participación y colaboración cercana con todos los representantes de las empresas para obtener su sentido de compromiso con el proyecto, y para ofrecer un valor más significativo.

### 8.1.2.2 Sesiones de diseño de aplicación conjunta (JAD)

La sesión de diseño de aplicación conjunta (JAD por sus siglas en inglés) es una técnica de recopilación de requisitos. Se trata de un taller impartido y altamente estructurado que acelera el proceso de *creación de la visión del proyecto*, ya que permite al(los) socio(s) y a otras personas que toman decisiones llegar a un consenso sobre el alcance, los objetivos, y otras especificaciones del proyecto.

Esta técnica consiste en métodos para aumentar la participación del usuario, lo que acelera el desarrollo y la mejora de las especificaciones. El(los) socios(s) relevante(s) del programa, el propietario del producto del

programa, el Scrum Master del programa y el jefe propietario del producto podrían reunirse para delinear y analizar los resultados de negocio deseados, y visualizar su visión para el proyecto Scrum.

### 8.1.2.3 Análisis FODA

El FODA (SWOT en inglés) es un enfoque estructurado para la planificación que ayuda a evaluar las Fortalezas, Oportunidades, Debilidades y Amenazas relacionadas con un proyecto. Este tipo de análisis ayuda a identificar tanto los factores internos como los externos que podrían afectar el proyecto. Las fortalezas y debilidades son factores internos, mientras que las oportunidades y amenazas son factores externos. La identificación de estos factores ayuda a los socios y a aquellos que toman decisiones a finalizar los procesos, las herramientas y las técnicas que se utilizarán para lograr los objetivos del proyecto. La realización de un FODA permite la identificación temprana de las prioridades, los cambios potenciales, y los riesgos.

### 8.1.2.4 Análisis de brecha

El análisis de brecha es una técnica que se utiliza para comparar el estado actual con el estado deseado. En una organización, esto involucra determinar y documentar la diferencia entre las capacidades actuales del negocio y el conjunto final de capacidades deseado. Normalmente, se inicia un proyecto para que una organización alcance una situación deseada, por lo que llevar a cabo un análisis de brecha ayudaría a quienes toman decisiones a determinar la necesidad de un proyecto.

Los principales pasos a seguir en el análisis de brecha se presentan en la Figura 8-5.

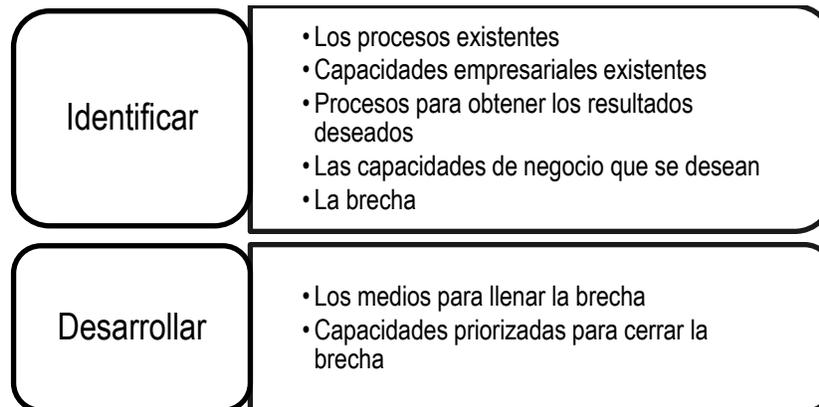


Figura 8-5: El proceso de análisis de brecha

## 8.1.3 Salidas

### 8.1.3.1 Propietario del producto identificado\*

Uno de los resultados de este proceso es identificar al propietario del producto. El propietario del producto es la persona responsable de lograr el valor máximo empresarial para el proyecto. También está a cargo de la articulación de requerimientos por parte de los clientes y de mantener la justificación del negocio para el proyecto. El propietario del producto representa la voz del cliente.

Cada equipo Scrum tendrá un propietario del producto designado. Un pequeño proyecto puede tener sólo un propietario del producto, mientras que los proyectos más grandes pueden tener varios. Estos propietarios del producto son responsables de la gestión de sus secciones de la lista priorizada de pendientes del producto. Ellos escriben las historias de usuario y gestionan el mantenimiento de la lista priorizada de pendientes del producto.

El papel del propietario del producto se describe en más detalle en la sección 3.4.

### 8.1.3.2 Reunión de la visión del proyecto\*

El resultado clave del proceso de creación de la visión del proyecto es una reunión de la visión del proyecto bien estructurada. Una buena visión del proyecto explica la necesidad del negocio, y qué es lo que el proyecto tiene como objetivo satisfacer, en lugar de cómo se va a satisfacer la necesidad.

La declaración de la visión del proyecto no debería ser muy específica y debe ser flexible. Es posible que el conocimiento actual sobre el proyecto esté basado en suposiciones que luego vayan a cambiar conforme avanza el proyecto, por lo que es importante que la visión del proyecto sea lo suficientemente flexible como para adaptarse a estos cambios. La visión del proyecto debe centrarse en el problema y no en la solución.

*Ejemplo:*

VMFoods, una cadena de supermercados, quiere expandirse con un portal de comercio electrónico en línea y se ha comunicado con su empresa para crear el producto.

Visión del proyecto: Desarrollar una herramienta fácil de usar y un canal de ventas en línea para VMFoods que sea estéticamente agradable.

### 8.1.3.3 Acta constitutiva del proyecto

Un acta constitutiva del proyecto es una declaración oficial de los objetivos y resultados deseados del proyecto. En varias organizaciones, el acta constitutiva del proyecto es el documento que autoriza el proyecto oficialmente y formalmente, dándole al equipo la autoridad por escrito para comenzar el proyecto.

### 8.1.3.4 Presupuesto del proyecto

El presupuesto del proyecto es un documento financiero que incluye el costo del personal, materiales y otros gastos relacionados en un proyecto. Por lo general, dicho presupuesto es firmado por el(los) patrocinador(es) para asegurar que haya suficientes fondos disponibles. Una vez que se autoriza, el propietario del producto y el Scrum Master estarán involucrados regularmente con la gestión del presupuesto del proyecto para garantizar que el personal y los recursos necesarios para las actividades del proyecto estén disponibles.

## 8.2 Identificar al Scrum Master y al socio(s)

La Figura 8-6 muestra todas las entradas, las herramientas y las salidas para el proceso de *Identificación del Scrum Master y el(los) socio(s)*

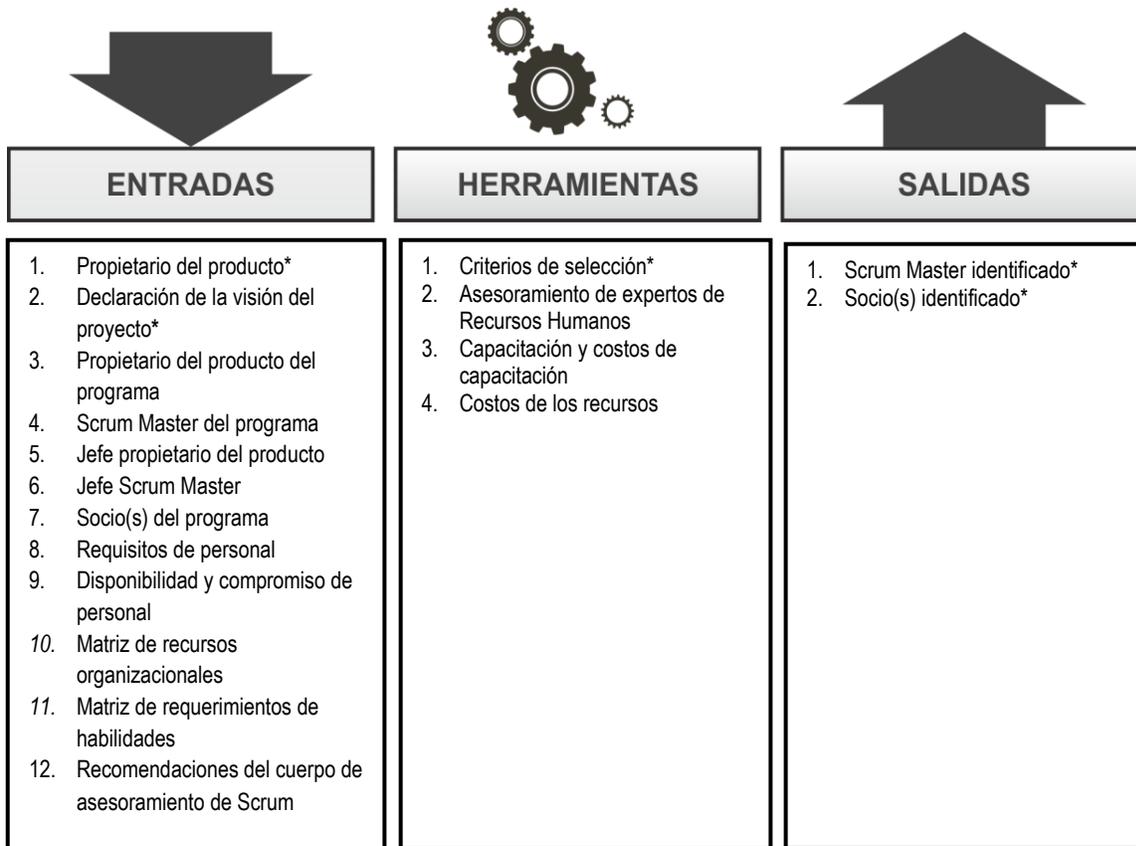


Figura 8-6: Identificación del Scrum Master y el(los) socio(s)—Entradas, herramientas y salidas

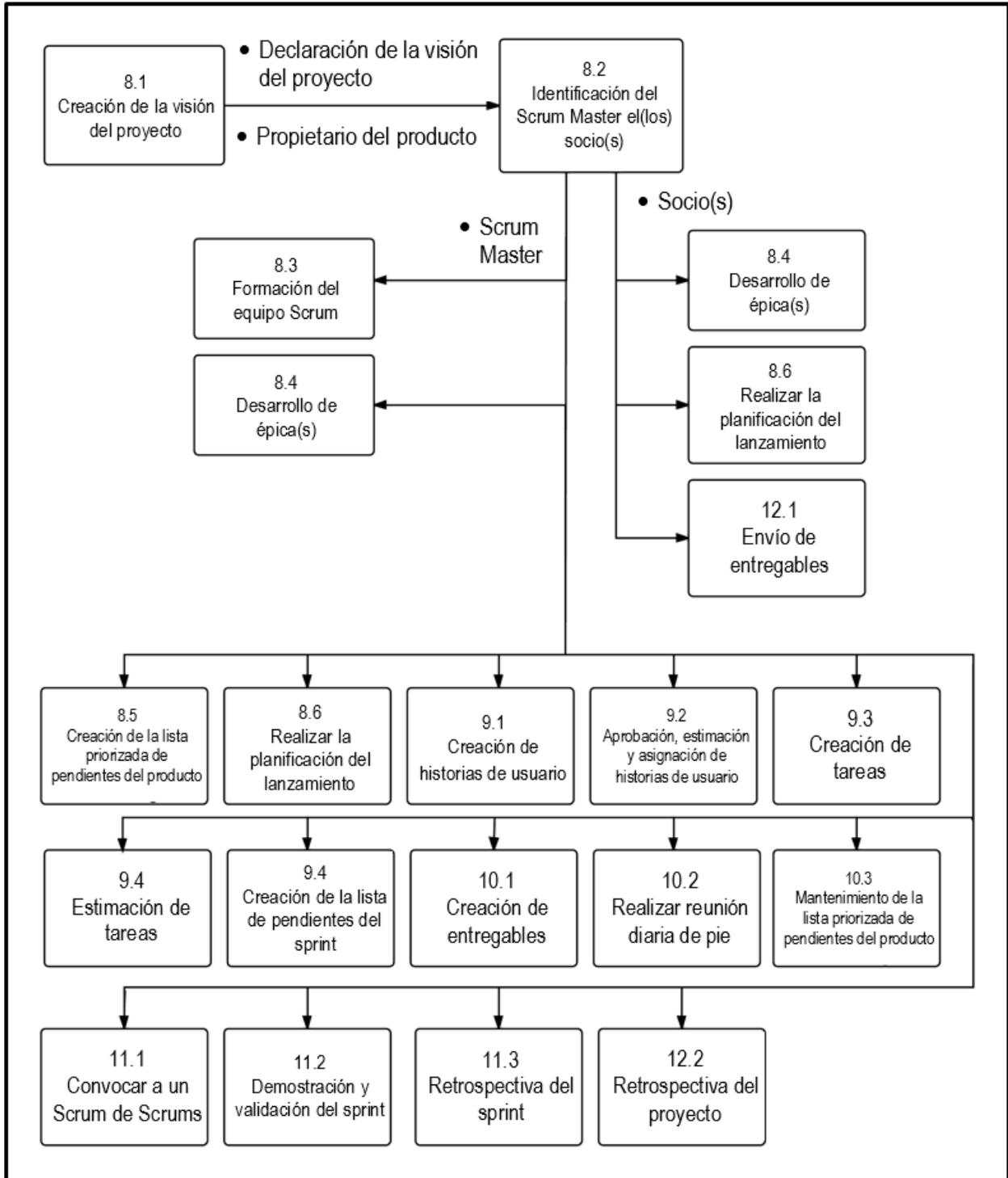


Figura 8-7: Identificación del Scrum Master y el(los) socio(s)—Diagrama de flujo de datos

## **8.2.1 Entradas**

### **8.2.1.1 Propietario del producto\***

Descrito en la sección 8.1.3.1.

### **8.2.1.2 Reunión de la visión del proyecto\***

Descrito en la sección 8.1.3.2.

### **8.2.1.3 Programa propietario del producto**

Descrito en la sección 8.1.1.2.

### **8.2.1.4 Programa Scrum Master**

Descrito en la sección 8.1.1.3.

### **8.2.1.5 Jefe propietario del producto**

Descrito en la sección 8.1.1.5.

### **8.2.1.6 Jefe Scrum Master**

Los grandes proyectos requieren que múltiples Scrum Masters trabajen en paralelo. Es posible que la información obtenida por un equipo se le deba comunicar adecuadamente a los otros equipos –el Jefe Scrum Master es quien es responsable de esta actividad.

La coordinación entre distintos equipos de Scrum que colaboran en un proyecto se realiza normalmente a través de la reunión de Scrum de Scrums (véase la sección 3.7.2.1). Esto es análogo a la reunión diaria de pie y es facilitada por el jefe Scrum Master. Por lo general, el jefe Scrum Master es el responsable de atender los impedimentos que más afectan a un equipo Scrum.

### **8.2.1.7 Socio(s) del programa**

Descrito en la sección 8.1.1.4.

### 8.2.1.8 Requerimientos de las personas

Los requerimientos de las personas es uno de los pasos iniciales en la selección del Scrum Master y el(los) socio(s). Es importante documentar los roles y responsabilidades de todos los que se estarían involucrados en la realización de las tareas del proyecto. Esto incluye a todas las personas involucradas en el proyecto, en cualquier calidad, independientemente de que su papel sea central o no lo sea.

Por lo general, el propietario del producto o el Scrum Master colaboran con el departamento de recursos humanos de la empresa para determinar y concluir los requerimientos de las personas.

### 8.2.1.9 Disponibilidad y compromiso de las personas

Antes de seleccionar al Scrum Master y al(los) socio(s), debe confirmarse su disponibilidad. Solamente deben seleccionarse los miembros del equipo que estarán disponibles y que puedan comprometerse plenamente con el proyecto. La disponibilidad y el compromiso de las personas se representa comúnmente en forma de calendario, el cual muestra cuándo estarán disponibles los recursos humanos para trabajar a través de toda la duración del proyecto.

Para ser eficaces, los equipos Scrum deberían tener idealmente de seis a diez miembros; y la sustitución de las personas o el cambio de los miembros del equipo no son aconsejables en un equipo principal de Scrum. Por lo tanto, es importante contar con personas en un equipo principal de Scrum que estén disponibles y plenamente comprometidas con el proyecto.

### 8.2.1.10 Matriz de recursos organizacionales

La matriz de recursos organizacionales es una representación jerárquica de una combinación de una estructura de organización funcional y una estructura organizativa proyectizada. Las organizaciones matriciales reúnen a miembros de varios equipos del proyecto de diferentes departamentos funcionales, tales como alguien de tecnología, finanzas, comercialización, ventas, manufactura, y otros departamentos—para así crear equipos multifuncionales.

Los miembros del equipo en una organización matricial cumplen dos objetivos: funcionales y de proyecto. Los miembros del equipo son dirigidos por el(los) propietario(s) del producto con respecto a las actividades de proyectos, mientras que los directores funcionales realizan actividades de gestión relacionadas con sus departamentos, como las evaluaciones de desempeño y la aprobación para ausentarse del trabajo.

### 8.2.1.11 Matriz de requerimientos de habilidades

La matriz de requerimientos de habilidades, también llamada marco de competencias, se utiliza para evaluar las carencias de habilidades y los requisitos de formación para los miembros del equipo. Una matriz de habilidades asigna habilidades, capacidades y el nivel de interés de los miembros del equipo en el uso de dichas habilidades y capacidades en un proyecto. Al utilizar esta matriz, la organización puede evaluar

los vacíos de habilidades en los miembros del equipo e identificar a los empleados que van a necesitar más formación en un área o competencia particular.

### 8.2.1.12 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

## 8.2.2 Herramientas

### 8.2.2.1 Criterios de selección\*

La selección adecuada de Scrum Master(s) y la identificación del(los) socio(s) relevante(s) es crucial para el éxito de cualquier proyecto. En algunos proyectos, puede ser que haya habido condiciones fijadas previamente que estipulen a determinados miembros del equipo y sus roles.

Cuando hay flexibilidad en la elección del(los) Scrum Master(s), se deben considerar los siguientes criterios de selección:

1. *Habilidades para la resolución de problemas*—Es uno de los principales criterios a considerar al seleccionar al(los) Scrum Master(s). El(los) Scrum Master(s) debe(n) tener las habilidades y experiencia necesarias para ayudar a eliminar cualquier impedimento que enfrente el equipo Scrum.
2. *Disponibilidad*—El Scrum Master debe estar disponible para programar, supervisar y facilitar varias reuniones, incluyendo la reunión de planificación de del lanzamiento, la reunión diaria de pie, y otras reuniones relacionadas al Sprint.
3. *Compromiso*—El Scrum Master se debe comprometer a que el equipo Scrum esté dotado de un ambiente de trabajo propicio para asegurar la entrega exitosa de los proyectos Scrum.
4. *Estilo de liderazgo servicial*—Para más detalles, consulte la sección 3.10.4.1

En la identificación del(los) socio(s), es importante recordar que se incluye a todos los clientes, usuarios y patrocinadores, quienes a menudo interactúan con el propietario del producto, Scrum Master y equipo Scrum para proveer entradas y facilitar la creación de los productos del proyecto. Los socios influyen en el proyecto a lo largo de su ciclo de vida.

### 8.2.2.2 Asesoramiento de expertos RH

El asesoramiento de expertos de RH puede ser útil en la identificación del Scrum Master y del(los) socio(s). El departamento de recursos humanos posee un conocimiento específico sobre los empleados de una organización y las diversas técnicas que pueden ayudar a identificar a un Scrum Master y socio(s).

### 8.2.2.3 Capacitación y costos de capacitación

Scrum es un marco radicalmente diferente a los métodos tradicionales de gestión de proyectos. Los miembros del equipo no siempre pueden poseer los conocimientos o habilidades necesarias para trabajar en el ambiente de Scrum. El propietario del producto debe evaluar las necesidades de capacitación de los miembros potenciales del equipo y facilitar la capacitación para eliminar brechas de conocimiento del equipo. El propietario del producto es normalmente responsable de la evaluación y la selección de los miembros del equipo, pero a menudo lo hace en consulta con el Scrum Master, quien puede tener un conocimiento adicional de los recursos por trabajar con ellos en otros proyectos.

Se debe proporcionar una capacitación adecuada a los miembros del equipo Scrum, tanto antes del inicio del trabajo y así como mientras están trabajando en sus proyectos. Los miembros del equipo Scrum también deben estar dispuestos a aprender de los demás, y de quienes tienen más experiencia en el equipo.

### 8.2.2.4 Costo de los recursos

Una de las principales consideraciones en la selección de las personas tiene que ver con las ventajas y desventajas relacionadas con el nivel de experiencia en comparación al salario. Hay otros factores relacionados a las personas que influyen en el costo que también se deben considerar. Idealmente, el(los) Scrum Master(s), los miembros del equipo, y el(los) socios(s) deben estar co-ubicados, para que puedan comunicarse con frecuencia y facilidad. Si la co-ubicación no es posible y hay equipos en diferentes zonas geográficas, se debe tener recursos adicionales para facilitar la comunicación, el entendimiento de las diferencias culturales, sincronizar el trabajo y fomentar el intercambio de conocimientos.

## 8.2.3 Salidas

### 8.2.3.1 Scrum Master identificado\*

Un Scrum Master es un facilitador y un líder servicial que se asegura de que el equipo Scrum esté dotado de un ambiente propicio para completar el proyecto con éxito. El Scrum Master guía, facilita y les enseña prácticas de Scrum a todos los involucrados en el proyecto; elimina impedimentos que enfrenta el equipo; y asegura que se estén siguiendo los procesos de Scrum. Es la responsabilidad del propietario del producto identificar al Scrum Master para un proyecto Scrum.

El papel de Scrum Master se describe con más detalle en la sección 3.4

### 8.2.3.2 Socio(s) identificado(s)\*

El(los) socio(s), que es un término colectivo que incluye a clientes, usuarios y patrocinadores, con frecuencia interactúa(n) con el equipo principal de Scrum e influye(n) en el proyecto durante todo el proceso de desarrollo de productos. Es para los socios que el proyecto produce los beneficios deseados de colaboración.

La función del(los) socios(s) se describe en la sección 3.3.2.

## 8.3 Formación de un equipo Scrum

La figura 8-8 muestra todas las entradas, las herramientas y las salidas para el proceso de *Formación de un equipo Scrum*.

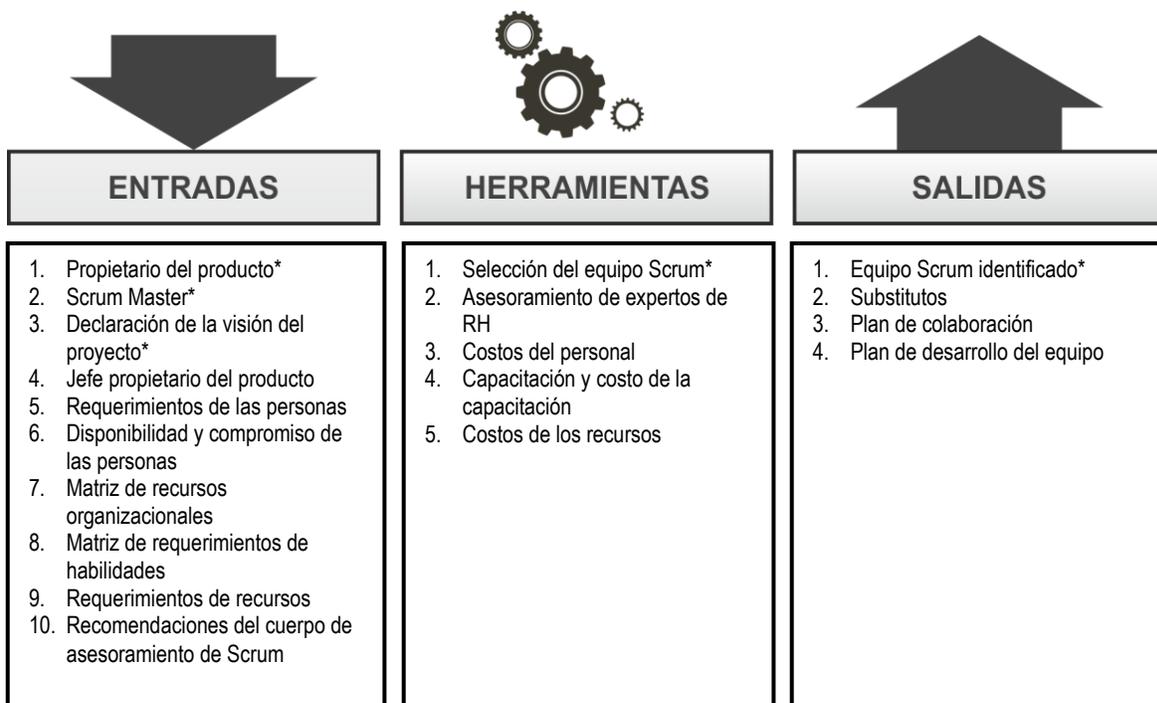


Figura 8-8: Formación de un equipo Scrum—Entradas, herramientas y salidas

Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

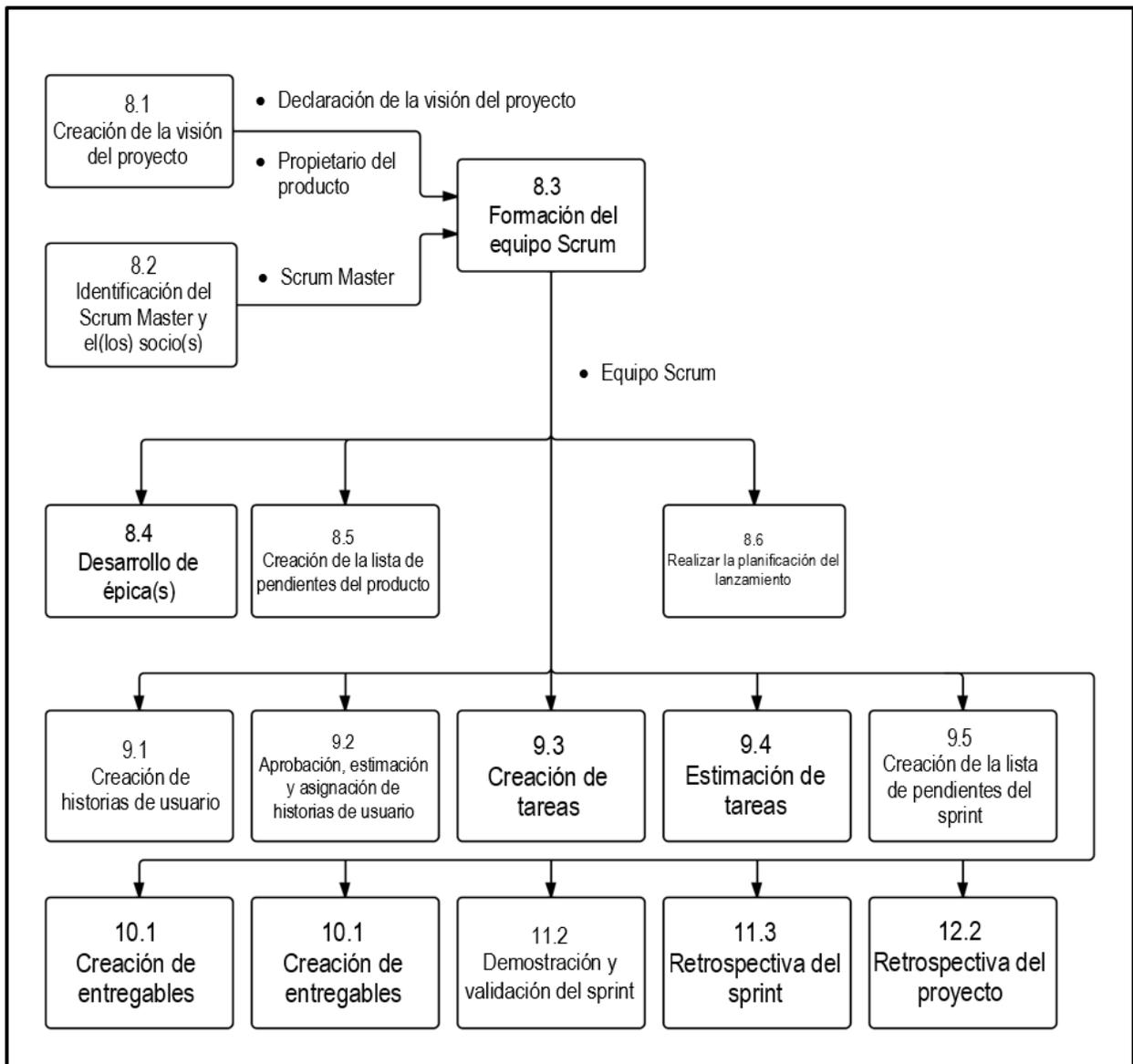


Figura 8-9: Formación de un equipo Scrum—Diagrama de flujo de datos

## 8.3.1 Entradas

### 8.3.1.1 Propietario del producto\*

Descrito en la sección 8.1.3.1.

### 8.3.1.2 Scrum Master\*

Descrito en la sección 8.2.3.1.

### 8.3.1.3 Reunión de la visión del proyecto\*

Descrito en la sección 8.1.3.2.

### 8.3.1.4 Jefe propietario del producto

Descrito en la sección 8.1.1.5.

### 8.3.1.5 Requerimientos de las personas

Descrito en la sección 8.2.1.8.

### 8.3.1.6 Disponibilidad y compromiso de las personas

Descrito en la sección 8.2.1.9.

### 8.3.1.7 Matriz de recursos organizacionales

Descrito en la sección 8.2.1.10.

### 8.3.1.8 Matriz de requerimientos de habilidades

Descrito en la sección 8.2.1.11.

### 8.3.1.9 Requisitos de recursos

Estos requisitos incluyen todos los recursos, humanos o no, necesarios para que el equipo Scrum trabaje eficazmente. Estos recursos incluyen infraestructura de oficinas, espacios de reunión, equipos de trabajo, tableros Scrum, etc. En el caso de los equipos virtuales, recursos adicionales, tales como herramientas de colaboración, videoconferencia, repositorios de documentos compartidos, servicios de traducción, etc., tienen que considerarse.

### 8.3.1.10 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

## 8.3.2 Herramientas

### 8.3.2.1 Selección del equipo Scrum\*

El equipo Scrum es la base de cualquier proyecto de Scrum y el tener a los miembros adecuados para el equipo es vital para la entrega exitosa de los proyectos Scrum. Los miembros del Equipo Scrum son generalistas/especialistas ya que cuentan con conocimiento en diversos campos y son expertos en al menos uno. Más allá de la experiencia en la materia, son las habilidades interpersonales de cada miembro del equipo que determinará el éxito de los equipos auto-organizados.

Los miembros ideales del equipo Scrum son independientes, auto-motivados, se enfocan en el cliente, y tienen un sentido alto de la responsabilidad y colaboración. El equipo debe ser capaz de fomentar un ambiente de reflexión independiente y de tomar decisiones con el fin de extraer los mayores beneficios de la estructura.

### 8.3.2.2 Asesoramiento de expertos en recursos humanos

El asesoramiento de expertos de la gerencia de recursos humanos (RH) puede ser valiosa al formar un equipo Scrum. Este departamento posee un conocimiento especializado sobre los empleados de una organización y de las numerosas técnicas que pueden ayudar a los propietarios del producto, Scrum Masters, y a los patrocinadores para identificar a los mejores miembros del equipo.

### 8.3.2.3 Costo asociado con el personal

Todos los costos asociados con los requisitos del personal deben evaluarse, analizarse, aprobarse y presupuestarse.

### 8.3.2.4 Costos de formación y capacitación

Es posible que los miembros del equipo no cuenten con las habilidades o conocimientos necesarios para llevar a cabo tareas especializadas. En ese caso, el propietario del producto debe evaluar las necesidades de capacitación de los miembros potenciales del equipo y proporcionarles capacitación, cuando se ha determinado una carencia de habilidades o conocimientos.

Para que una implementación de Scrum sea realmente eficaz, tiene que haber un alto nivel de conocimiento dentro de la organización de los principios y valores de Scrum. Este conocimiento ayudará a la ejecución exitosa de Scrum. El equipo Scrum tiene que estar capacitado en las prácticas de Scrum, y es el Scrum Master quien debe desempeñar el rol de capacitador de este tema para el equipo. Dado a que la planificación de los sprints es un factor muy importante para el éxito, la capacitación ayudará a los miembros del equipo a entender cómo discutir e identificar metas alcanzables en un sprint. El Scrum Master tiene que aprovechar lo mejor del equipo Scrum motivándolos y facilitando el proceso de desarrollo. Con la capacitación correcta, el Scrum Master puede ayudar al equipo a articular los problemas y desafíos que puedan enfrentar. Normalmente, cualesquier problemas o conflictos en el equipo se resuelven por el mismo

equipo con capacitación y asistencia del Scrum Master, según sea necesario. El Scrum Master debe atender los problemas como la baja moral o falta de coordinación dentro del equipo. Es responsable de la eliminación de impedimentos para el equipo. Cuando sea necesario, el Scrum Master puede presentarles estos problemas externos e impedimentos de gestión a la dirección para llegar a una resolución.

Los costos de formación y capacitación se tratan en el proceso de identificación del Scrum Master y el(los) socio(s), en la sección 8.2.2.3.

### **8.3.2.5 Costo de los recursos**

Los costos asociados con todos los requisitos no relacionados con las personas deben evaluarse, analizarse, aprobarse y presupuestarse. Un recurso en el ambiente del proyecto se identifica como aquello que se usa para realizar una tarea o actividad, incluyendo —pero sin limitarse— a equipos, materiales, servicios externos, y el ambiente físico.

## **8.3.3 Salidas**

### **8.3.3.1 Equipo Scrum identificado\***

El equipo Scrum, a veces conocido como equipo de desarrollo, es un grupo o equipo de personas que son responsables de la comprensión de los requerimientos especificados por el propietario del producto, la estimación de historias de usuario y la creación definitiva de los entregables del proyecto. Los equipos Scrum son interfuncionales y auto-organizados. El equipo decide la cantidad de trabajo al que se comprometerá en un sprint y determina la mejor manera de realizar las tareas. El equipo se compone de miembros interfuncionales, que llevan a cabo todo el trabajo involucrado en la creación de entregables que potencialmente se puedan enviar, incluyendo el desarrollo, la prueba, la garantía de calidad, etc.

La identificación del equipo Scrum es la responsabilidad del propietario del producto, a menudo en consulta con el Scrum Master.

El rol del equipo Scrum se describe con más detalle en la sección 3.6.

### **8.3.3.2 Substitutos**

Al seleccionar a los equipos, otro aspecto importante es la creación de substitutos para cada miembro del equipo Scrum. Aunque la disponibilidad y compromiso por los miembros del equipo se confirman por adelantado, pueden surgir problemas tales como una enfermedad, emergencia familiar, o simplemente el hecho de que un miembro deje de ser parte de la organización. Los equipos Scrum trabajan en pequeños grupos de seis a diez personas. El contar substitutos asegura que no haya una disminución importante en la productividad debido a la falta de un miembro del equipo.

### 8.3.3.3 Plan de colaboración

La colaboración es un elemento muy importante en Scrum. La planificación de cómo las distintas personas que toman decisiones, los socios, y miembros del equipo participan y colaboran entre sí, es vital. El plan de colaboración es una salida opcional que puede ser formal o informal. A veces, puede ser simplemente un entendimiento verbal entre los diversos socios, ya en que Scrum se evita toda la documentación innecesaria. Sin embargo, para los proyectos más grandes y complejos, especialmente aquellos con equipos en diferentes áreas geográficas, puede ser necesario poner en marcha un acuerdo más formal. El plan puede abordar cómo los miembros de equipo principal de Scrum, socio(s) y otras personas involucradas en el proyecto Scrum se comunican y colaboran durante todo el proyecto y también puede definir las herramientas o técnicas específicas que se utilizarán para este fin. Por ejemplo, en el caso de equipos distribuidos, puede haber una necesidad de un acuerdo sobre cuándo y cómo se llevarán a cabo las reuniones, qué tipo de herramientas de comunicación se utilizarán, y quién debe estar involucrado en las diversas reuniones.

### 8.3.3.4 Plan de desarrollo del equipo

Debido a que un equipo Scrum es interfuncional, cada miembro debe participar activamente en todos los aspectos del proyecto. El Scrum Master debe identificar los problemas con los miembros del equipo y hacer frente a estos a fin de mantener un equipo eficaz.

Para tener cohesión en el equipo, el Scrum Master debe asegurarse que las relaciones entre los miembros del equipo sean positivas y que estén unificados en el cumplimiento de la totalidad del proyecto y de los objetivos de la organización, lo que conduce a una mayor eficiencia y una mayor productividad.

En este contexto, es importante estudiar la sección 3.10, que habla sobre las teorías populares de recursos humanos y su relevancia para Scrum.

## 8.4 Desarrollo de épica(s)

La Figura 8-10 muestra todas las entradas, herramientas y salidas para el proceso de Desarrollo de épica(s).

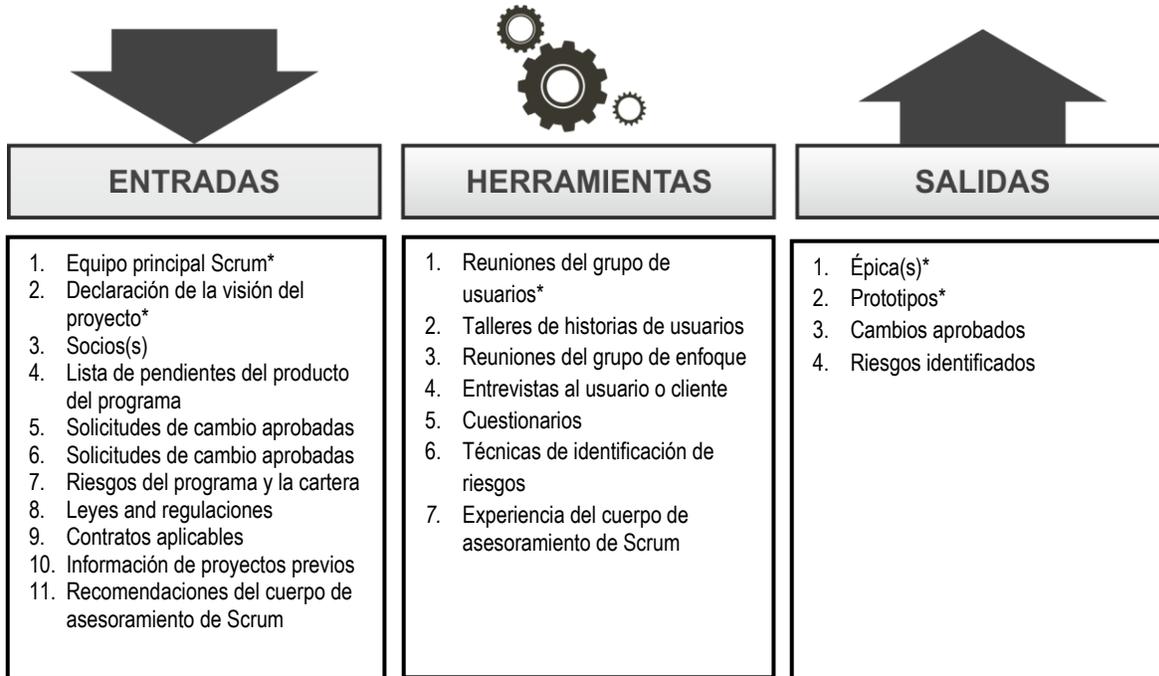


Figura 8-10: Desarrollo de épica(s)—Entradas, herramientas y salidas

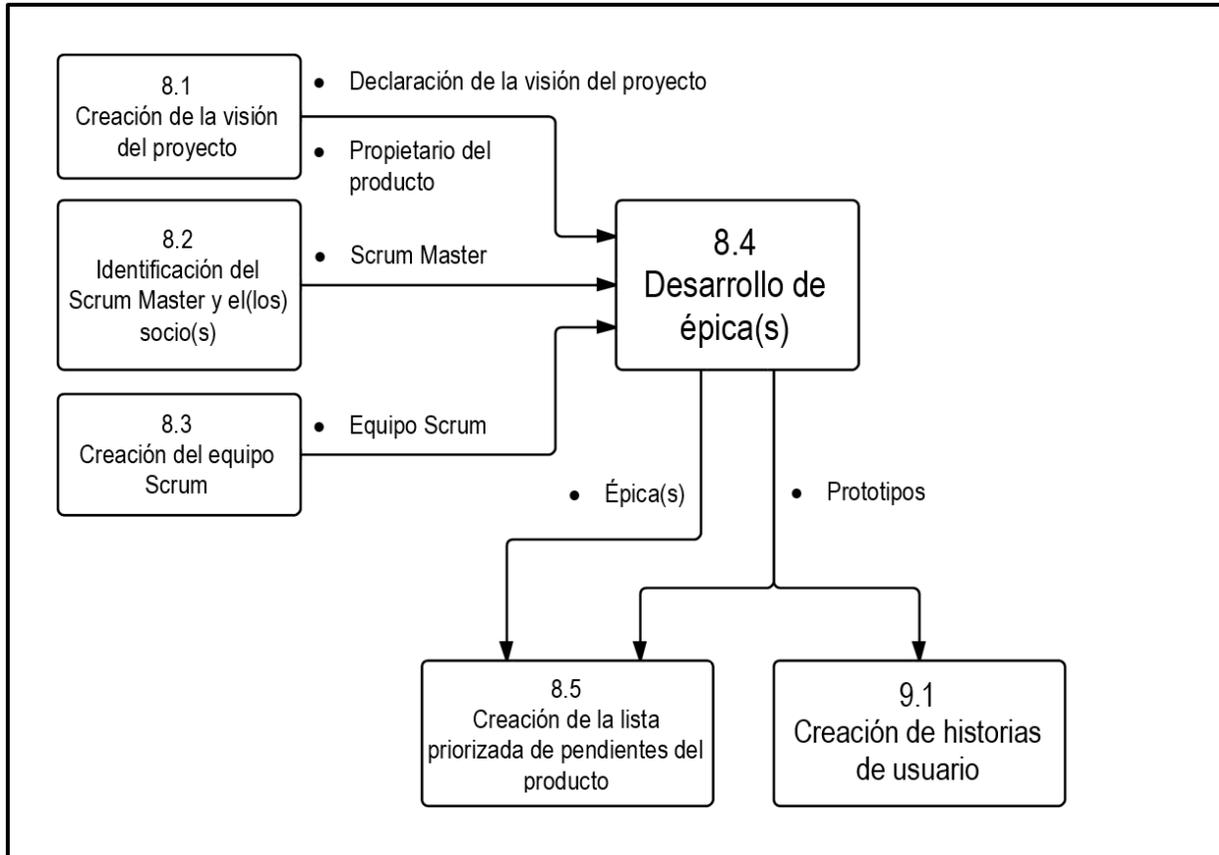


Figura 8-11: Desarrollo de épica(s)—Diagrama de flujo

## 8.4.1 Entradas

### 8.4.1.1 Equipo principal Scrum \*

El equipo principal de Scrum consiste en el equipo Scrum, el Scrum Master y el propietario del producto como se describe en la sección 3.3.1.

### 8.4.1.2 Reunión de la visión del proyecto\*

Descrito en la sección 8.1.3.2.

### 8.4.1.3 Socio(s)

Descrito en la sección 8.2.3.2.

### 8.4.1.4 Lista de pendientes del producto del programa

Descrito en la sección 8.1.1.6.

#### 8.4.1.5 Solicitudes de cambio aprobadas

Las solicitudes de cambio aprobadas que se originan del programa o la cartera son entradas que se añadirán a la lista de cambios aprobados de proyecto para su ejecución en sprints futuros. Cada cambio puede requerir su propia épica o historia de usuario y pudiera convertirse en un aporte al proceso de *desarrollo de épica(s)*. En este proceso, las solicitudes de cambio aprobadas también pudieran ser el resultado de otros procesos de Scrum.

Las solicitudes de cambio y las solicitudes de cambio aprobadas se tratan en las secciones 6.3.1, 6.4.2.1 y 6.6.

#### 8.4.1.6 Solicitudes de cambio no aprobadas

Una solicitud para realizar cambios se presenta por lo general como solicitudes de cambio y permanecen en un estado de “no aprobadas” hasta que se aprueban formalmente. Las solicitudes de cambio no aprobadas para el proceso de desarrollo de épica(s) pudiera provenir de los procesos de crear entregables, realizar la reunión diaria de pie, así como otros procesos.

Las solicitudes de cambio y las solicitudes de cambio no aprobadas se tratan en las secciones 6.3.1, 6.4.2.1 y 6.6.

#### 8.4.1.7 Riesgos del programa y cartera

Los riesgos relacionados con una cartera o programa también tendrán un impacto en los proyectos que forman parte de la respectiva cartera o programa. Durante la evaluación del riesgo en carteras y programas, si se determina que el riesgo puede afectar a un proyecto individual, la información relevante sobre el mismo debe comunicarse al propietario del producto y al equipo Scrum. Estos riesgos pudieran ser entradas para el proceso de *desarrollo de épica(s)*, y pudieran tener un impacto global sobre cómo se lleva a cabo este proceso.

Los riesgos del programa y cartera se describen en la sección 7.5.1.

### 8.4.1.8 Leyes y regulaciones

Dependiendo del proyecto, hay leyes y regulaciones impuestos por un consejo de administración que afectan la planificación y ejecución. Las leyes son externas a la organización e impuestas por una entidad gubernamental. Las regulaciones pueden ser o bien internas o externas. Las regulaciones internas son aquellas que son aplicables dentro de la empresa, por lo general basadas en la política de la empresa. Estas regulaciones pueden estar relacionadas con sistemas de gestión de la calidad, regulaciones financieras, reglamentos de personal, etc. Las regulaciones externas son las relativas a los criterios, las normas y los requisitos gubernamentales establecidos.

Las leyes y regulaciones se deben considerar al desarrollar épicas. Las épicas se basan en los requerimientos de negocio. Para cumplir con estos requerimientos, el equipo del proyecto tiene que cumplir con ambas leyes y reglamentos internos y externos.

A veces, algunas de las leyes y regulaciones que afectan a múltiples proyectos Scrum pueden incluirse como parte de las recomendaciones del cuerpo de asesoramiento de Scrum, como se discutió en la sección 8.1.1.12.

### 8.4.1.9 Contratos aplicables

Si todo el proyecto o partes del mismo se cumplen en virtud de un contrato, el contrato ha de definir el alcance del trabajo y las condiciones específicas del mismo. El tipo de contrato que se utiliza tendrá influencia en el riesgo del proyecto.

Algunos de los tipos más comunes de contratos utilizados en proyectos Scrum son los siguientes:

**Contrato de entrega incremental**—Este contrato incluye puntos de inspección en intervalos frecuentes. Ayuda a que el cliente o los socios tomen decisiones periódicas sobre el desarrollo a lo largo de un proyecto en cada punto de inspección. El cliente puede aceptar el desarrollo del producto, optar por detener su desarrollo o solicitar modificaciones al mismo.

**Contrato de empresa conjunta**—Este tipo de contratos se utiliza generalmente cuando dos o más partes se asocian para realizar el trabajo de un proyecto. Ambas partes involucradas en el proyecto lograrán algún retorno sobre la inversión, ya que los ingresos o beneficios que sean generados, se habrán de compartir entre las partes.

**Contrato de desarrollo en fases**—Este contrato facilita la disponibilidad de fondos cada mes o cada trimestre después de concluir satisfactoriamente un lanzamiento. Incentiva tanto al cliente como al proveedor y garantiza que el riesgo monetario del cliente esté limitado a un periodo específico, ya que los lanzamientos fracasados no reciben financiamiento.

**Contrato de incentivos y sanciones**—Este contrato se basa en el acuerdo de que el proveedor será recompensado con un incentivo económico si los productos del proyecto se entregan a tiempo, pero incurrirá en sanciones económicas si la entrega se realiza tarde.

Otros tipos de contratos más populares incluyen el pago por las características del contrato, el tiempo y los materiales del contrato, precio fijo y contrato de alcance fijo, y el contrato de beneficio fijo.

Las épicas deberían desarrollarse teniendo en cuenta los términos y condiciones del tipo de contrato que se utilice.

#### **8.4.1.10 Información de proyectos previos**

La información y los conocimientos adquiridos en proyectos anteriores similares dentro de la organización son entradas valiosas para el desarrollo de épicas y la evaluación de riesgos. La información de proyectos previos podría incluir notas del director del proyecto, registros del proyecto y comentarios hechos por el(los) socio(s).

Algunos detalles y las mejores prácticas relacionadas con información de proyectos anteriores pueden estar disponibles a través de las recomendaciones del cuerpo de asesoramiento de Scrum.

#### **8.4.1.11 Recomendaciones del cuerpo de asesoramiento de Scrum**

Tratados en la sección 8.1.1.12

Las recomendaciones del cuerpo de asesoramiento de Scrum pueden incluir información sobre normas, reglamentos, regulaciones y mejores prácticas para el desarrollo de épicas.

### **8.4.2 Herramientas**

#### **8.4.2.1 Reuniones del grupo de usuarios\***

Las reuniones del grupo de usuarios involucran a socios relevantes (principalmente usuarios o clientes del producto). Ellos proporcionan al equipo principal de Scrum información de primera mano acerca de las expectativas del usuario. Esto ayuda en la formulación de los criterios de aceptación para el producto y proporciona información valiosa para el desarrollo de épicas. Las reuniones del grupo de usuarios son de vital importancia en la prevención de trabajo costoso que puedan deberse a la falta de claridad con respecto a las expectativas y exigencias. Estas reuniones también promueven un sentido de compromiso con el proyecto, así como un entendimiento común entre el equipo principal de Scrum y los socios relevantes.

### 8.4.2.2 Talleres de historias de usuario

Los talleres de historias de usuarios se llevan a cabo como parte del proceso de *desarrollo de épica(s)*. El Scrum Master facilita estas sesiones en las que todo el equipo principal Scrum interviene y, en ocasiones, es deseable incluir a otro(s) socio(s). Estos talleres ayudan al propietario del producto a dar prioridad a los requerimientos y permitir que el equipo principal Scrum tenga una perspectiva compartida de los criterios de aceptación. Estos aseguran de que las épicas e historias de usuario describan la funcionalidad desde el punto de vista de los usuarios, sean fáciles de entender, y se puedan calcular de forma fiable. Dichos talleres son útiles en la comprensión de las expectativas del usuario para los entregables, y son excelentes para la formación de equipos. También facilitan la preparación para la planificación del próximo sprint. Un taller de historias del usuario es una buena plataforma para discutir y aclarar todos los elementos de un producto, y con frecuencia para profundizar en los detalles más pequeños para garantizar la claridad.

### 8.4.2.3 Reuniones del grupo de enfoque

En las reuniones del grupo de enfoque las personas se reúnen en una sesión guiada para proporcionar sus opiniones, percepciones o valoraciones de un producto, servicio o resultado deseado. Los miembros del grupo de enfoque tienen la libertad de hacerse preguntas el uno al otro y de obtener aclaraciones sobre temas o conceptos específicos. A través de cuestionamiento, la crítica constructiva y la retroalimentación, los grupos de enfoque conducen a un producto de mejor calidad y con ello contribuyen a la satisfacción de las expectativas de los usuarios. En estas reuniones, los miembros del grupo de enfoque a veces llegan a un consenso en ciertas áreas, mientras que en otras áreas las opiniones pueden ser diferentes. Cuando los miembros del grupo tienen diferentes opiniones o puntos de vista, no se escatiman esfuerzos para resolver las diferencias con el fin de llegar a un consenso.

Las sesiones de grupos de enfoque pueden ayudar a los equipos a crear ideas innovadoras, resolver problemas, y dar sugerencias para mejorar. Estas reuniones generan ideas y retroalimentación de los usuarios potenciales y desarrolladores de productos. Estas reuniones se realizan normalmente para la planificación, evaluación, o mejora de un producto o servicio. El conocimiento obtenido a partir de estas reuniones también puede ayudar con el desarrollo de épicas e historias del usuario. A veces, los grupos de enfoque se llevan a cabo para resolver problemas que puedan surgir durante el desarrollo de las épicas.

### 8.4.2.4 Entrevistas al usuario o cliente

El incorporar a los socios, incluyendo al patrocinador, usuarios y clientes del producto es importante para obtener el contexto y la visión necesarios para desarrollar épicas. Una buena cantidad de tiempo invertida en entrevistar a los usuarios y clientes resultará en un mejor alineamiento de los requisitos en las épicas con la visión general del proyecto, ofreciendo de esta forma un mayor valor.

Estas entrevistas ayudan a:

- Identificar y entender las necesidades y expectativas del socio
- Recopilar opiniones y hechos
- Comprender la perspectiva del socio sobre el producto final
- Recopilar retroalimentación sobre el producto iterado o parcialmente desarrollado

#### **8.4.2.5 Cuestionarios**

Una forma económica de obtener una perspectiva estadística cuantitativa y cualitativa de un gran número de usuarios o clientes es el uso de encuestas o cuestionarios. Un cuestionario es un instrumento de investigación con el fin de recopilar información sobre un asunto o tema específico. Los cuestionarios pueden ser auto-administrados o administrados por un entrevistador.

Se debe ejercer gran cuidado en el diseño de cuestionarios, la selección adecuada del público, y la determinación de un método apropiado de implementación de encuestas es también de gran importancia para evitar errores y prejuicios.

Durante el desarrollo de épicas, el propietario del producto o el Scrum Master podría llevar a cabo una encuesta para recopilar la información pertinente de socios o el equipo Scrum.

#### **8.4.2.6 Técnicas de identificación de riesgos**

Descrito en la sección 7.4.1.1

#### **8.4.2.7 Experiencia del cuerpo de asesoramiento de Scrum**

Descrito en la sección 3.3.2

Al crear las épicas, la experiencia del cuerpo de asesoramiento de Scrum podría referirse a las normas y reglamentos documentados; o los estándares y mejores prácticas para crear las épicas. Es posible que exista también un equipo de expertos en la materia que pudiera ayudar al propietario del producto a crear épicas. Este equipo podría incluir a analistas empresariales, arquitectos líderes, desarrolladores, expertos de Scrum o demás personas con experiencia. Por lo general, este grupo de expertos no es el mismo equipo que permanecerá y trabajará en un proyecto en particular, ya que tienden a pasar de un proyecto a otro durante la "fase de venta" o "fase cero" con los clientes o usuarios.

## 8.4.3 Salidas

### 8.4.3.1 Épica(s)\*

Las épicas se redactan en las etapas iniciales del proyecto, cuando la mayoría de las historias de usuario son funcionalidades de alto nivel o descripciones de productos que están ampliamente definidas. Las épicas son historias de usuario grandes sin refinar en la lista priorizada de pendientes del producto.

Una vez que estas épicas aparecen en la lista priorizada de pendientes del producto para ser terminadas en el próximos, se convierten en historias de usuario más pequeñas. Estas historias más pequeñas son generalmente funcionalidades simples, cortas y fáciles de implementar, o bloques de tareas que deben completarse en un sprint.

### 8.4.3.2 Prototipos\*

Los prototipos son personajes ficticios altamente detallados. Estos representarán a la mayoría de los usuarios y otros socios que pudieran no utilizar directamente el producto final. Los prototipos se crean para identificar las necesidades de los usuarios. La creación de prototipos específicos puede ayudar al equipo a entender mejor a los usuarios y sus necesidades y metas. Basado en un prototipo, el propietario del producto puede priorizar de manera más efectiva las funciones para crear la lista priorizada de pendientes del producto.

**Creación de un prototipo:** Esto implica la asignación de un nombre ficticio y de preferencia una foto para el personaje, como una fotografía de archivo. Al prototipo se le incluirán atributos muy específicos como su edad, género, nivel de educación, ambiente, intereses y metas. Una cita que ilustre las necesidades del prototipo puede incluirse también. A continuación, se muestra un ejemplo de un prototipo para un sitio web de viajes.

*Ejemplo:*

Vanessa tiene 39 años de edad y es residente de San Francisco. Ella es apasionada de los viajes y después de haber tenido una exitosa carrera como abogada, ha decidido disfrutar de dicha pasión. Le gusta tener disponibles opciones al seleccionar sus viajes por avión y servicios de alojamiento para que ella pueda elegir el mejor, a un precio accesible. Ella experimenta frustración con los sitios web lentos y desordenados.

### 8.4.3.3 Cambios aprobados

Las solicitudes de cambio no aprobadas pueden aprobarse por el propietario del producto durante el proceso de *desarrollo de épica(s)*, a veces con sugerencias que proporcionan los socios relevantes. Tales cambios se clasifican como cambios aprobados, y pueden ser priorizados e implementados en los futuros sprints.

Las solicitudes de cambio y las solicitudes de cambio aprobadas se tratan en las secciones 6.3.1, 6.4.2.1 y 6.6.

#### **8.4.3.4 Riesgos identificados**

Al crear épicas, se pueden identificar nuevos riesgos, los cuales constituyen una salida importante de esta etapa. Estos riesgos contribuyen al desarrollo de la lista priorizada de pendientes del producto (también se conoce como lista de pendientes del producto del riesgo ajustado).

La identificación de riesgos se describe en la sección 7.4.1.

## 8.5 Creación de la lista priorizada de pendientes del producto

La Figura 8-12 muestra todas las entradas, las herramientas y salidas para el proceso de *creación de la lista priorizada de pendientes del producto*.

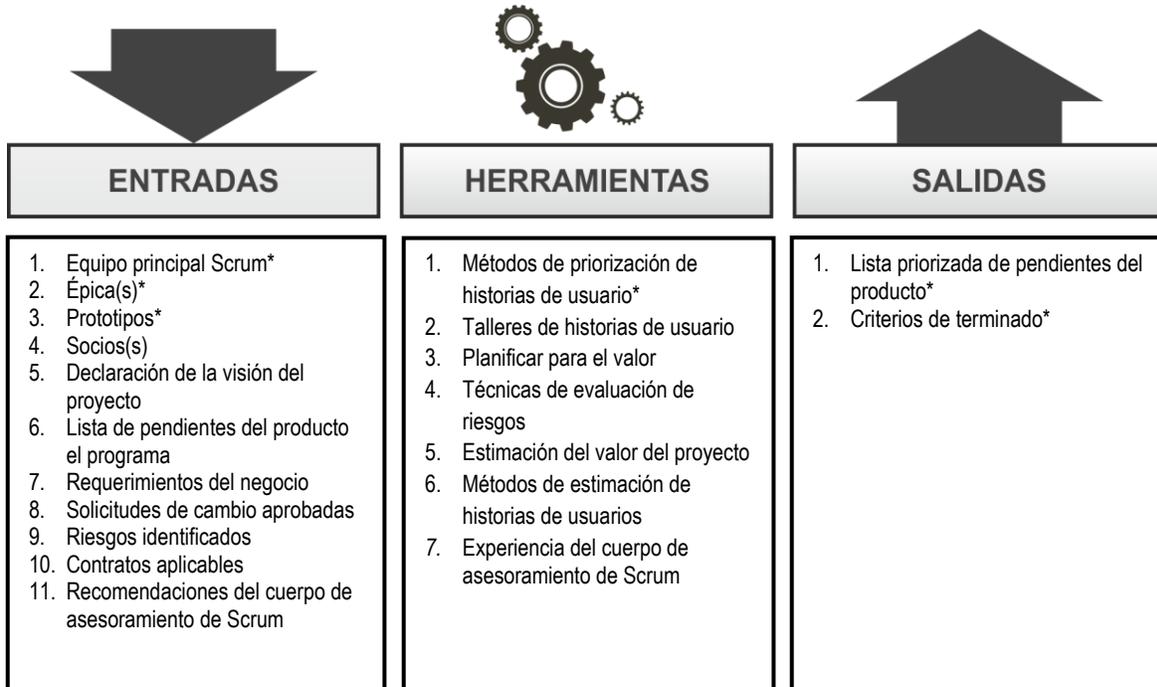


Figura 8-12: Creación de la lista priorizada de pendientes del producto—Entradas, herramientas y salidas

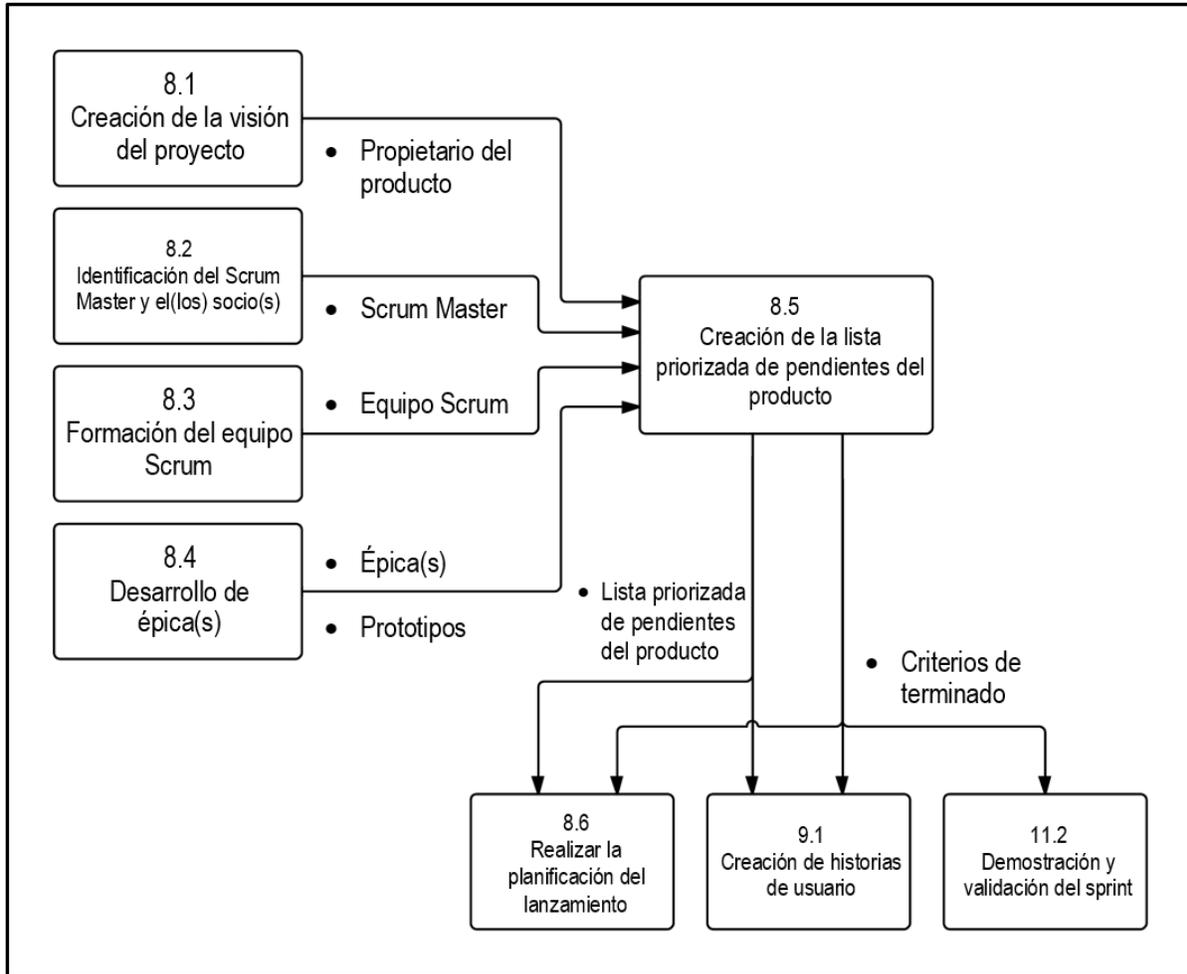


Figura 8-13: Creación de la lista priorizada de pendientes del producto— Diagrama de flujo de datos

## **8.5.1 Entradas**

### **8.5.1.1 Equipo principal de Scrum\***

Descrito en la sección 8.4.1.1.

### **8.5.1.2 Épica(s)\***

Descrito en la sección 8.4.3.1.

### **8.5.1.3 Prototipos\***

Descrito en la sección 8.4.3.2.

### **8.5.1.4 Socios(s)**

Descrito en la sección 8.2.3.2.

### **8.5.1.5 Reunión de la visión del proyecto**

Descrito en la sección 8.1.3.2.

### **8.5.1.6 Lista de pendientes del producto programa**

Descrito en la sección 8.1.1.6.

### **8.5.1.7 Requerimientos de negocio**

La suma de todos los conocimientos adquiridos a través de diversas herramientas tales como las entrevistas a los clientes o usuarios o los cuestionarios, las sesiones de diseño de aplicación conjunta, el análisis de brecha, el análisis FODA, y otras reuniones, ayudan a desarrollar una mejor perspectiva sobre los requerimientos y ayuda en la creación de la lista priorizada de pendientes del producto.

### **8.5.1.8 Solicitudes de cambio aprobadas**

Descrito en la sección 8.4.3.3.

### **8.5.1.9 Riesgos identificados**

Descrito en la sección 8.4.3.4.

### 8.5.1.10 Contratos aplicables

Descrito en la sección 8.4.1.9.

### 8.5.1.11 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

Durante la creación de la lista priorizada de pendientes del producto, las recomendaciones del cuerpo de asesoramiento de Scrum pueden incluir información sobre las normas, los reglamentos, los procedimientos y las mejores prácticas para el desarrollo de la lista priorizada de pendientes del producto.

## 8.5.2 Herramientas

### 8.5.2.1 Métodos de priorización de las historias de usuarios

A continuación, se presentan algunas de las técnicas que se utilizan para dar prioridad a las historias de usuarios o requerimientos en la lista priorizada de pendientes del producto sobre la base del valor de negocio:

- **Esquema de priorización MoSCoW**—El esquema de priorización MoSCoW obtiene su nombre de la versión en inglés de las frases: “Debe tener” (Must have), “Debería tener” (Should have), “Podría tener” (Could have) y “No tendrá” (Won’t have). Las etiquetas están en orden de prioridad decreciente con historias de usuario con características de “debería tener”, siendo aquellas sin las que el producto no tendrá valor, e historias de usuarios con características de “gustaría que tuviera” siendo aquellas que, a pesar de que sería bueno tener, no se es necesario incluir.
- **Comparación por pares**—En esta técnica, la comparación por pares es una técnica donde se prepara una lista de todas las historias de usuario en la lista priorizada de pendientes del producto. Después, cada historia de usuario se toma en forma individual y se compara con otras historias en la lista, una a la vez. Cada vez que se comparan dos historias de usuario, se toma una decisión en cuanto a cuál de las dos es más importante. Por medio de este proceso, se puede generar una lista priorizada de las historias de usuario.
- **El método de los 100 puntos**—El método de los 100 puntos fue desarrollado en el 2003 por Dean Leffingwell y Don Widrig. Dicho método implica otorgar 100 puntos al cliente a fin de que los pueda utilizar para votar por las características que consideren más importantes. El objetivo es dar más peso a las historias de usuarios que son de mayor prioridad en comparación con las otras historias de usuario disponibles. Cada miembro del grupo asigna puntos a las diversas historias de usuarios, dando más puntos a las que opinan son más importantes. Al finalizar el proceso de votación, la priorización se determina calculando el total de puntos asignados a cada historia de usuarios

- **Análisis de Kano**

Descrito en la sección 4.5.2

### **8.5.2.2 Talleres de historias de usuarios**

Descrito en la sección 8.4.2.2.

### **8.5.2.3 Planificación para el valor**

Descrito en la sección 4.5.2

### **8.5.2.4 Técnicas de evaluación de riesgos**

Descrito en la sección 7.4.2.1.

### **8.5.2.5 Estimación de valor del proyecto**

Descrito en la sección 4.5.1.

### **8.5.2.6 Métodos de estimación de historias de usuarios**

Todas las herramientas empleadas para los procesos aprobación, estimación y asignación de historias de usuario (como se describe en la sección 9.2.2) se pueden utilizar para crear estimaciones de alto nivel para la épica(s) cuando se crea la lista priorizada de pendientes del producto. Algunas herramientas importantes son:

1. Reuniones de grupos de usuario
2. Póker de planificación
3. Puño de cinco
4. Puntos de estimación de costos
5. Otras técnicas de estimación

### **8.5.2.7 Experiencia del cuerpo de asesoramiento de Scrum**

Descrito en la sección 8.4.2.7

Durante la creación de la lista priorizada de pendientes del producto, la experiencia del cuerpo de asesoramiento de Scrum podría relacionarse con reglamentos y regulaciones documentadas o con normas y mejores prácticas para la creación de épicas. También puede haber un equipo de expertos en la materia que podría ayudar al propietario del producto con en el proceso de *creación de la lista priorizada de*

*pendientes del producto*. Este equipo podría incluir analistas de negocios, arquitectos líderes, desarrolladores, expertos de Scrum, y otras personas con experiencia. Generalmente, este grupo de expertos no es el mismo equipo que se quedará trabajando en este proyecto, ya que tienden a pasar de un proyecto a otro durante la "fase de venta" o "fase cero" con los clientes o usuarios.

### 8.5.3 Salidas

#### 8.5.3.1 Lista priorizada de pendientes del producto\*

El propietario del producto desarrolla una lista priorizada de pendientes del producto, que contiene una lista priorizada de los requerimientos del negocio y de los proyectos escritos en forma de épica(s), que son historias de usuario de alto nivel. La lista priorizada de pendientes del producto se basa en tres factores principales: valor, riesgo o incertidumbre, y dependencias. También se le conoce como lista de pendientes del producto del riesgo ajustado, dado a que incluye riesgos identificados y evaluados relacionados con el proyecto. También incluye cambios aprobados que pueden ser priorizados adecuadamente en la lista priorizada de pendientes del producto (tal como se describe en la sección 6.3.1).

- **Valor**—Es la responsabilidad del propietario del producto asegurar la entrega de los productos que ofrezcan el mayor valor primero. Incluso un producto de gran valor no puede ser parte del primer lanzamiento si hay otros productos de mayor valor que son suficientes para un primer lanzamiento.
- **Riesgo e incertidumbre**—Cuanta más incertidumbre existe, más riesgoso es el proyecto. Por lo tanto, es importante que se les dé mayor prioridad los productos de mayor riesgo en la lista priorizada de pendientes del producto. Los productos que llevan un mayor nivel de riesgo también requerirán acciones de mitigación de riesgos. Cuando estas acciones de mitigación de riesgos se priorizan frente a la lista de pendientes, el resultado es una lista de pendientes del producto del riesgo ajustado. Tratar con riesgos al principio del proyecto no garantiza que el proyecto tendrá éxito, pero sí mejorará la capacidad del equipo para hacer frente a los riesgos. Esto se describe en la sección 7.4.3.
- **Dependencias**—Por lo general, no es posible crear un lista priorizada de pendientes del producto en la que no existan dependencias entre las historias de usuarios. Los requerimientos funcionales a menudo dependen de otros requerimientos funcionales e incluso no funcionales. Estas dependencias pueden afectar cómo se priorizan las historias de usuarios en la lista priorizada de pendientes del producto. Dos de las formas más comunes para resolver las dependencias son, o bien dividir una sola historia en varias partes, o combinar historias interdependientes.
- **Estimaciones**—Las estimaciones de alto nivel para la épica(s) también están disponibles en la lista priorizada de pendientes del producto.

### 8.5.3.2 Criterios de terminado\*

Criterios de terminado es un conjunto de reglas que se aplican a todas las historias de usuarios. Una definición clara de terminado es crítica, ya que elimina la ambigüedad de los requisitos, y ayuda a que el equipo se adhiera a las normas de calidad obligatorias. Esta clara definición se utiliza para crear los criterios de terminado, que son un resultado del proceso de *creación de la lista priorizada de pendientes del producto*. Una historia de usuario se considera terminada cuando se le demuestra y se aprueba por el propietario del producto, que juzga sobre la base de los criterios de terminado y los criterios de aceptación de las historias de usuarios.

*Ejemplo de criterios de terminado:*

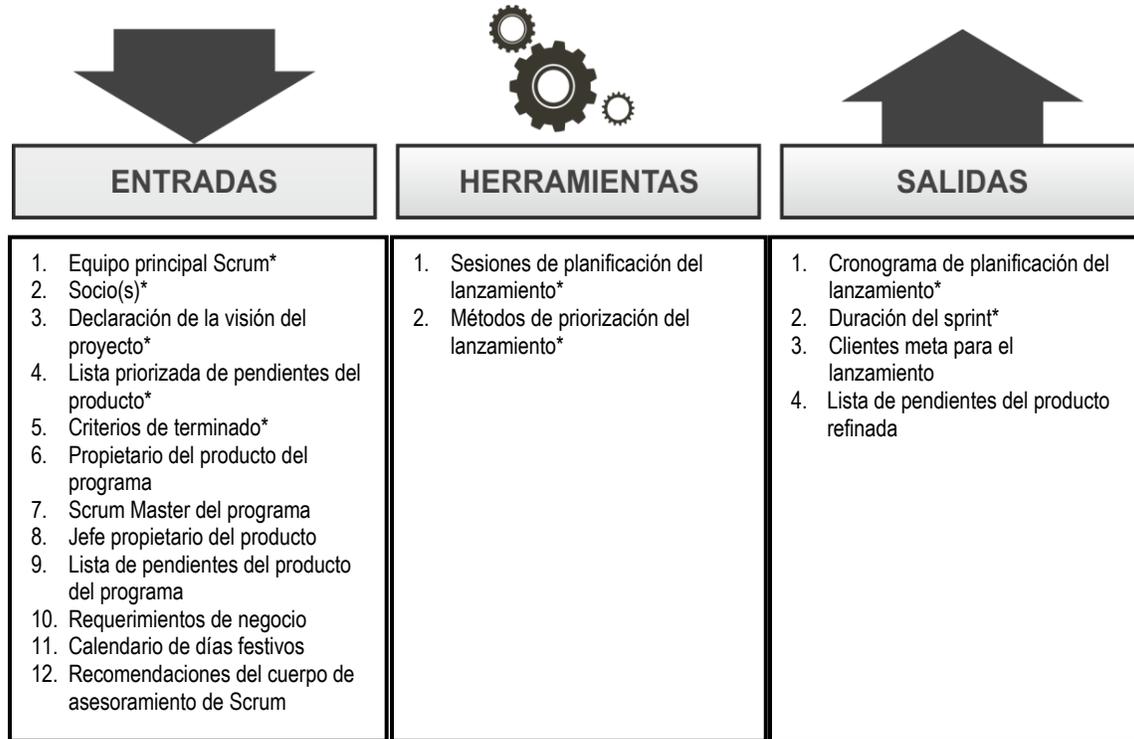
Proyecto: Diseñar las nuevas variantes de un popular auto deportivo en la empresa LRA Ltd.

*Criterios de terminado:*

- El diseño es aprobado por la división de excelencia técnica.
- El prototipo pasa todas las pruebas de túnel de viento mandado por la división de Aerodinámica.
- El diseño se aprueba para la producción por la división de Propiedad Intelectual.
- Las expectativas de diseño de seguridad son corroboradas por el informe de Diseño de Seguridad de la División de Seguridad.
- El informe de estimación de costos para el diseño es aprobado por la división de Finanzas.

## 8.6 Realizar el plan de lanzamiento

La Figura 8-14 muestra todas las entradas, herramientas y salidas para el proceso de *realizar la planificación del lanzamiento*.



**Figura 8-14: Realizar la planificación del lanzamiento—Entrada, herramientas y salidas**

*Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.*

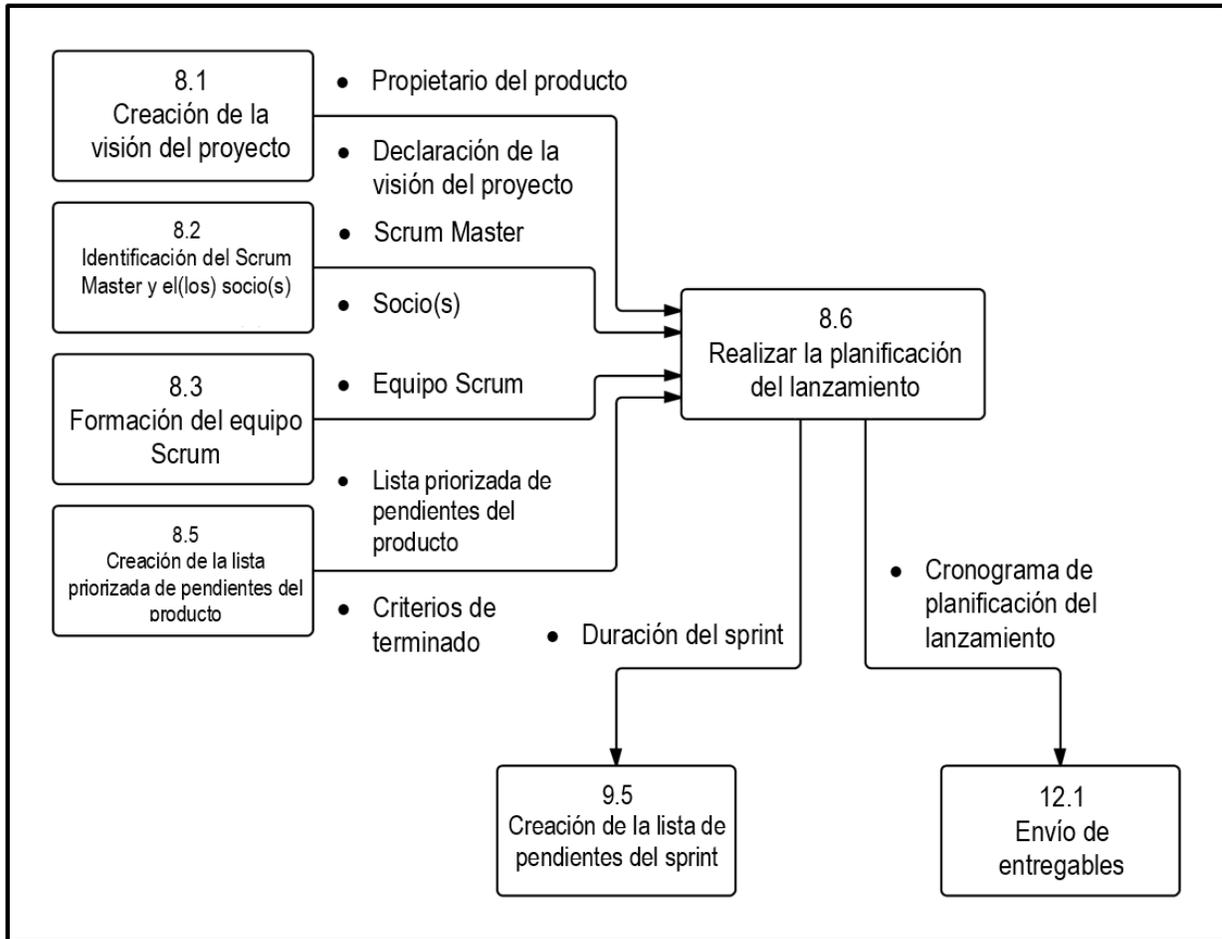


Figura 8-15: Realizar el plan de lanzamiento—Diagrama de flujo de datos

## **8.6.1 Entradas**

### **8.6.1.1 Equipo principal de Scrum\***

Descrito en la sección 8.4.1.1.

### **8.6.1.2 Socio(s)\***

Descrito en la sección 8.2.3.2.

### **8.6.1.3 Reunión de la visión del proyecto\***

Descrito en la sección 8.1.3.2.

### **8.6.1.4 Lista priorizada de pendientes del producto\***

Descrito en la sección 8.5.3.1.

### **8.6.1.5 Criterios de terminado\***

Descrito en la sección 8.5.3.2.

### **8.6.1.6 Propietario del producto del programa**

Descrito en la sección 8.1.1.2.

### **8.6.1.7 Scrum Master del programa**

Descrito en la sección 8.1.1.3.

### **8.6.1.8 Jefe propietario del producto**

Descrito en la sección 8.1.1.5.

### **8.6.1.9 Lista de pendientes del producto del programa**

Descrito en la sección 8.1.1.6.

### 8.6.1.10 Requerimientos del programa

Descrito en la sección 8.5.1.7.

### 8.6.1.11 Calendario de días feriados

Es importante para el equipo Scrum estar al tanto de las fechas claves y la disponibilidad de todos los miembros del equipo. Esto se puede lograr mediante el uso de un calendario compartido que proporciona información sobre los días feriados, ausencias, faltas al trabajo, planes de viaje, eventos, etc. Este calendario ayudará al equipo en la planificación y ejecución de los sprints.

### 8.6.1.12 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12

En el proceso de *Realizar la planificación del lanzamiento*, las recomendaciones de *Cuerpo de asesoramiento de Scrum* pueden relacionarse con las normas, reglamentos, estándares y mejores prácticas para el desarrollo del Plan de Lanzamiento. El Cuerpo de Orientación puede ser la mejor autoridad para definir las directrices relacionadas con el valor del negocio, las expectativas de liberación, estrategias de implementación, la calidad y la seguridad.

8

## 8.6.2 Herramientas

### 8.6.2.1 Sesiones de planificación del lanzamiento\*

Las reuniones de planificación del lanzamiento se llevan a cabo para desarrollar un plan del lanzamiento. El plan define cuando varios conjuntos de funcionalidad o productos utilizables serán entregados al cliente. En Scrum, el objetivo principal de una reunión de planificación de lanzamiento es hacer que el equipo Scrum tenga una visión general de los lanzamientos y el calendario de entrega del producto que están desarrollando para que puedan alinearse con las expectativas del propietario del producto y los socios relevantes (principalmente el patrocinador del proyecto).

Muchas organizaciones tienen una estrategia en relación con el lanzamiento de los productos. Algunas organizaciones prefieren un despliegue continuo, donde se produce un lanzamiento después de la creación de la funcionalidad útil especificada. Otras organizaciones prefieren un despliegue por etapas, donde los lanzamientos se hacen en intervalos predefinidos. Dependiendo de la estrategia de la organización, la sesión de planificación del lanzamiento en los proyectos puede ser impulsado por la funcionalidad, en el que el objetivo es hacer un lanzamiento una vez que un conjunto predeterminado de funcionalidades se ha desarrollado; o la planificación puede ser impulsada por la fecha, en la que ocurre el lanzamiento en una fecha predefinida.

Dado a que el marco de Scrum promueve la planificación con base en información y la práctica de decisiones iterativas, sobre la planificación detallada por adelantado como en la gestión de proyectos tradicional de estilo en Cascada, no es necesario que la sesión de planificación del lanzamiento elabore un plan de lanzamiento detallado de todo el proyecto. En Scrum, el plan de lanzamiento puede actualizarse continuamente a medida que la información relevante está disponible.

### **8.6.2.2 Métodos de priorización del lanzamiento\***

Los métodos de priorización del lanzamiento se utilizan para desarrollar un plan del lanzamiento. Estos métodos son específicos a la industria y organización, y generalmente son determinados por la alta gerencia de la organización.

## **8.6.3 Salidas**

### **8.6.3.1 Cronograma de planificación del lanzamiento\***

Un cronograma de planificación del lanzamiento es una de las salidas más importantes del proceso llamado realizar la planificación del lanzamiento. Un cronograma de planificación del lanzamiento indica que los entregables van a ser lanzadas a los clientes, junto con intervalos planificados y fechas para los lanzamientos. Puede que no haya un lanzamiento programado a finales de cada iteración del sprint. A veces, un lanzamiento puede ser planificado después de que un grupo de iteraciones del sprint se ha completado. Dependiendo de la estrategia de la organización, las sesiones de la planificación del lanzamiento en los proyectos pueden ser impulsadas por la funcionalidad en la que el objetivo es la entrega una vez que un conjunto predeterminado de funcionalidades se ha desarrollado, o la planificación puede ser impulsada por la fecha, en la que ocurre el lanzamiento en una fecha predefinida. El entregable de debe lanzar cuando ofrece suficiente valor empresarial para el cliente.

### **8.6.3.2 Duración del sprint\***

Basado en las diversas entradas que incluyen los requerimientos de negocio y el cronograma de planificación del lanzamiento, el propietario del producto y el equipo Scrum deciden respecto a la duración del sprint para el proyecto. Una vez determinada, la duración del sprint sigue siendo con frecuencia la misma durante todo el proyecto.

Sin embargo, la duración del sprint puede cambiarse si, y como, lo consideren apropiado el propietario del producto y el equipo Scrum. Puede ser que en una etapa temprana del proyecto todavía estén experimentando para encontrar el mejor tiempo de duración del sprint. Si más adelante en el proyecto hay un cambio en la duración del sprint, típicamente este sería una reducción del sprint debido a las mejoras en el entorno del proyecto. Un sprint podría ser asignado a un bloque de tiempo de 1 a 6 semanas. Sin embargo, para obtener los máximos beneficios de un proyecto Scrum, siempre se recomienda mantener el sprint en un bloque de tiempo de 4 semanas, a menos que existan proyectos con requisitos muy estables, donde los sprints pueden extenderse hasta 6 semanas.

El impacto de un cambio esperado en la duración del sprint se describe en la sección 6.5.1

### 8.6.3.3 Clientes meta para el lanzamiento

No todos los lanzamientos se dirigirán a todos los socios o usuarios. El(los) socio(s) puede(n) optar por limitar ciertos lanzamientos a un subconjunto de usuarios. El plan de lanzamiento debe especificar los clientes en quienes se va a enfocar el lanzamiento.

### 8.6.3.4 Lista priorizada de pendientes del producto refinada

La lista priorizada de pendientes del producto, desarrollada en el proceso de *creación de la lista priorizada de pendientes del producto*, se puede refinar en este proceso. Es posible que haya más claridad sobre las historias de usuarios en la lista priorizada de pendientes del producto después de que el equipo principal de Scrum lleve a cabo las sesiones de planificación del lanzamiento con el(los) socio(s).

## 8.7 Diagrama de la fase del flujo de datos

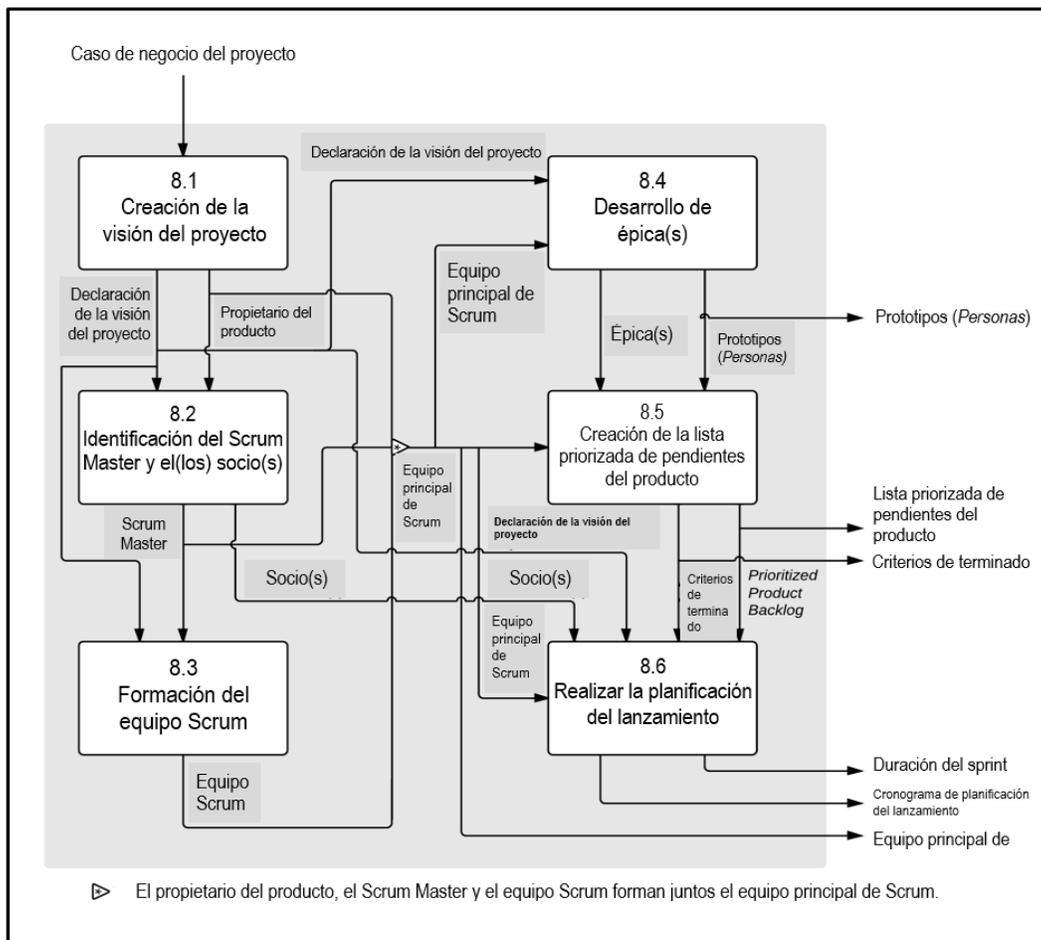


Figura 8-16: Fase de inicio—Diagrama del flujo de datos

## 9. PLANIFICACIÓN Y ESTIMACIÓN

La fase de planificación y estimación consiste en procesos relacionados a la planificación y estimación de tareas, los cuales incluyen: *Creación de historias de usuario; Aprobación, estimación y asignación de historias de usuario; Creación de tareas; Estimación de tareas y Creación de lista de pendientes del sprint.*

La *planificación y estimación*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se les entregará a los socios;
- Proyectos de cualquier tamaño o complejidad.

El término “producto” en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicarse de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Para facilitar la mejor aplicación del marco de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como “obligatorias” u “opcionales”. Las entradas, herramientas y salidas indicadas con asteriscos (\*) son obligatorias, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el equipo Scrum y aquellas personas que empiezan a aprender sobre el marco y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los propietarios del producto, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™*, no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está escrito desde la perspectiva de un equipo Scrum que trabaja en un sprint para producir entregables que se pueden enviar como parte de un proyecto más amplio. Sin embargo, la información que se describe aplica de igual forma a proyectos enteros, programas y carteras. Hay información adicional relacionada con el uso de Scrum para proyectos, programas y portafolios que está disponible en los capítulos 2 al 7, y que cubren los principios y aspectos de Scrum.

La figura 9-1 brinda una descripción general de los procesos en la fase de planificación y estimación, que son los siguientes:

**9.1 Creación de historias de usuario**—En este proceso, se crean las historias de usuario y sus respectivos criterios de aceptación. Las historias de usuario generalmente las escribe el propietario del producto y se diseñan para garantizar que los requerimientos del cliente estén claramente representados y que todos los socios las pueden entender completamente. Se pueden llevar a cabo talleres de redacción de historias de usuario donde se involucre a los miembros del equipo Scrum en la creación de dichas historias. Tales historias se incorporan en la lista priorizada de pendientes del producto.

**9.2 Aprobación, estimación y asignación de historias de usuario**—En este proceso, el propietario del producto aprueba las historias de usuario para un sprint. Después, el Scrum Master y el equipo Scrum estiman el trabajo que será necesario para desarrollar la funcionalidad descrita en cada historia de usuario. Por último, el equipo Scrum se compromete a entregar los requerimientos del cliente en forma de historias de usuario aprobadas, estimadas y asignadas.

**9.3 Creación de tareas**—En este proceso, las historias de usuario aprobadas, estimadas y asignadas se dividen en tareas específicas y se copilan en una lista de tareas. Generalmente, se lleva a cabo una reunión de planificación de tareas para tales efectos.

**9.4 Estimación de tareas**—En este proceso, el equipo principal de Scrum, durante las reuniones de estimación de tareas, calcula el trabajo que será necesario para cumplir con cada tarea en la lista. Este proceso da como resultado una lista de tareas del esfuerzo estimado.

**9.5 Creación de la lista de pendientes del Sprint**—En este proceso, el equipo principal de Scrum lleva a cabo reuniones de planificación del sprint, donde el grupo elabora una lista de pendientes del sprint la cual incluye todas las tareas que habrán de completarse en el sprint.

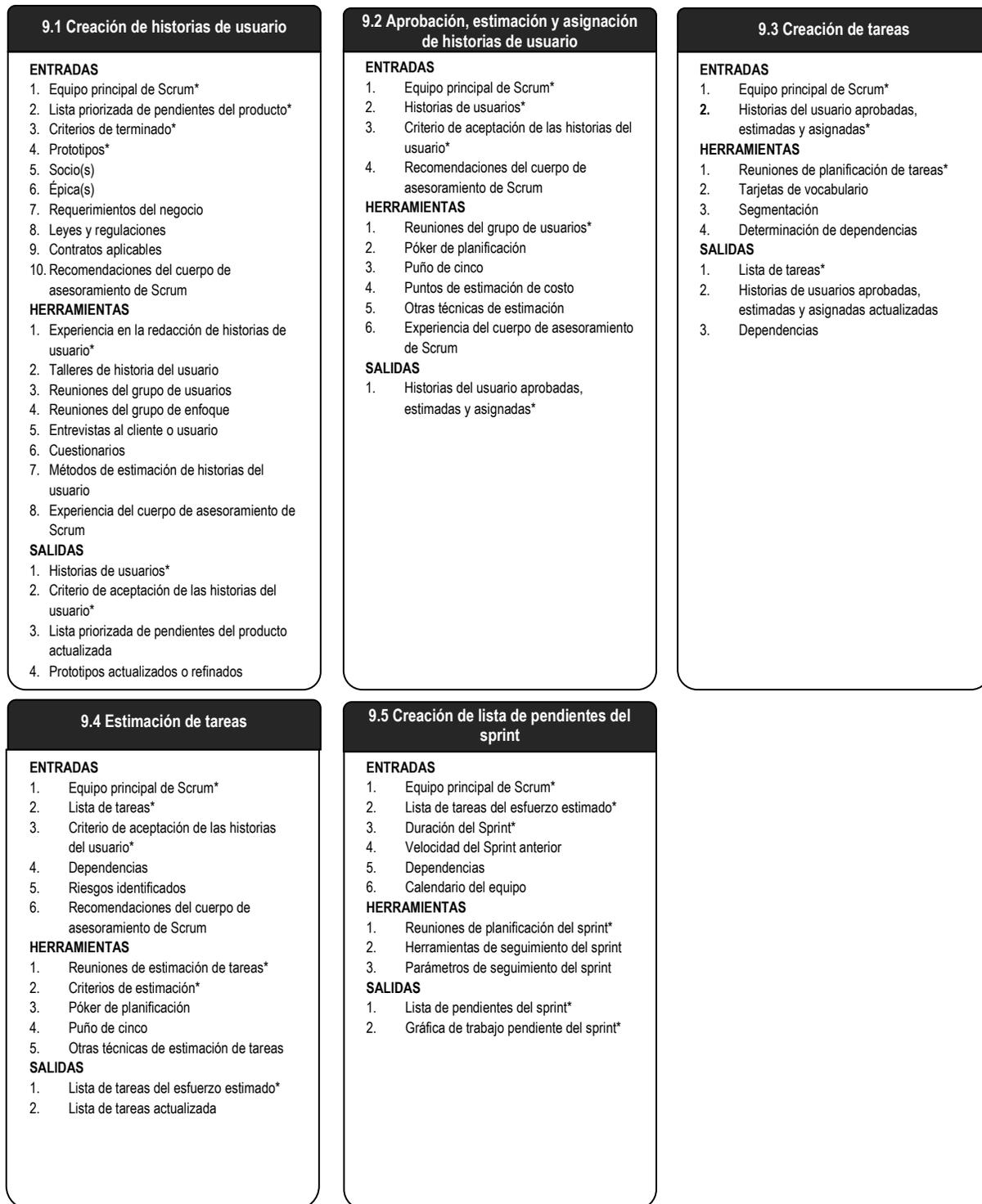


Figura 9-1: Resumen de la planificación y estimación

Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

La figura 9-2 muestra las entradas obligatorias, las herramientas y las salidas de los procesos en la fase de planificación y estimación.

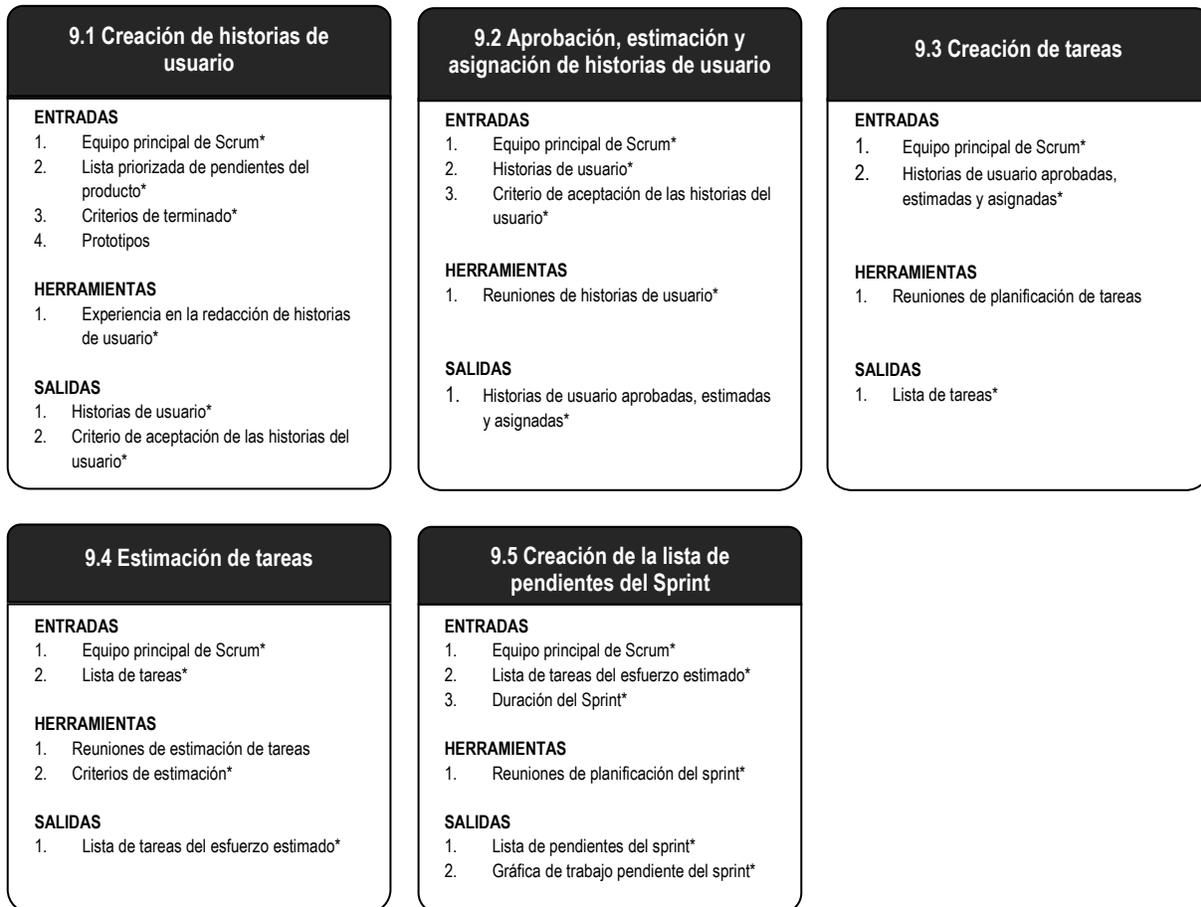


Figura 9-2: *Plan and Estimate* General (Esenciales)

Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.

## 9.1 Creación de historias de usuario

La figura 9-3 muestra todas las entradas, las herramientas y las salidas para el proceso de creación de historias de usuario.



Figura 9-3: Creación de historias de usuario—Entradas, herramientas y salidas

*Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.*

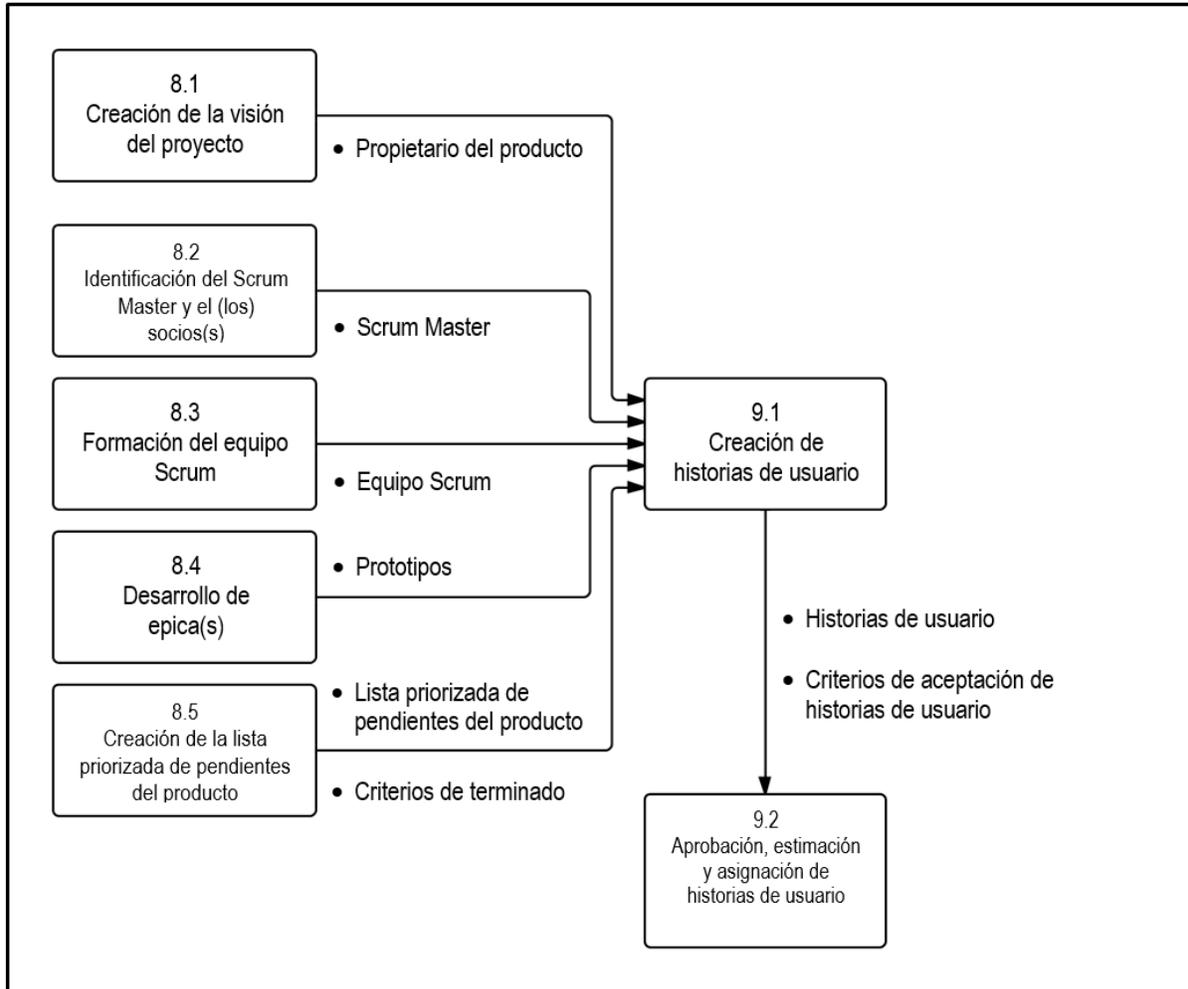


Figura 9-4: Creación de historias de usuario—Diagrama de flujo de datos

## **9.1.1 Entradas**

### **9.1.1.1 Equipo principal de Scrum\***

Descrito en la sección 8.4.1.1.

### **9.1.1.2 Lista priorizada de pendientes del producto\***

Descrito en la sección 8.5.3.1.

### **9.1.1.3 Criterios de terminado\***

Descrito en la sección 8.5.3.2.

### **9.1.1.4 Prototipos\***

Descrito en la sección 8.4.3.2.

### **9.1.1.5 Socio(s)**

Descrito en la sección 8.2.3.2.

### **9.1.1.6 Épica(s)**

Descrito en la sección 8.4.3.1.

### **9.1.1.7 Requerimientos del negocio**

Descrito en la sección 8.5.1.7.

### **9.1.1.8 Leyes y regulaciones**

Descrito en la sección 8.4.1.8.

### **9.1.1.9 Contratos aplicables**

Descrito en la sección 8.4.1.9.

### 9.1.1.10 Recomendaciones de cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

En el proceso de creación de historias de usuario, las recomendaciones del cuerpo de asesoramiento de Scrum pueden incluir información sobre las reglas, regulaciones, estándares y mejores prácticas necesarias para crear historias de usuario efectivas.

## 9.1.2 Herramientas

### 9.1.2.1 Experiencia en la redacción de historias de usuario\*

El propietario del producto —con base en su interacción con los socios, en su conocimiento del negocio, en la experiencia y en las aportaciones del equipo—, desarrolla las historias de usuario que habrán de formar la lista inicial de pendientes del producto para el proyecto. La lista priorizada de pendientes del producto representa la suma total de lo que debe completarse en el proyecto. El objetivo de este ejercicio es crear historias de usuario elaboradas y refinadas que pudieran ser aprobadas, estimadas y asignadas por el equipo Scrum. En ocasiones, el propietario del producto pudiera incluir a un analista empresarial para que ayude en la redacción de historias de usuario.

Aunque el propietario del producto tiene la principal responsabilidad de escribir las historias de usuario y generalmente lleva a cabo esta actividad por sí mismo, se puede también llevar a cabo un taller de redacción de historias de usuario si así se desea.

### 9.1.2.2 Talleres de historias de usuario

Descrito en la sección 8.4.2.2.

### 9.1.2.3 Reuniones de grupo de usuarios

Descrito en la sección 8.4.2.1.

### 9.1.2.4 Reuniones de grupo de enfoque

Las reuniones de grupos de enfoque son una técnica cualitativa para cuantificar y entender las necesidades y expectativas del usuario respecto al producto que se propone. Se selecciona un pequeño grupo de usuarios para que formen parte del grupo de enfoque. Este grupo pudiera ser seleccionado en forma aleatoria de entre muchos usuarios o bien, puede seleccionarse específicamente para representar a todos los prototipos de cliente (*personas*) meta. Las reuniones de grupo de enfoque normalmente se apegan a cierto formato en el cual se le hacen preguntas que después analizan entre ellos. Cada reunión de grupo de

enfoque puede contar con su propio reglamento de discusiones según lo hayan decidido los organizadores. Dichas reuniones generalmente se llevan a cabo con la presencia de un moderador.

#### **9.1.2.5 Entrevistas al cliente o usuario**

Descrito en la sección 8.4.2.4.

#### **9.1.2.6 Cuestionarios**

Descrito en la sección 8.4.2.5.

#### **9.1.2.7 Métodos de estimación de historias de usuario**

Todas las herramientas utilizadas en el proceso de aprobación, estimación y asignación de historias de usuario (descritas en la sección 9.2.2) se pueden implementar para crear estimaciones de alto nivel para las épicas al momento de crear la lista priorizada de pendientes del producto. Algunas herramientas importantes son:

1. Reunión del grupo de usuarios\*
2. Póker de planificación
3. Puño de Cinco
4. Puntos de estimación de costo
5. Otras técnicas de estimación

#### **9.1.2.8 Experiencia del cuerpo de asesoramiento de Scrum**

Descrito en la sección 8.4.2.7.

Al crear historias de usuario, la experiencia del cuerpo de asesoramiento de Scrum podría referirse a las normas y regulaciones o a los estándares y mejores prácticas para crear historias de usuario. Pudiera haber también un equipo de expertos en la materia que pudiera ayudar al propietario del producto o brindar orientación sobre cómo crear historias de usuario. Dicho equipo pudiera incluir a analistas empresariales, arquitectos líderes, desarrolladores experimentados, expertos en Scrum o demás personas con experiencia. Este grupo no es generalmente el mismo equipo que permanecerá y trabajará en el proyecto, ya que tienden a trasladarse de proyecto en proyecto brindando orientación a los equipos Scrum si así lo requieren.

### 9.1.3 Salidas

#### 9.1.3.1 Historias de usuario\*

Las historias de usuario se apegan a una estructura específica predefinida y son una forma simple de documentar los requerimientos y funcionalidades que desea el usuario final. Una historia de usuario incluye tres elementos sobre el requerimiento: ¿Quién? ¿Qué? y ¿Por qué? Los requerimientos expresados en las historias de usuario son oraciones breves, sencillas y fáciles de entender. El formato estándar predefinido da como resultado en una comunicación mejorada entre los socios, así como en mejores estimaciones por parte del equipo. Algunas historias de usuario tal vez sean demasiado extensas como para poderse manejar dentro un solo sprint. A estas amplias historias generalmente se les llama épicas. Una vez que las épicas surgen en la lista priorizada de pendientes del producto para completarse en otro sprint, se fragmentan aún más en historias de usuario.

La lista priorizada de pendientes del producto es una lista dinámica que se actualiza constantemente debido a la re-priorización de historias de usuarios nuevas, actualizadas, refinadas y en ocasiones eliminadas. Tales actualizaciones a la lista de pendientes son generalmente el resultado de cambios en los requerimientos del negocio.

Véase también la sección 8.5.3.1 para conocer más sobre la lista priorizada de pendientes del producto.

Formato de historia de usuario:

Como <rol/prototipo de cliente>, yo debería <requerimiento> a fin de <beneficio>.

Ejemplo de historia de usuario:

Como administrador de una base de datos, yo debería contar con la capacidad de revertir una cantidad selecta de actualización de la base de datos a fin de que se restablezca a la versión deseada.

#### 9.1.3.2 Criterios de aceptación de historias de usuario\*

Cada historia de usuario cuenta con sus respectivos criterios de aceptación. Las historias de usuario son subjetivas de tal forma que los criterios de aceptación brindan la objetividad requerida para que las historias de usuario se consideren terminadas o no terminadas durante la revisión del sprint. Los criterios de aceptación brindan claridad al equipo respecto a los que se espera en una historia de usuario; eliminan la ambigüedad de los requerimientos, ayudando a la alineación de las expectativas. El propietario del producto define y comunica los criterios de aceptación al equipo Scrum. En las reuniones de revisión del sprint, los criterios de aceptación brindan el contexto para que el propietario del producto decida si la historia de usuario se ha completado satisfactoriamente. Es responsabilidad del Scrum Master asegurar que el propietario del producto no cambie los criterios de aceptación de una historia de usuario asignada a mitad de un sprint.

### 9.1.3.3 Lista de pendientes de producto actualizada

La lista priorizada de pendientes del producto elaborada en el proceso de su creación, se actualiza con la información en las historias de usuario, épicas, estimaciones para historias de usuario, así como los criterios de aceptación de las mismas.

La lista priorizada de pendientes del producto se describe en la sección 8.5.3.1.

### 9.1.3.4 Prototipos actualizados o refinados

Los prototipos (del inglés: *personas*) se elaboran en el proceso de desarrollo de épica(s). Al momento de redactar las historias de usuario, el equipo Scrum puede llegar a una decisión colectiva de que dichos prototipos iniciales son inadecuados y deben refinarse. En caso de que exista la necesidad de refinar los prototipos, esto se lleva a cabo normalmente casi al final del proceso de creación de historias de usuario.

Los prototipos se describen en la sección 8.4.3.2.

## 9.2 Aprobación, estimación y asignación de historias de usuarios

La figura 9-5 muestra todas las entradas, las herramientas y las salidas para el proceso de aprobación, estimación y asignación de historias de usuario.

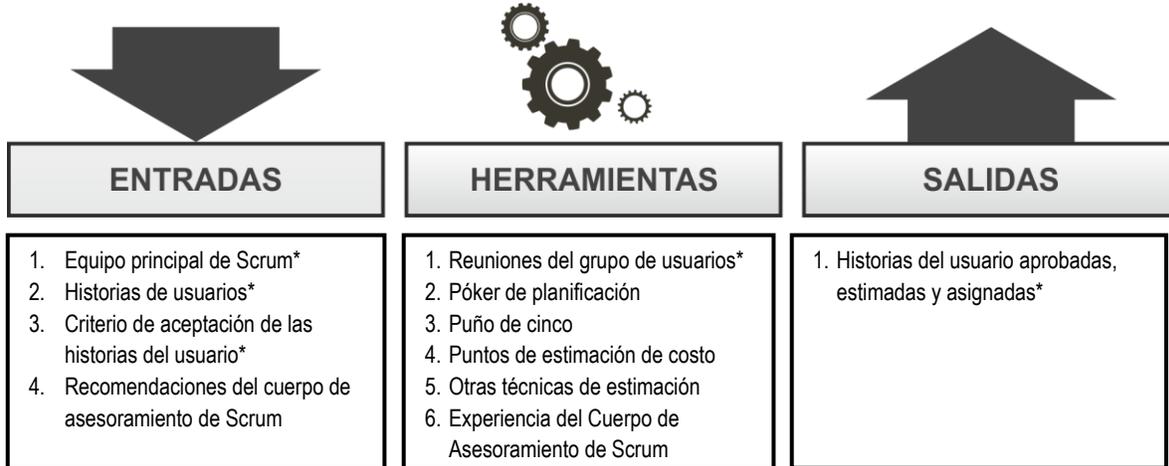


Figura 9-5: Aprobación, estimación y asignación de historias de usuario—Entrada, herramientas y salidas

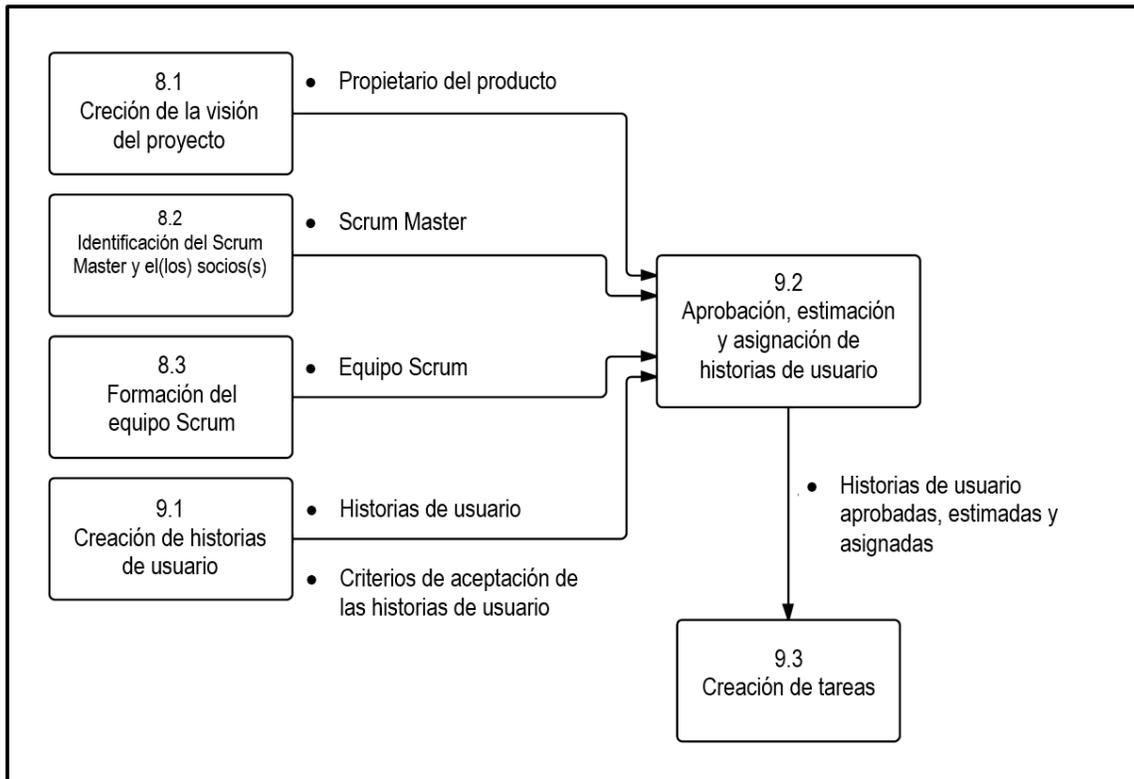


Figura 9-6: Aprobación, estimación y asignación de historias de usuario— Diagrama de flujo de datos

Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.

## 9.2.1 Entradas

### 9.2.1.1 Equipo principal de Scrum\*

Descrito en la sección 8.4.1.1.

### 9.2.1.2 Historias de usuario\*

Descrito en la sección 9.1.3.1.

Las historias de usuario cuentan con estimaciones de alto nivel en los procesos de creación de la lista priorizada de pendientes del producto y de creación de historias de usuario. Dichas estimaciones las utilizaría el propietario del producto para crear una lista de historias de usuario aprobadas, mismas que serían estimadas con mayor precisión por el equipo Scrum. Tales estimaciones se asignan después para que el equipo Scrum las complete en el sprint.

### 9.2.1.3 Criterios de aceptación de historias de usuario\*

Descrito en la sección 9.1.3.2.

### 9.2.1.4 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

En el proceso de aprobación, estimación y asignación de historias de usuario, las recomendaciones del cuerpo de asesoramiento de Scrum pudieran incluir información sobre las reglas, regulaciones, estándares y mejores prácticas necesarias para aprobar, estimar y asignar eficazmente las historias de usuario.

## 9.2.2 Herramientas

### 9.2.2.1 Reuniones de grupo de usuarios\*

Descrito en la sección 8.4.2.1.

### 9.2.2.2 Póker de planificación

El póker de planificación, conocido también como póker de estimación, es una técnica de estimación que implementa el consenso para estimar los tamaños relativos de las historias de usuario o el trabajo necesario para desarrollarlos.

En el póker de planificación, a cada miembro del equipo se le asigna una baraja. Cada carta está enumerada en forma secuencial y los números representan la complejidad del problema en términos de tiempo o esfuerzo, según lo estimado por el miembro del equipo. El propietario del producto elige una historia de usuario de la lista priorizada de pendientes del producto y se la presenta al equipo. Los miembros del equipo Scrum evalúan la historia de usuario e intentan entenderla mejor antes de brindar su estimación para su desarrollo. Después, cada miembro elige una carta de la baraja que represente su estimación para la historia de usuario. Si la mayoría, o todos los miembros del equipo seleccionan la misma carta, entonces el cálculo que indique la carta será el estimado para esta historia de usuario. Si no hay un consenso, entonces los integrantes del equipo discuten las razones de la selección de distintas cartas o estimaciones. Después del análisis seleccionan nuevamente las cartas. Esta secuencia continúa hasta que se entienden todas las presuposiciones; hasta que se resuelven los malentendidos o hasta que se llega a un consenso o a un acuerdo.

El póker de planificación promueve una mayor interacción y una mejor comunicación entre los participantes. Facilita el pensamiento independiente por parte de los participantes, evitando con ello el fenómeno del pensamiento en grupo.

### 9.2.2.3 Puño de cinco

9

El puño de cinco es un mecanismo sencillo y rápido para lograr el consenso en un grupo y guiar una conversación. Tras el debate inicial sobre una propuesta o una decisión pendiente, se les pide a los miembros del equipo Scrum que voten en una escala de 1 a 5 utilizando sus dedos. El valor en el uso de esta técnica no es sólo la creación de consenso, sino también la reflexión y charla, ya a que a cada miembro del equipo se le pide que explique el motivo de su clasificación. También se les da la oportunidad de expresar cualquier problema o preocupación. Una vez que el equipo ha discutido, se toma una decisión colectiva.

El número de dedos que se utiliza para la votación indica el nivel de acuerdo y el deseo para el debate:

1. Un dedo: No estoy de acuerdo con la conclusión del grupo y tengo grandes inquietudes.
2. Dos dedos: No estoy de acuerdo con la conclusión del grupo y me gustaría hablar sobre algunos asuntos menores.
3. Tres dedos: No estoy seguro y me gustaría sumarme a la conclusión de consenso del grupo.
4. Cuatro dedos: Estoy de acuerdo con la conclusión del grupo y me gustaría discutir algunos asuntos menores.
5. Cinco dedos: Estoy totalmente de acuerdo con la conclusión del grupo.

#### 9.2.2.4 Puntos de estimación de costo

La estimación de costos se puede lograr mediante el uso de unidades relativas (por ejemplo: las estimaciones de esfuerzo) en lugar de unidades absolutas (los costos reales incurridos). A fin de estimar los costos para implementar una historia de usuario, el equipo Scrum puede utilizar puntos de historia. Cuando se da este caso, el costo estimado de cada tarea se representará en forma de puntos de historia en vez de unidades monetarias. A fin de lograrlo satisfactoriamente, el equipo Scrum debe identificar una historia de usuario de referencia con la que se puedan identificar todos los miembros del equipo. Una vez identificada dicha base, todas las estimaciones de costo de las historias de usuario deben compararse con la base. Dichos cálculos permanecen fijos durante el sprint debido a que los equipos no deben cambiar durante un sprint.

#### 9.2.2.5 Otras técnicas de estimación

##### 9.2.2.5.1 Wideband Delphi

Wideband Delphi es una técnica de estimación basada en grupo para determinar la cantidad de trabajo necesario y el tiempo que tardará en completarse. Los individuos en el equipo proporcionan estimaciones anónimas para cada característica y las estimaciones iniciales se trazan en una gráfica. Posteriormente, el equipo analiza los factores que influyeron en sus estimaciones y proceden a una segunda ronda de estimación. Este proceso se repite hasta que las estimaciones de los individuos quedan cerca una de la otra y se puede llegar a un consenso para la estimación final.

El póker de planificación (tal como se describe en la sección 9.2.2.2) es un ejemplo de una técnica Wideband Delphi. También es importante destacar que la opinión individual recopilada por un mecanismo, lo cual evita el pensamiento en grupo. Después, las opiniones individuales se utilizan para una decisión colectiva.

##### 9.2.2.5.2 Tamaño relativo/Puntos de historia

Además de utilizarse para la estimación de costos, los puntos de historia (*Story Points*), pueden emplearse también para calcular el tamaño total de una historia de usuario o de una característica. Este método asigna un punto de historia con base en una evaluación general del tamaño de la historia de usuario tomando en cuenta el riesgo, la cantidad requerida de actividades y el nivel de complejidad. Esta evaluación la llevará a cabo el equipo Scrum y se asignará un valor de punto de historia. Una vez realizada la evaluación en una de las historias de usuario en la lista priorizada de pendientes del producto, el equipo Scrum puede evaluar otras historias de usuario relativas a la primera. Por ejemplo, una característica con un valor de 2 puntos de historia debe ser doblemente difícil de completar que una característica con un punto de historia de 1; por otra parte, un valor de 3 en puntos de historia sería tres veces más difícil de completar que la característica con valor de 1.

### 9.2.2.5.3 Estimación por afinidad

La estimación por afinidad (del inglés: *Affinity Estimation*) es una técnica que se utiliza para estimar rápidamente un gran número de historias de usuarios con el uso de categorías. Utilizando notas adhesivas o fichas y cinta, cada equipo coloca las historias de usuario en la pared o en cualquier otra superficie en orden desde la más pequeña hasta la más amplia. Para ello, cada integrante del equipo inicia con un subconjunto de historias de usuario de toda la lista priorizada de pendientes del producto para colocarse por tamaño relativo. Esta colocación inicial se hace en silencio. Una vez que todos han colocado en la pared sus historias de usuario, el equipo las revisa y las puede mover según sea necesario. Esta segunda parte del ejercicio incluye discusiones. Por último, el propietario del producto indicará en la pared algunas categorías de tamaño. Dichas categorías pueden ser pequeñas, medianas o grandes, o bien, pueden estar enumeradas utilizando valores de punto de la historia (*point story values*) para indicar el tamaño relativo. Después el equipo reubicará las historias de usuario en dichas categorías en el paso final del proceso. Algunos de los beneficios claves de este método son que el proceso es muy transparente, visible para todos y fácil de llevar a cabo.

### 9.2.2.5.4 Rango de estimación

Las estimaciones de los proyectos deben presentarse en rangos. Las cifras exactas pueden dar la impresión de ser muy específicas cuando en realidad tal vez no lo sean. De hecho, se entiende que las estimaciones, por su definición, no son precisamente exactas. Los rangos de estimación deben basarse en el nivel de confianza del equipo en cada estimación que se haga. El rango puede ser reducido cuando el equipo está confiando y amplio cuando el equipo se siente menos seguro.

### 9.2.2.6 Experiencia del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.4.2.7.

Los conflictos relacionados con las estimaciones para completar ciertas historias de usuario pudieran surgir durante este proceso debido a que las perspectivas de los miembros del equipo pudieran diferir y debido a que tienen tal vez suficiente experiencia para estimar los sprints. En tales casos, la experiencia del cuerpo de asesoramiento puede ayudar en la resolución de conflictos.

## 9.2.3 Salidas

### 9.2.3.1 Historias del usuario aprobadas, estimadas y asignadas\*

Las historias de usuario que son entradas en este proceso tienen estimaciones de alto nivel de los procesos de *creación de la lista priorizada de pendientes del producto* y de *creación de historias de usuario*. Estas estimaciones las implementa el propietario del producto para aprobar las historias de usuario para el sprint.

Se debe destacar que es responsabilidad del propietario del producto garantizar que las historias de usuario otorguen valor y atiendan las necesidades y requerimientos del proyecto de los socios. Una vez aprobadas, las historias de usuario son estimadas por el equipo utilizando las diversas técnicas de estimación analizadas en esta sección. Después de la estimación, el equipo se avoca a un subconjunto de historias de usuario estimadas que consideran que pueden completar en el siguiente sprint. Estas historias de usuario se aprueban, se estiman y se asignan y formarán parte de la lista de pendientes del sprint.

Aunque el propietario del producto aprueba las historias de usuario iniciales para un sprint, la decisión final sobre cuáles historias específicas (entre las aprobadas por el propietario del producto) deben seleccionarse para el sprint, recae en el equipo Scrum. El equipo Scrum (consultando con el propietario del producto en caso de ser necesario), finaliza las historias de usuario en las que trabajarán durante el sprint.

## 9.3 Creación de tareas

La figura 9-7 muestra todas las entradas, las herramientas y las salidas del proceso de creación de tareas.

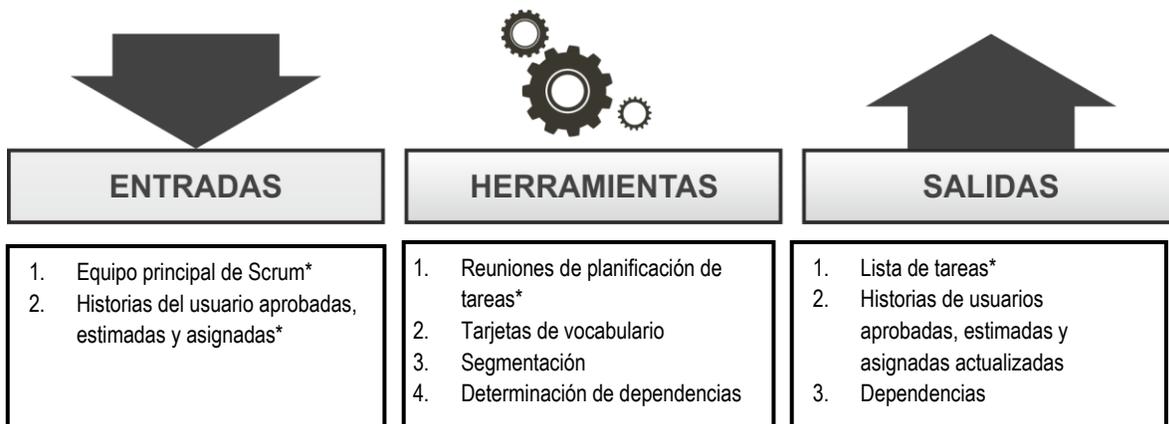


Figura 9-7: Creación de tareas—Entradas, herramientas y salidas

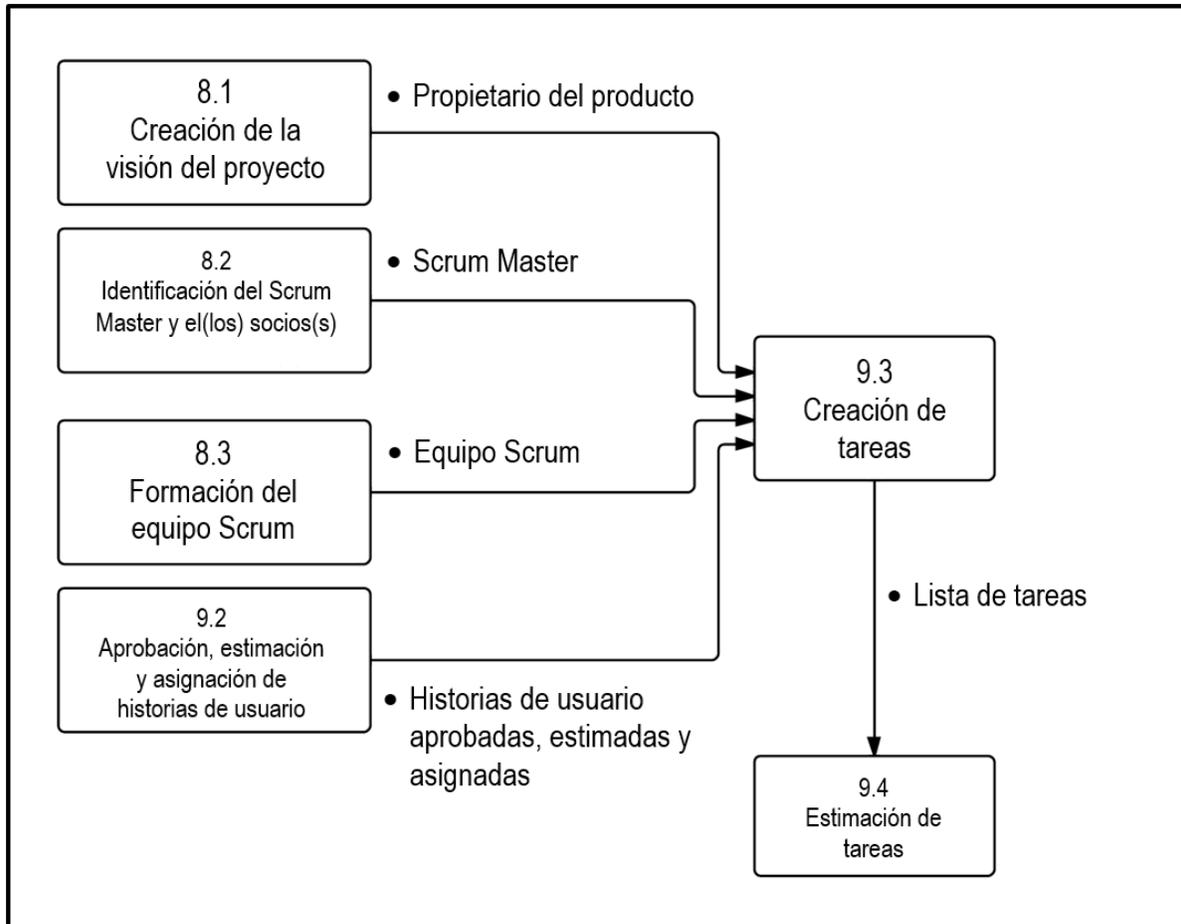


Figura 9-8: Creación de tareas—Diagrama de flujo de datos

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

### 9.3.1 Entradas

#### 9.3.1.1 Equipo principal de Scrum\*

Descrito en la sección 8.4.1.1.

#### 9.3.1.2 Historias de usuario aprobadas, estimadas y asignadas\*

Descrito en la sección 9.2.3.1.

## 9.3.2 Herramientas

### 9.3.2.1 Reuniones de planificación de tareas\*

En las reuniones de planificación de tareas, el equipo Scrum se reúne para planificar el trabajo que se hará en el sprint. El equipo revisa las historias de usuario asignadas que encabezan la lista priorizada de pendientes del producto. El propietario del producto se encuentra presente durante las reuniones en caso de ser necesaria una clarificación relacionada a las historias de usuario en la lista antes mencionada y para ayudar al equipo a tomar decisiones sobre el diseño. Para ayudar a garantizar que el equipo no se salga del tema, la reunión debe tener un bloque de tiempo asignado con una duración estándar limitada de dos horas por semana de duración del sprint. Esto ayuda a prevenir la tendencia de desviarse hacia discusiones que deberían realizarse en otras reuniones, tales como en las de planificación del lanzamiento o en reuniones de revisión del sprint. Al final de la reunión, todo el equipo Scrum se avocará completamente a entregar un subconjunto de historias de usuario de la lista priorizada de pendientes en el sprint.

La reunión de planificación de tareas se divide normalmente en dos secciones con un fin determinado y una amplia agenda para cada uno (véase la figura 9-9).

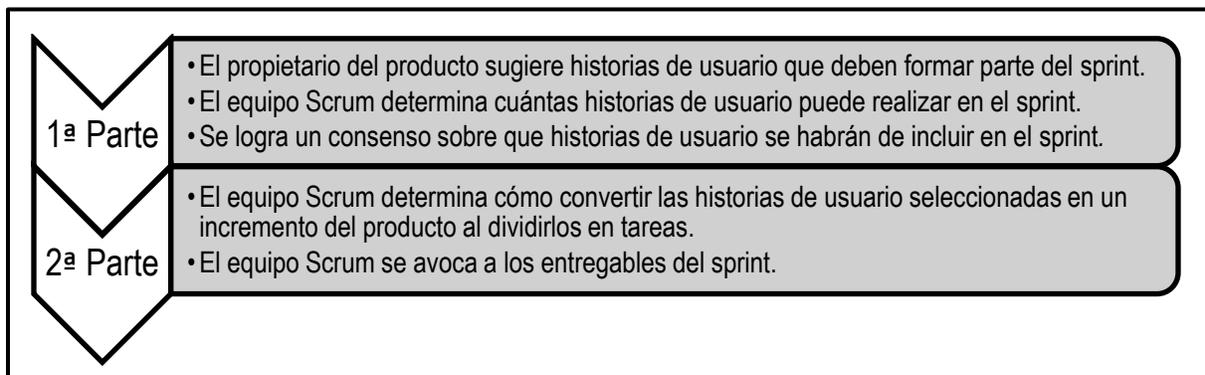


Figura 9-9: Reuniones de planificación de tareas

A las reuniones de planificación de tareas en ocasiones se les puede llamar “reuniones de planificación del sprint”. Tales reuniones se pueden combinar también con las reuniones de estimación de tareas, tal como se describe en la sección 9.4.2.1.

### 9.3.2.2 Fichas

En Scrum, las historias de usuario se escriben en pequeñas fichas (tarjetas). En las tarjetas únicamente se documentan los detalles esenciales, mismos que puede utilizar el equipo Scrum para colaborar y discutir. Estas tarjetas —generalmente llamadas tarjetas de historia—, aumentan la visibilidad y la transparencia, facilitando el descubrimiento oportuno de cualquier problema que pudiera surgir.

### 9.3.2.3 Segmentación

La segmentación (del inglés: *decomposition*) es una herramienta donde las tareas de alto nivel se dividen en niveles inferiores y más detallados. Los miembros del equipo Scrum segmentan las historias de usuario en tareas. Las historias de usuario de la lista priorizada de pendientes del producto deben estar suficientemente segmentadas a un nivel que le brinde al equipo Scrum la información adecuada para crear entregables a partir de las tareas mencionadas en la lista de tareas.

### 9.3.2.4 Determinación de dependencia

Una vez que el equipo Scrum ha seleccionado las historias de usuario para un determinado sprint, deben considerar las dependencias, incluyendo aquellas relacionadas a la disponibilidad de personal, así como cualquier dependencia técnica. Documentar adecuadamente las dependencias ayuda a los equipos Scrum a determinar el orden relativo en el cual deben ejecutarse las tareas para crear los entregables del sprint. Las dependencias destacan también la relación e interacción entre las tareas dentro del equipo Scrum que trabaja en un determinado Sprint y con otros equipos Scrum en el proyecto.

Existen numerosos tipos de dependencias: obligatorias y discrecionales; internas y externas, o alguna combinación de estas. Por ejemplo, una dependencia puede ser tanto obligatoria como externa.

- **Dependencias obligatorias**— Dependencias que son tanto inherentes en la naturaleza del trabajo (como una limitación física) como pueden darse debido a obligaciones contractuales o por requerimientos legales. Por ejemplo, el trabajo en el primer piso no puede iniciar hasta que se hayan terminado los cimientos del edificio. A las dependencias obligatorias también se les llama comúnmente como “lógica dura” (*hard logic*).
- **Dependencias discrecionales**—Las dependencias discrecionales son aquellas que se colocan en el flujo de trabajo por decisión propia. Normalmente, el equipo Scrum determina cuáles son las dependencias discrecionales con base en experiencias anteriores o las mejores prácticas en un campo o dominio específico. Por ejemplo, el equipo puede optar por completar una tarea antes de trabajar en otra ya que sería una mejor práctica, aunque no obligatoria. Por ejemplo, el equipo pudiera optar por construir marcos de puertas y ventanas antes de toda la pared esté en pie.
- **Dependencias externas**—Las dependencias externas son aquellas que están relacionadas a las tareas, actividades o productos fuera del alcance del trabajo a realizar por el equipo Scrum, pero que son necesarias para concluir una tarea del proyecto o crear un entregable del proyecto. Las dependencias externas generalmente están fuera del alcance del equipo Scrum. Por ejemplo, si el equipo Scrum no está a cargo de la adquisición de los materiales necesarios para la construcción de paredes, entonces dichos materiales y las tareas relacionadas a su adquisición se considerarían dependencias externas.

**Dependencias internas**—Las dependencias internas son aquellas entre las tareas, productos o actividades que están bajo el control del equipo Scrum y dentro de un enfoque de trabajo a ser ejecutado por dicho equipo. Por ejemplo, la instalación de paneles de yeso debe ser completada antes de pintar la pared. Este es un ejemplo de una dependencia interna debido a que ambas tareas son parte del proyecto. En este caso, también es obligatorio, ya que se basa en una limitación física. No es posible pintar la pared antes que la pared esté seca.

### 9.3.3 Salidas

#### 9.3.3.1 Lista de tareas\*

Lista completa que contiene todas las tareas a las que se ha abocado el equipo Scrum en el actual sprint. Contiene descripciones de cada tarea, así como las estimaciones derivadas durante el proceso de creación de tareas. La lista de tareas debe incluir cualquier prueba y actividad de integración de manera que el incremento del producto del sprint puede integrarse con éxito en los entregables de sprints anteriores.

Aunque las tareas por lo general se basan en actividades, el nivel de detalle al que se fragmentan las tareas lo decide el equipo Scrum.

#### 9.3.3.2 Historias de usuario, aprobadas, estimadas y asignadas actualizadas

En este proceso se actualizan las historias de usuario. Las actualizaciones pueden incluir revisiones de las estimaciones originales de las historias de usuario con base en la creación de tareas y en los factores de complejidad analizados durante la reunión de planificación del sprint. Las historias de usuario aprobadas, estimadas y asignadas se describen en la sección 9.2.3.1.

#### 9.3.3.3 Dependencias

Las dependencias describen la relación e interacción entre distintas tareas en un proyecto y pueden clasificarse como obligatorias o discrecionales; internas o externas, tal como se analiza en la sección 9.3.2.4.

Existen numerosas formas de identificar, definir y presentar las tareas y sus dependencias. Dos de los métodos comunes implican el uso de diagramas de flujo del producto y gráficas de Gantt.

## 9.4 Estimación de tareas

La figura 9-10 muestra todas las entradas, las herramientas y las salidas para el proceso de estimación de tareas.

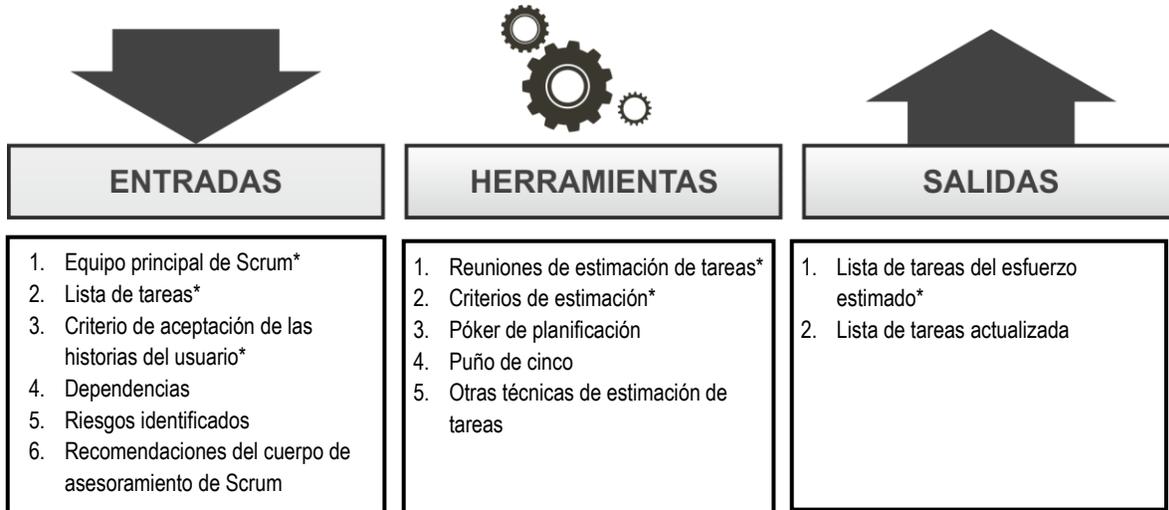


Figura 9-10: Estimación de tareas—Entradas, herramientas y salidas

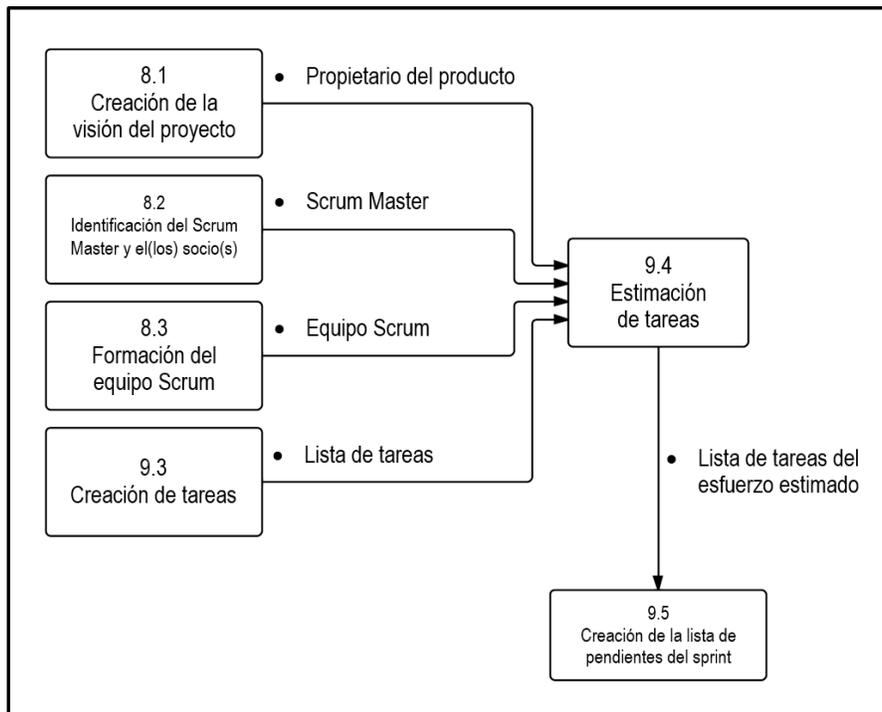


Figura 9-11: Estimación de tareas—Diagrama de flujo de datos

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

## 9.4.1 Entradas

### 9.4.1.1 Equipo principal de Scrum\*

Descrito en la sección 8.4.1.1.

### 9.4.1.2 Lista de tareas\*

Descrito en la sección 9.3.3.1.

### 9.4.1.3 Criterios de aceptación de las historias de usuario

Descrito en la sección 9.1.3.2.

El equipo Scrum debe asegurarse de que los criterios de aceptación definidos sean adecuados para las historias de usuario y brinden claridad respecto a los requerimientos para el equipo Scrum. Las pruebas de aceptación son la evaluación de la capacidad con la que cuentan los entregables completados para cumplir sus criterios de aceptación. Esto brinda información para que el propietario del producto pueda tomar una decisión sobre la aprobación o rechazo del entregable.

En el desarrollo de los criterios de aceptación de las historias de usuario, se debe tomar en cuenta lo siguiente:

- Los criterios de aceptación no deben ser vagos, ambiguos o demasiado generalizados.
- Los criterios de aceptación definidos garantizan que el equipo pueda verificar que los resultados estén alineados con las metas y objetivos de la organización patrocinadora.

### 9.4.1.4 Dependencias

Descrito en la sección 9.3.3.3

### 9.4.1.5 Riesgos identificados

Descrito en la sección 8.4.3.4.

### 9.4.1.6 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

En el proceso de aprobación, estimación y asignación de historias de usuario, las recomendaciones del cuerpo de asesoramiento de Scrum pudieran incluir información sobre las reglas, regulaciones, estándares y mejores prácticas necesarias para estimar eficazmente las tareas en la lista de tareas.

## 9.4.2 Herramientas

### 9.4.2.1 Reuniones de estimación de tareas\*

Las reuniones de estimación de tareas le permiten al equipo Scrum estimar el trabajo necesario para completar una tarea o conjunto de tareas, así como estimar el trabajo laboral y demás recursos necesarios para llevar a cabo las tareas dentro de un determinado sprint. En tales reuniones, los integrantes del equipo Scrum utilizan la lista de tareas para calcular la duración y trabajo para completar en el sprint las historias de usuario.

Una de los beneficios clave de esta técnica es que permite al equipo contar con una perspectiva compartida de las historias de usuario y los requerimientos para que puedan calcular de forma fiable el esfuerzo necesario. La información desarrollada en las reuniones de estimación de tareas se incluye en la lista de tareas del esfuerzo estimado y se utiliza para determinar la velocidad del sprint.

En este taller, el equipo Scrum puede utilizar diversas técnicas tales como la fragmentación, el juicio de los expertos, cálculos análogos y cálculos paramétricos.

A las reuniones de estimación de tareas en ocasiones se les denomina “reuniones de planificación del sprint”. Tales reuniones pueden también combinarse con las reuniones de planificación de tareas, tal como se describe en la sección 9.3.2.1.

### 9.4.2.2 Criterios de estimación\*

El objetivo principal del uso de los criterios de estimación es mantener los tamaños relativos de estimación y minimizar la necesidad de volver a realizar el cálculo. Los criterios de estimación se pueden expresar de muchas formas. Dos ejemplos comunes son: puntos de historia y tiempo ideal. Por ejemplo, un momento ideal normalmente describe la cantidad de horas que un integrante del equipo Scrum trabaja exclusivamente en el desarrollo de los entregables del proyecto sin incluir el tiempo dedicado a otras actividades o el trabajo que realiza fuera del proyecto. Los criterios de estimación facilitan al equipo Scrum el cálculo de las actividades y le permite evaluar y atender las ineficiencias cuando sea necesario.

### 9.4.2.3 Póker de planificación

Descrito en la sección 9.2.2.2.

#### **9.4.2.4 Puño de cinco**

Descrito en la sección 9.2.2.3.

#### **9.4.2.5 Otras técnicas de estimación de tareas**

Descrito en la sección 9.2.2.5.

### **9.4.3 Salidas**

#### **9.4.3.1 Lista de tareas del esfuerzo estimado\***

La lista de tareas del esfuerzo estimado es una lista de tareas relacionadas a las historias de usuario asignadas que se incluyen en un sprint. Por lo regular, la precisión de los cálculos varía con las capacidades del equipo. El esfuerzo estimado se expresa en términos de criterios de estimación aceptados por el equipo. La lista de tareas del esfuerzo estimado la utiliza el equipo Scrum durante las reuniones de planificación del sprint para crear la lista de pendientes del sprint y la gráfica de trabajo pendiente. Se utiliza también para determinar cuándo necesitará el equipo reducir su compromiso o cuándo puede asumir una historia de usuario adicional durante la planificación del sprint.

#### **9.4.3.2 Lista de tareas actualizada**

La lista de tareas, desarrollada como parte del proceso de creación de tareas, incluye las estimaciones iniciales de la historia de usuario que deberán ser revisadas con base en actividades de estimación más detalladas realizadas en el proceso de estimación de tareas. Pudiera haber también re-estimaciones como resultado de una revisión de sprints anteriores o un cambio en el entendimiento colectivo del equipo Scrum respecto a las historias de usuario y los requerimientos.

## 9.5 Creación de la lista de pendientes del sprint

La figura 9-12 muestra todas las entradas, las herramientas y las salidas para el proceso creación de la lista de pendientes del sprint.



Figura 9-12: Creación de la lista de pendientes del Sprint—Entradas, herramientas, y salidas

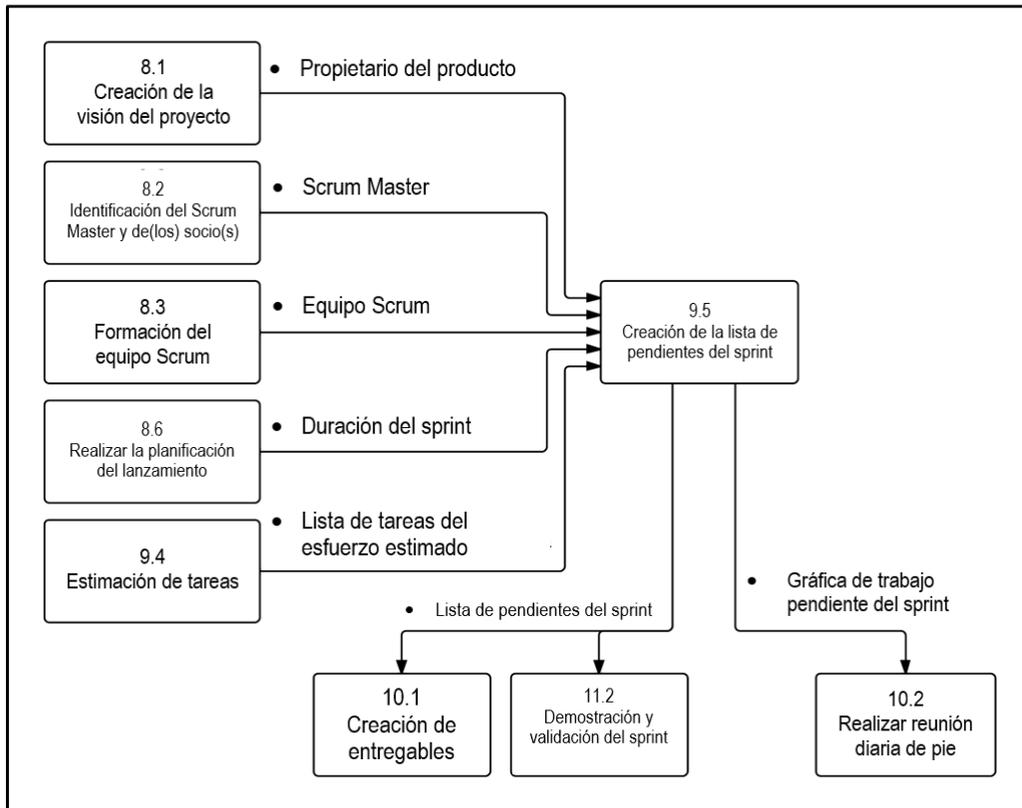


Figura 9-13: Creación de la lista de pendientes del sprint—Diagrama de flujo de datos

Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.

## 9.5.1 Entradas

### 9.5.1.1 Equipo principal de Scrum\*

Descrito en la sección 8.4.1.1.

### 9.5.1.2 Lista de tareas del esfuerzo estimado\*

Descrito en la sección 9.4.3.1.

### 9.5.1.3 Duración del sprint\*

Descrito en la sección 8.6.3.2.

### 9.5.1.4 Velocidad sprint anterior

La velocidad del sprint es la velocidad en la que el equipo puede completar el trabajo en un sprint. Por lo general se expresa en las mismas unidades que se utilizan para la estimación, normalmente en puntos de historia o tiempo ideal. Se lleva un registro de la velocidad del sprint del equipo para cada sprint y se utiliza como referencia para futuros sprints. La velocidad del sprint anterior se convierte en el factor más importante para determinar la cantidad del trabajo a la que se avocará el equipo en un subsecuente sprint. Cualquier cambio en la situación o en las condiciones a partir del último sprint, se toma en cuenta para garantizar un cálculo preciso de la velocidad del sprint para el siguiente.

### 9.5.1.5 Dependencias

Descritas en la sección 9.3.3.3.

### 9.5.1.6 Calendario del equipo

El calendario del equipo contiene información sobre la disponibilidad de los miembros, incluyendo información relacionada a las vacaciones de los empleados, licencias, ausencias temporales y días festivos.

Uno de los principales objetivos del uso del calendario del equipo es dar seguimiento al trabajo que realiza cada integrante del equipo durante el proyecto. Ayuda al equipo no solo a planificar y llevar a cabo eficientemente los sprints, sino que ayuda a alinearlos con las fechas de lanzamiento.

## 9.5.2 Herramientas

### 9.5.2.1 Reuniones de planificación del sprint\*

Durante las reuniones de planificación del sprint, las historias de usuario que son aprobadas, estimadas y asignadas durante el proceso de aprobación, estimación y asignación, se someten a discusión del equipo Scrum. Cada miembro del equipo Scrum utiliza también la lista de tareas del esfuerzo estimado para seleccionar las tareas en las que planean trabajar en el sprint con base en sus habilidades y experiencias. El equipo Scrum elabora también la lista de pendientes del sprint y la gráfica de pendientes del sprint con el uso de historias de usuario y la lista de tareas del esfuerzo estimado durante las reuniones de planificación del sprint.

### 9.5.2.2 Herramientas de seguimiento del sprint

Es importante rastrear el proceso de un sprint y saber en qué lugar se encuentra el equipo Scrum en términos de conclusión de tareas en la lista de pendientes del sprint. Se puede utilizar una variedad de herramientas para dar seguimiento al trabajo de un sprint, pero una de las más comunes es el tablero de Scrum, conocido también como tablero de tareas o gráfica de proceso. El tablero de Scrum se divide en secciones: Por hacer (*To Do*), que también se le llama: “trabajo no iniciado”. Las otras secciones son: trabajo en progreso (*In Progress*) y trabajo concluido. Se colocan notas adhesivas en la categoría correspondiente para representar cada tarea o historia de usuario a fin de reflejar el estado actual de las actividades. Se reacomodan en la siguiente categoría conforme avanza el trabajo.

### 9.5.2.3 Parámetros de seguimiento del sprint

Los parámetros que se utilizan en los proyectos Scrum incluyen: velocidad, valor empresarial entregado y cantidad de historias.

**Velocidad:** Representa el número de historias de usuario o de funcionalidades entregadas en un solo sprint.

**Valor empresarial entregado:** Mide el valor de las historias de usuario entregadas desde la perspectiva empresarial.

**Número de historias:** Describe cuántas historias de usuario se entregan como parte de un solo sprint. Se puede expresar en términos de un simple conteo o de un conteo ponderado.

## 9.5.3 Salidas

### 9.5.3.1 Lista de pendientes del sprint\*

A la lista de las tareas que habrá ejecutar el equipo Scrum en el siguiente sprint se le denomina: lista de pendientes del sprint.

Es una práctica común que la lista de pendientes del sprint se represente en un tablero Scrum o en un tablero de tareas, el cual brinda una percepción visual constante sobre el estado de las historias de usuario en la lista de pendientes. En la lista de pendientes del sprint se incluye también cualquier riesgo asociado con las varias tareas. Cualquier actividad de mitigación para atender los riesgos identificados también se incluiría como tarea en la lista de pendientes del sprint.

Una vez que se finaliza y se asigna la lista de pendientes del sprint, no se deben agregar nuevas historias de usuario; sin embargo, tal vez sea necesario agregar aquellas tareas de las historias de usuario asignadas que se pasaron por alto o que fueron ignoradas. Si surgen nuevos requerimientos durante un sprint, estos se agregarían a la lista priorizada de pendientes del producto en un futuro sprint.

### 9.5.3.2 Gráfica de trabajo pendiente del sprint\*

La gráfica de trabajo pendiente del sprint es un diagrama que muestra la cantidad de trabajo pendiente en el actual sprint. La gráfica inicial de trabajo pendiente se acompaña de un trabajo pendiente planificado (*planned burndown*). La gráfica de trabajo pendiente debe actualizarse al final de cada día conforme se concluye el trabajo. Dicha gráfica muestra el progreso que ha realizado el equipo Scrum y permite también la detección de estimaciones que pudieron haberse hecho incorrectamente. Si la gráfica de trabajo pendiente muestra que el equipo Scrum no va por el rumbo correcto en la conclusión a tiempo del sprint, el Scrum Master debe identificar cualquier obstáculo o impedimento de una conclusión satisfactoria e intentar eliminarlos.

Una tabla relacionada es la gráfica del trabajo terminado del sprint (*Sprint Burnup Chart*). A diferencia de la gráfica de trabajo pendiente —la cual muestra la cantidad de trabajo pendiente—, la gráfica de trabajo terminado del sprint muestra el trabajo concluido como parte del sprint.

## 9.6 Diagrama de flujo de datos de fase

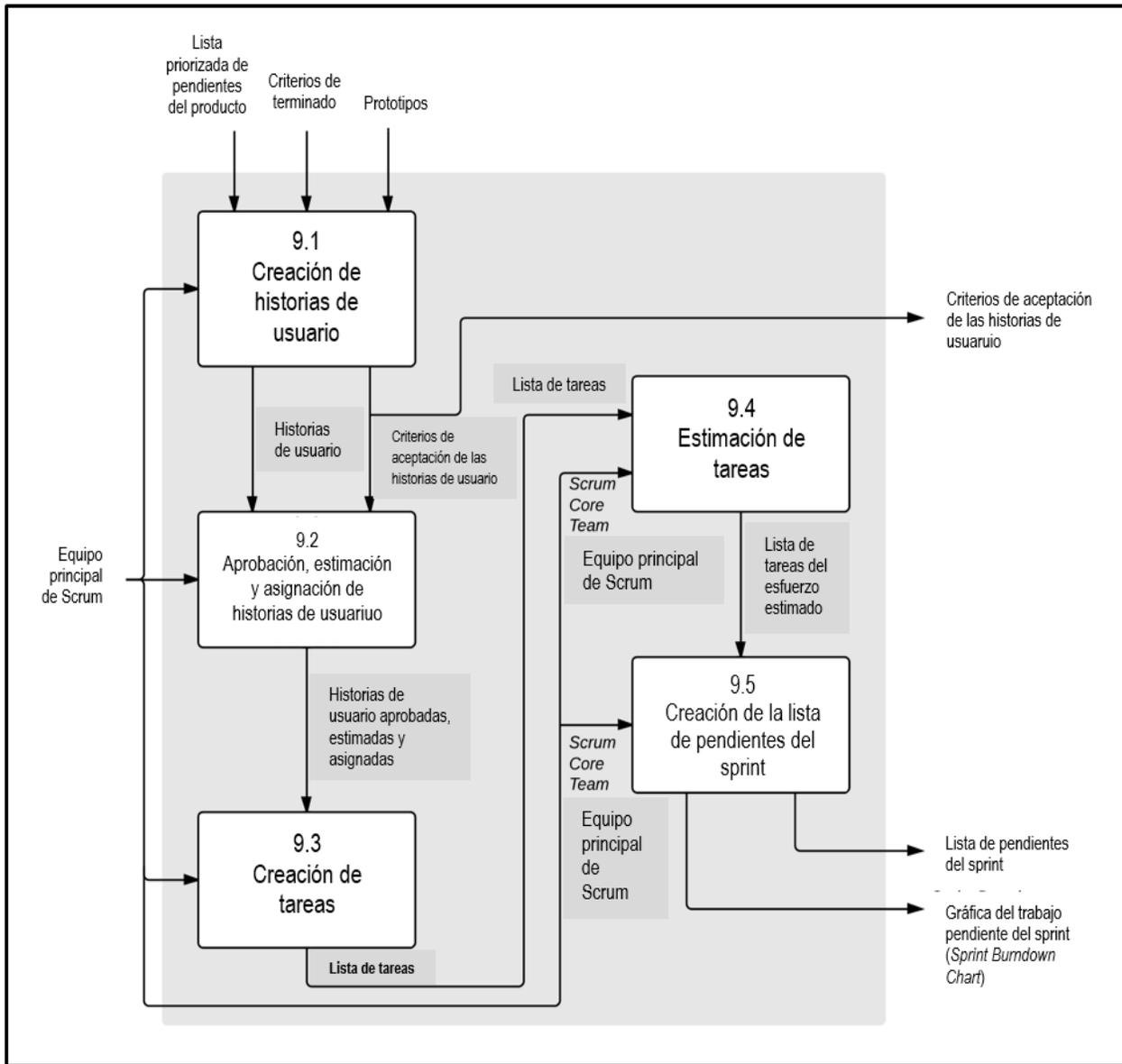


Figura 9-14: Fase de planificación y estimación—Diagrama de flujo de datos

## 10. IMPLEMENTACIÓN

La fase de implementación está relacionada a la ejecución de las tareas y actividades para crear el producto de un proyecto. Estas actividades incluyen la creación de varios entregables, la realización de reuniones diarias de pie (*Daily Standup Meetings*) y el mantenimiento (revisiones, ajustes y actualización periódica) de la lista de pendientes del producto en intervalos frecuentes.

La *implementación*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se les entregará a los socios;
- Proyectos de cualquier tamaño o complejidad.

El término “producto” en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Para facilitar la mejor aplicación del marco de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como “obligatorias” u “opcionales”. Las entradas, herramientas y salidas indicadas con asteriscos (\*) son obligatorias, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el equipo Scrum y aquellas personas que empiezan a aprender sobre el marco y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los propietarios del producto, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™*, no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

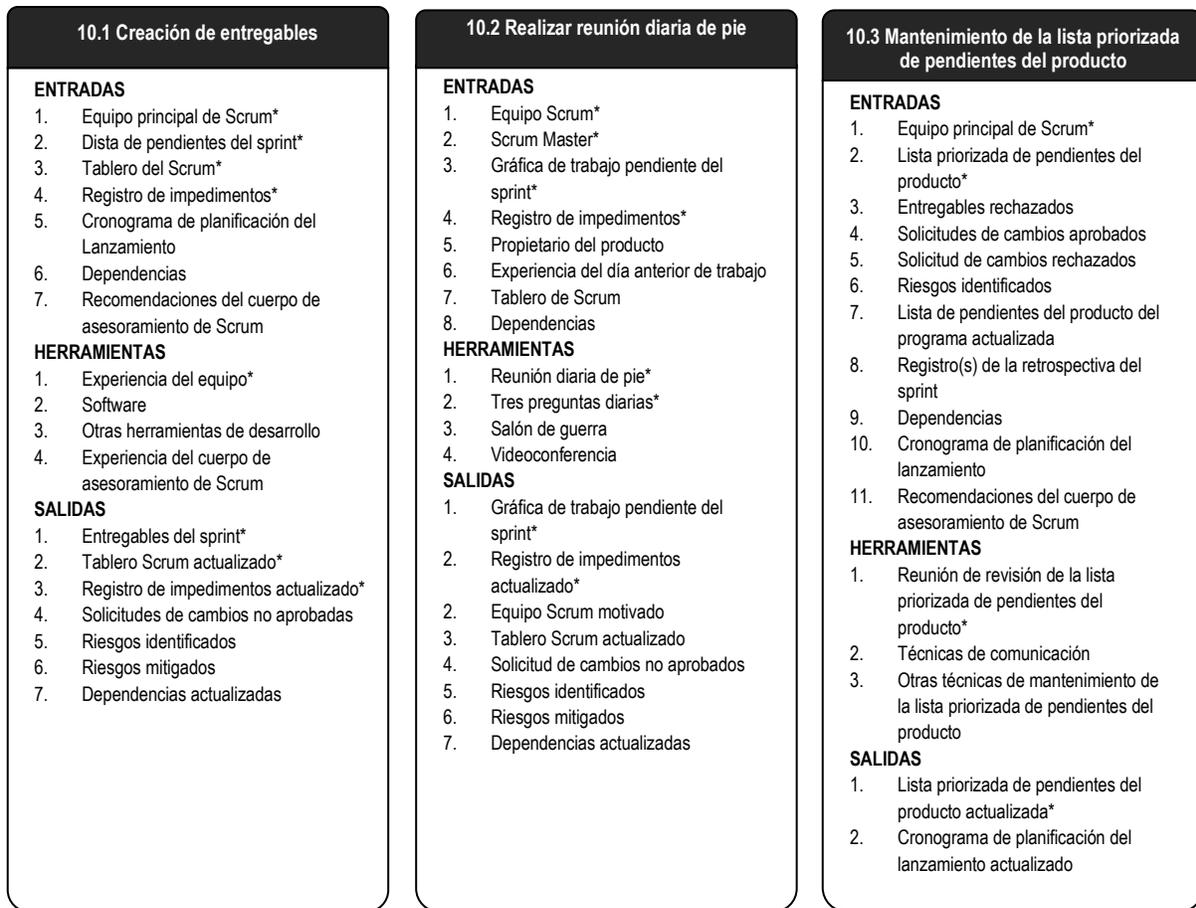
Este capítulo está escrito desde la perspectiva de un equipo Scrum que trabaja en un sprint para producir entregables que se pueden enviar como parte de un proyecto más amplio. Sin embargo, la información que se describe aplica de igual forma a proyectos enteros, programas y carteras. Hay información adicional relacionada con el uso de Scrum para proyectos, programas y portafolios que está disponible en los capítulos 2 al 7, y que cubren los principios y aspectos de Scrum.

La figura 10-1 brinda una visión general de los procesos de la fase implementación, los cuales son los siguientes:

**10.1 Creación de entregables**—En este proceso, el equipo Scrum trabaja en las tareas de la lista de pendientes del sprint para crear los entregables del sprint. Generalmente se utiliza un tablero de Scrum para dar seguimiento al trabajo y a las actividades que se llevan a cabo. Los problemas que enfrenta el equipo Scrum pueden actualizarse en un registro de impedimentos (del inglés: *Impediment Log*).

**10.2 Realizar reunión diaria de pie**—En este proceso se lleva a cabo diariamente una reunión altamente focalizada con un bloque de tiempo asignado y denominada: reunión diaria de pie (del inglés: *Daily Standup Meeting*). Es un foro para que el equipo Scrum se ponga al día sobre sus progresos y sobre cualquier impedimento que pudieran estar enfrentando.

**10.3 Mantenimiento de la lista priorizada de pendientes del producto**—En este proceso, constantemente se actualiza y se da mantenimiento a la lista priorizada de pendientes del producto (*Prioritized Product Backlog*). Se puede celebrar una reunión de revisión de dicha lista, en la cual los cambios y actualizaciones a la lista de pendientes se analizan y se incorporan a la lista priorizada de pendientes del producto según corresponda.



**Figura 10-1: Resumen de la implementación**

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

La figura 10-2 muestra las entradas, herramientas y salidas obligatorias para los procesos en la fase de implementación.

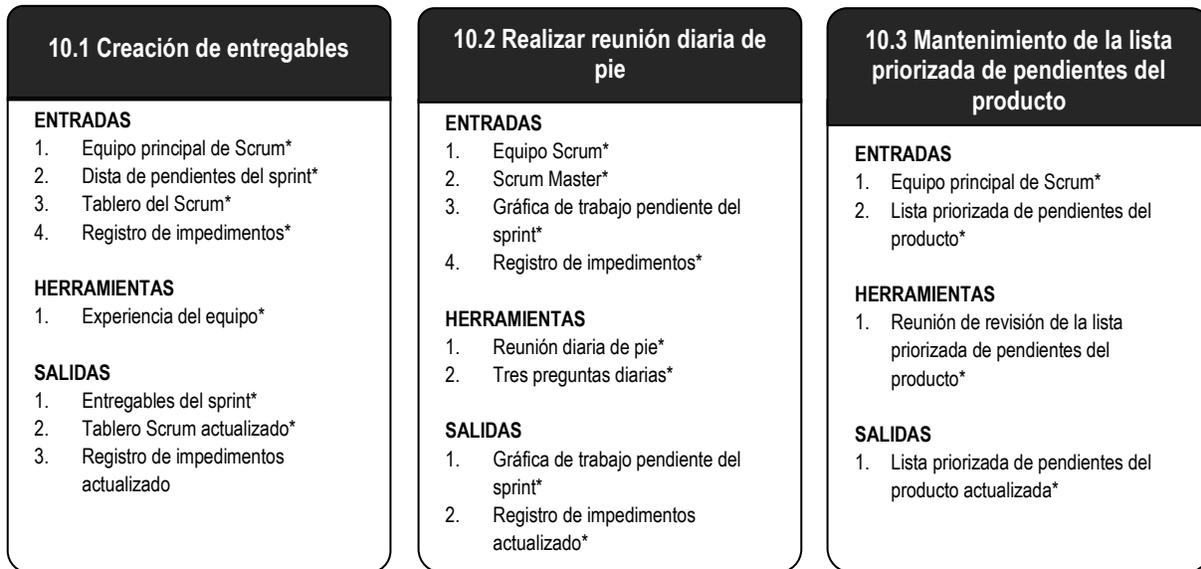


Figura 10-2: Resumen de la implementación (Esenciales)

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

## 10.1 Creación de entregables

La figura 10-3 muestra todas las entradas, las herramientas y salidas para el proceso de creación de entregables.

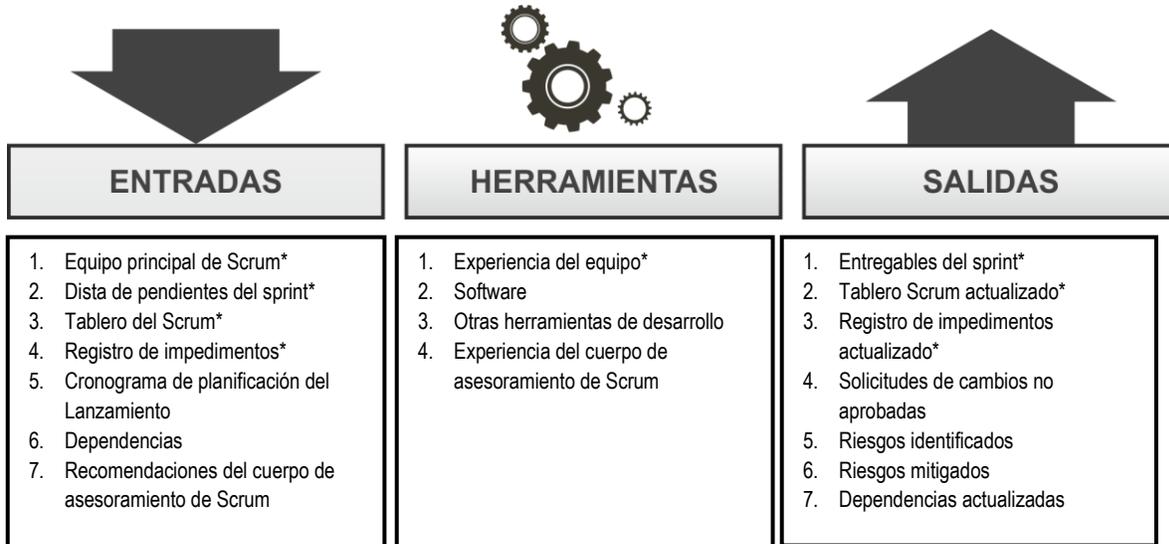


Figura 10-3: Creación de entregables—Entradas, herramientas y salidas

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

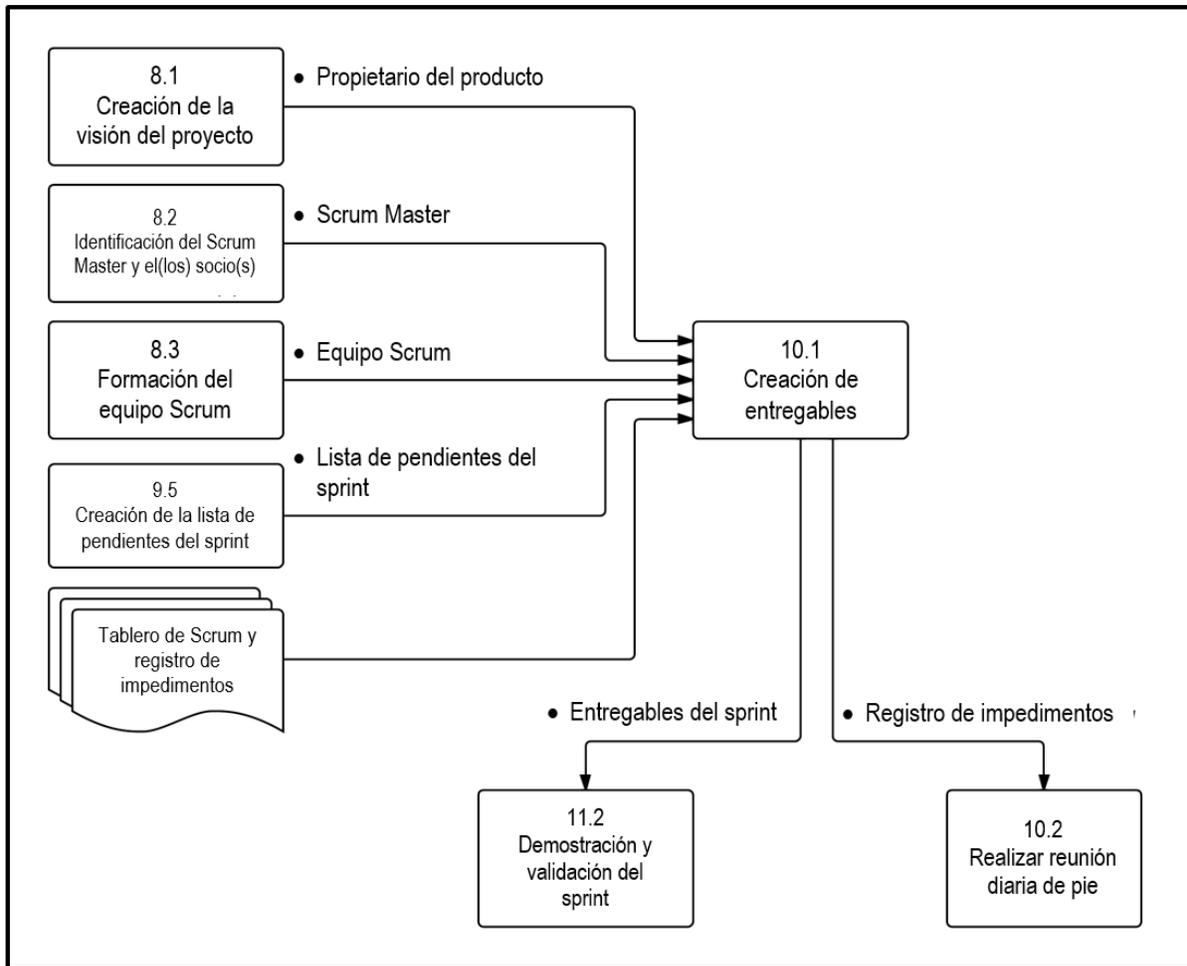


Figura 10-4: Creación de entregables—Diagrama de flujo de datos



El equipo Scrum debe cambiar y agregarle al tablero Scrum según sea necesario, de tal forma que brinde información visual y control sobre el trabajo en acción según lo acordado y asignado al equipo.

#### **10.1.1.4 Registro de impedimentos\* (Registro de impedimentos).**

Un impedimento es cualquier obstáculo o barrera que reduce la productividad del equipo Scrum. Los impedimentos deben identificarse, resolverse y eliminarse si el equipo sigue trabajando eficazmente. Los impedimentos pueden ser internos en el equipo, tales como un flujo de trabajo ineficiente o la falta de comunicación, o bien, pudieran ser externos. Algunos ejemplos de impedimentos externos pudieran ser problemas relacionados a licencias de software o requisitos de documentación innecesaria. El marco de Scrum, con su transparencia inherente, facilita la rápida y fácil identificación de impedimentos. Si no se identifican o se atienden los impedimentos, pudiera ser muy costoso. El Scrum Master debe registrar formalmente los impedimentos en un registro de impedimentos y se pueden analizar durante las reuniones diarias de pie y en las reuniones de revisión del sprint según sea necesario.

#### **10.1.1.5 Cronograma de planificación del lanzamiento**

Descrito en la sección 8.6.3.1.

#### **10.1.1.6 Dependencias**

Descrito en la sección 9.3.3.3.

#### **10.1.1.7 Recomendaciones del cuerpo de asesoramiento de Scrum**

Descrito en la sección 8.1.1.12.

En el proceso de creación de entregables, las recomendaciones del cuerpo de asesoramiento de Scrum pudieran incluir las mejores prácticas para crear eficazmente los entregables, incluyendo métodos preferenciales para realizar las revisiones, la evaluación, documentación, etc.

## 10.1.2 Herramientas

### 10.1.2.1 Experiencia del equipo\*

Hace referencia a la experiencia colectiva de los miembros del equipo Scrum para entender las historias de usuario y las tareas en la lista de pendientes del sprint a fin de crear los entregables finales. La experiencia del equipo se utiliza para evaluar las entradas necesarias a fin de ejecutar el trabajo planificado del proyecto. El juicio y la experiencia se aplican en todos los aspectos técnicos y administrativos del proyecto durante el proceso de creación de entregables. Los miembros del equipo Scrum cuentan con la autoridad y la responsabilidad para determinar las mejores formas de convertir los elementos de la lista priorizada de pendientes del producto en productos finales, sin solicitar la participación de ningún socio fuera del equipo. De ser necesario, el cuerpo de asesoramiento de Scrum cuenta con experiencia disponible.

### 10.1.2.2 Software

Las herramientas automatizadas de software se pueden utilizar para programar, recopilar información y para la distribución. Las herramientas virtuales de colaboración también son esenciales en proyectos donde el equipo Scrum no esté co-ubicado. Hay una variedad de herramientas automatizadas de software que permiten dar seguimiento al proceso, a recopilar información y para la distribución, y que contribuyen a acelerar el proceso.

10

### 10.1.2.3 Otras herramientas de desarrollo

Con base en los requerimientos específicos del proyecto y en las especificaciones de la industria, se pueden implementar otras herramientas de desarrollo según sea necesario.

#### 1. Refactorización

La refactorización es una herramienta específica para proyectos de software. El objetivo de esta técnica es mejorar el mantenimiento del código existente y hacerlo más sencillo, más conciso y más flexible. Refactorizar significa mejorar el diseño del código actual sin cambiar el comportamiento del mismo. Implica lo siguiente:

- La eliminación de código repetitivo y redundante
- Separar los métodos y las funciones en rutinas más pequeñas
- Definir claramente las variables y los nombres de los métodos
- Simplificar el diseño del código
- Hacer que el código sea más fácil de entender y modificar

La refactorización constante optimiza el diseño del código poco a poco durante cierto periodo. En última instancia, la refactorización da como resultado en un código más limpio y más fácil de mantener, preservando a la vez todas las funcionalidades.

## 2. Patrones de diseño

Los patrones de diseño proporcionan una manera formal de registrar una resolución a un problema de diseño en un campo específico de especialización. Dichos patrones registran tanto el proceso que se utiliza como la resolución, misma que puede reutilizarse después para mejorar la toma de decisiones y la productividad.

### 10.1.2.4 Experiencia del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.4.2.7.

En los procesos de creación de entregables y de aprobación, estimación y asignación de historias de usuario, el cuerpo de asesoramiento de Scrum pudiera relacionarse a las normas y regulaciones documentadas, directrices de desarrollo, o bien, a los estándares y mejores prácticas (por ejemplo, orientación sobre cómo realizar revisiones o pruebas). Pudiera haber también un equipo de expertos en la materia que brinden orientación al equipo Scrum sobre la creación de entregables. Dicho equipo puede incluir a arquitectos líderes, desarrolladores de alto nivel, expertos en seguridad o demás personas con experiencia.

## 10.1.3 Salidas

### 10.1.3.1 Entregables del sprint\*

Al final de cada sprint se completa un incremento de producto o entregable. El entregable debe incluir todas las características y funcionalidades definidas en las historias de usuario que se incluyen en el sprint y deben haber sido evaluadas satisfactoriamente.

### 10.1.3.2 Tablero de Scrum actualizado

El tablero Scrum se actualiza con regularidad a medida que el equipo completa las tareas. Sin embargo, al final del sprint, el tablero Scrum se reinicia o se borra y se crea un nuevo para el siguiente sprint.

### 10.1.3.3 Registro de impedimentos actualizado

Descrito en la sección 10.1.1.4.

#### **10.1.3.4 Solicitudes de cambio no aprobadas**

Descrito en la sección 8.4.1.6.

#### **10.1.3.5 Riesgos identificados**

Descritos en la sección 8.4.3.4.

#### **10.1.3.6 Riesgos mitigados**

A medida que el equipo Scrum ejecuta el trabajo de creación de entregables según las historias de usuario en la lista de pendientes del producto, lleva a cabo acciones de mitigación que fueron definidas para atender cualquier riesgo identificado anteriormente. Durante todo el proceso de creación de entregables, el equipo documenta cualquier nuevo riesgo identificado y se toman las acciones para mitigarlo. El registro de los riesgos del proyecto (del inglés: *record of project risks*) es un documento vivo que actualiza constantemente el equipo durante todo el proyecto a fin de reflejar el estado actual de todos los riesgos.

En la sección 7.4.3 se describe información adicional sobre la mitigación de riesgos.

#### **10.1.3.7 Dependencias actualizadas**

Descritas en la sección 9.3.3.3.

## 10.2 Realizar reunión diaria de pie

La figura 10-6 muestra todas las entradas, las herramientas y las salidas del proceso para realizar la reunión diaria de pie.

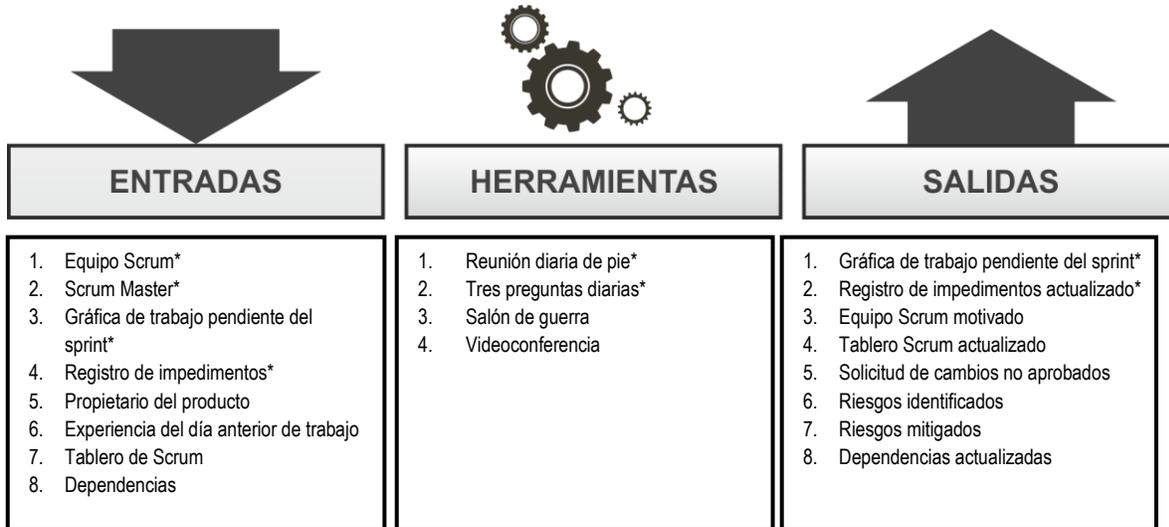


Figura 10-6: Realizar reunión diaria de pie—Entradas, herramientas y salidas

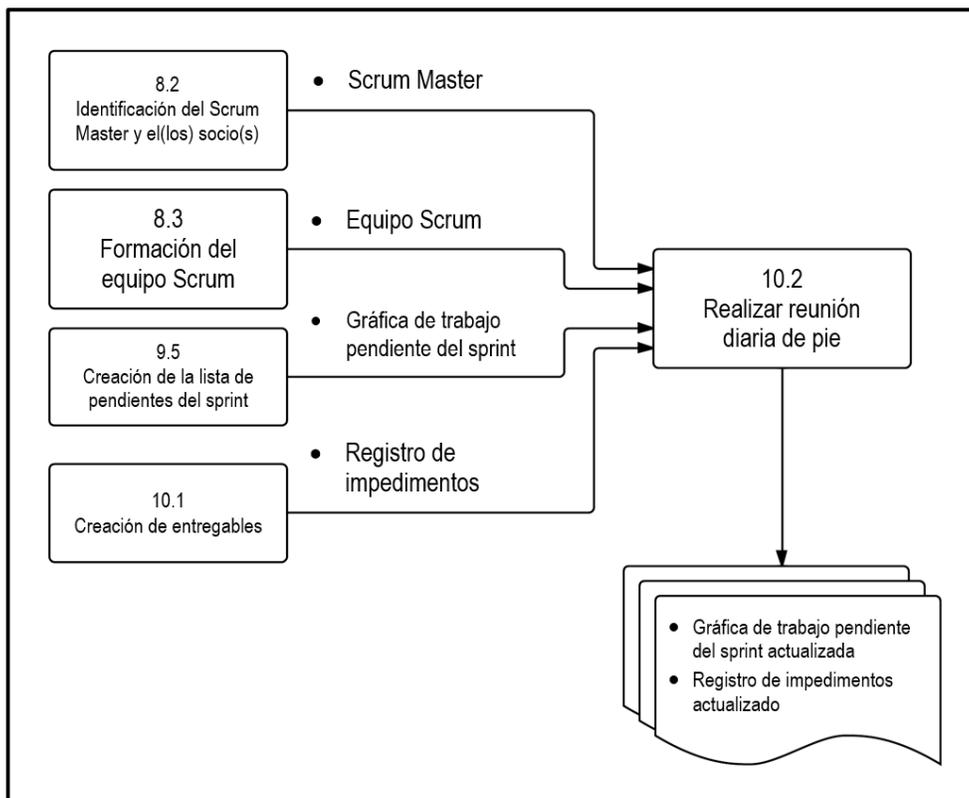


Figura 10-7: Realizar reunión diaria de pie—Diagrama de flujo de datos

Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.

## 10.2.1 Entradas

### 10.2.1.1 Equipo Scrum\*

Descrito en la sección 8.3.3.1.

### 10.2.1.2 Scrum Master\*

Descrito en la sección 8.2.3.1.

### 10.2.1.3 Gráfica de trabajo pendiente del sprint\*

Descrita en la sección 9.5.3.2.

### 10.2.1.4 Registro de impedimentos\*

Descrito en la sección 10.1.1.4.

### 10.2.1.5 Propietario del producto

Descrito en la sección 8.1.3.1.

### 10.2.1.6 Experiencia del día anterior de trabajo

Los miembros del equipo Scrum mantienen al tanto a sus compañeros durante las reuniones diarias de pie (*Daily Standup Meetings*). Esta sesión se denomina “de pie” (*standup*) ya que los miembros permanecen de pie durante toda la reunión. Los integrantes del equipo discuten los logros y la experiencia del día anterior. Esta experiencia es una contribución importante a la reunión diaria de pie.

### 10.2.1.7 Tablero de Scrum

Descrito en la sección 10.1.1.3.

### 10.2.1.8 Dependencias

Descritas en la sección 9.3.3.3.

## 10.2.2 Herramientas

### 10.2.2.1 Reunión diaria de pie\*

La reunión diaria de pie es una breve reunión diaria con un bloque de tiempo de 15 minutos. Los miembros del equipo se reúnen para dar un reporte sobre su progreso en el sprint y planificar las actividades del día. La duración de la reunión es muy corta y se busca que todos los integrantes del equipo Scrum estén presentes. Sin embargo, la reunión no se cancela o se retrasa si uno o más miembros no pueden asistir.

En la reunión cada miembro del equipo Scrum da respuesta a las tres preguntas diarias que se mencionan en la sección 10.2.2.2. Se recomiendan los debates entre el Scrum Master y el equipo o entre los miembros del equipo Scrum, aunque tales discusiones suceden después de la reunión a fin de garantizar que la reunión diaria de pie sea breve.

### 10.2.2.2 Tres preguntas diarias\*

Las tres preguntas diarias se utilizan en las reuniones diarias de pie, organizadas por el Scrum Master, donde cada miembro del equipo Scrum brinda información en forma de respuesta a tres preguntas específicas:

1. ¿Qué terminé ayer?
2. ¿Qué terminaré hoy?
3. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

Al centrarse en estas tres preguntas, todo el equipo puede tener una comprensión clara de la situación laboral. En ocasiones se pueden discutir otros elementos, pero se mantiene al mínimo en razón del tiempo asignado que tiene la reunión.

Es muy recomendable que los miembros del equipo den respuesta a las dos primeras preguntas en forma cuantificable, si es posible, en vez de largas respuestas cualitativas. Los miembros del equipo pueden organizar reuniones adicionales después de la reunión diaria de pie a fin de abordar temas que requieran de mayor discusión.

### 10.2.2.3 Salón de guerra

En Scrum, es preferible que el equipo este co-ubicado, donde todos los miembros trabajen en la misma ubicación. “Salón de guerra”, o *War Room* en inglés, es el término que comúnmente se utiliza para describir a dicho lugar. Normalmente está diseñado de tal manera que los miembros del equipo pueden moverse libremente, trabajar y comunicarse fácilmente, ya que se encuentran ubicados en proximidades inmediatas. Típicamente se distribuyen fichas, notas adhesivas o demás herramientas sencillas o sofisticadas a fin de facilitar el flujo de trabajo, la colaboración y la solución de problemas.

El salón en ocasiones es ruidoso debido a las conversaciones del equipo, aunque tales conversaciones contribuyen al progreso del equipo. Un buen salón de guerra es un cubículo libre que permite a todo el equipo sentarse junto, garantizando una comunicación cara a cara, lo cual conduce a la formación del equipo y a la apertura. El salón de guerra, o *War Room*, también es ideal para llevar a cabo las reuniones diarias de pie.

El (los) socio(s) de otros equipos de Scrum pudieran también llegar al salón de guerra y discutir temas relevantes.

#### **10.2.2.4 Videoconferencia**

En situaciones de la vida real, tal vez no sea posible que todo el equipo Scrum esté co-ubicado. En tales casos, resulta imperativo utilizar herramientas de videoconferencia que permitan la comunicación cara a cara.

### **10.2.3 Salidas**

#### **10.2.3.1 Gráfica de trabajo pendiente del sprint actualizada\***

Descrita en la sección 9.5.3.2.

#### **10.2.3.2 Registro de impedimentos actualizado\***

Descrito en la sección 10.1.1.4.

#### **10.2.3.3 Equipo Scrum motivado**

Las reuniones diarias de pie propagan la idea de que cada miembro del equipo es importante y es un importante colaborador, lo cual mejora la moral individual y del equipo. Ello, aunado al concepto de la auto-organización del equipo, mejora la motivación general, conduciendo a un mejor desempeño del equipo y mejorando la calidad de los entregables que se producen.

El equipo Scrum se describe en la sección 8.3.3.1.

#### **10.2.3.4 Tablero de Scrum actualizado**

Descrito en la sección 10.1.1.3.

### **10.2.3.5 Solicitudes de cambio no aprobadas**

Descritas en la sección 8.4.1.6.

### **10.2.3.6 Riesgos identificados**

Descritos en la sección 8.4.3.4.

### **10.2.3.7 Riesgos mitigados**

Descritos en la sección 10.1.3.6.

### **10.2.3.8 Dependencias actualizadas**

Descritas en la sección 9.3.3.3.

## 10.3 Mantenimiento de la lista priorizada de pendientes del producto

La figura 10-8 muestra todas las entradas, las herramientas y las salidas para el proceso de mantenimiento de la lista priorizada de pendientes del producto.

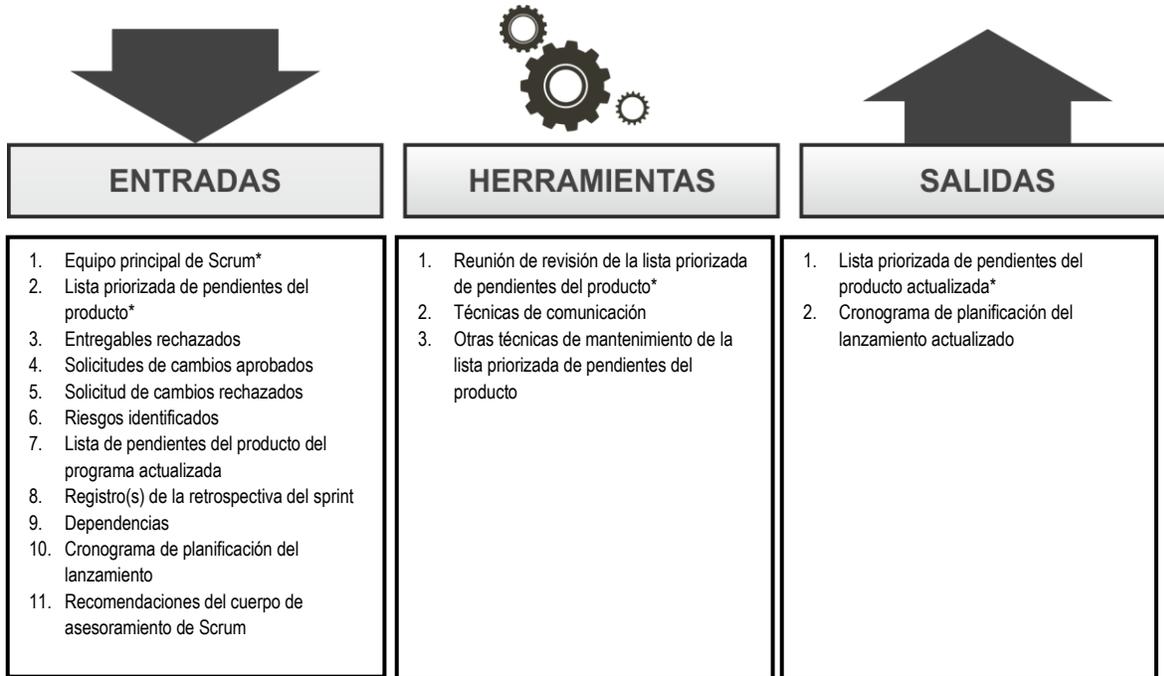


Figura 10-8: Mantenimiento de la lista priorizada de pendientes del producto—Entradas, herramientas y salidas

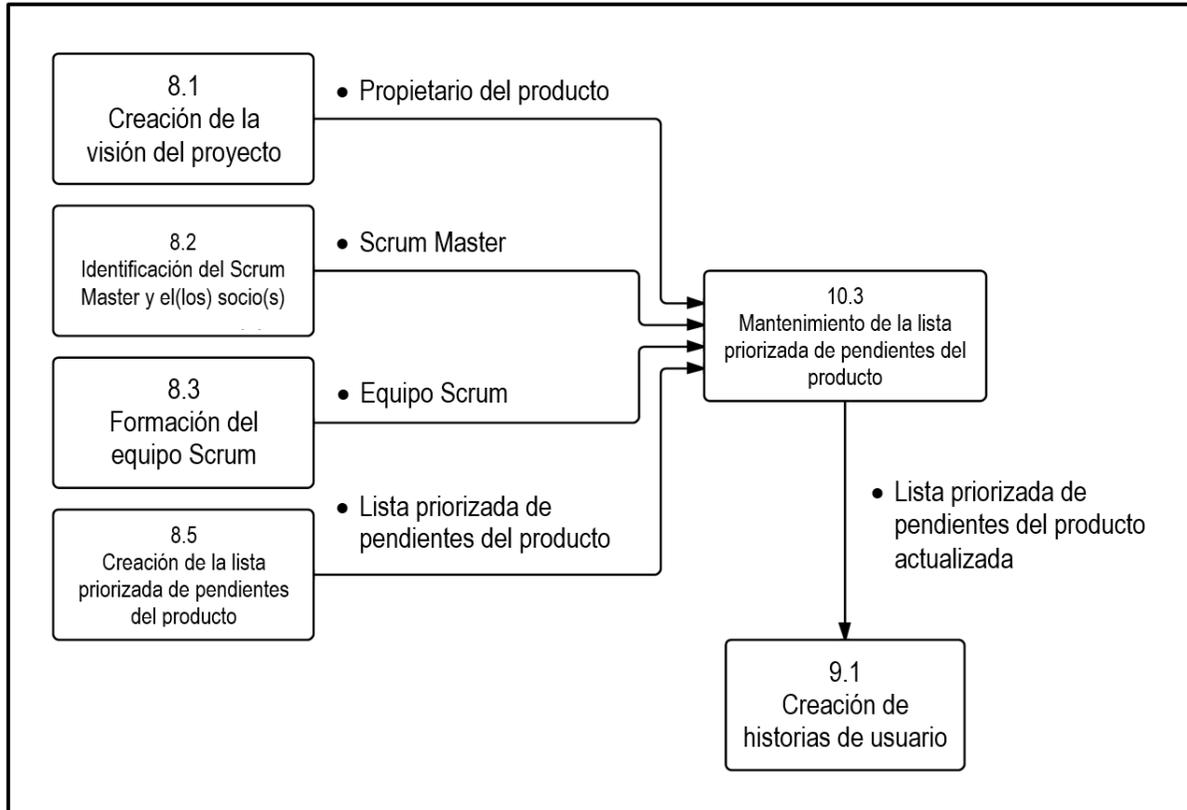


Figura 10-9: Mantenimiento de la lista priorizada de pendientes del producto—Diagrama de flujo de datos

### 10.3.1 Entradas

#### 10.3.1.1 Equipo principal de Scrum\*

Descrito en las secciones 8.1.3.1, 8.2.3.1 y 8.3.3.1.

#### 10.3.1.2 Lista priorizada de pendientes del producto\*

Descrita en la sección 8.5.3.1.

#### 10.3.1.3 Entregables rechazados

En los casos donde un entregable no cumpla con los criterios de aceptación, este se considera como entregable rechazado. Los entregables rechazados generalmente no se mantienen en una lista por separado, simplemente permanecen en la lista priorizada de pendientes del producto y no se marcan como terminados, de forma que se puedan volver a priorizar en el proceso de mantenimiento de la lista priorizada de pendientes del producto y se consideren para su desarrollo en el siguiente sprint.

#### **10.3.1.4 Solicitudes de cambio aprobadas**

Descrito en la sección 8.4.1.5.

#### **10.3.1.5 Solicitudes de cambio no aprobadas**

Descrito en la sección 8.4.1.6.

#### **10.3.1.6 Riesgos identificados**

Descritos en la sección 8.4.3.4.

#### **10.3.1.7 Lista priorizada de pendientes del producto actualizada**

Al igual que la lista de pendientes del producto del proyecto, la lista de pendientes del producto del programa puede también someterse a un mantenimiento constante a fin de incorporar cambios y nuevos requerimientos. Los cambios a la lista de pendientes del producto del programa pueden resultar a partir de los cambios tanto en las condiciones externas como internas. Las condiciones externas pueden incluir situaciones de cambio empresariales, tendencias en la tecnología o requerimientos de cumplimientos legales. Los factores internos que afectan la lista de pendientes del producto del programa pueden estar relacionados a modificaciones a las políticas o a las estrategias organizaciones, riesgos identificados o demás factores. Los cambios en los requerimientos en la lista de pendientes del producto de programa pueden tener un impacto de las listas de pendientes del producto del proyecto en proyectos subyacentes, a fin de que se puedan tomar en cuenta durante el proceso de mantenimiento de la lista priorizada de pendientes del producto.

#### **10.3.1.8 Retrospectiva del sprint**

Descrita en la sección 11.3.3.4.

#### **10.3.1.9 Dependencias**

Descritas en la sección 9.3.3.3.

#### **10.3.1.10 Cronograma de planificación del lanzamiento**

Descrito en la sección 8.6.3.1.

### 10.3.1.11 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

En el proceso de mantenimiento de la lista priorizada de pendientes del producto, las recomendaciones del cuerpo de asesoramiento de Scrum pueden incluir las mejores prácticas sobre cómo entender y recopilar sistemáticamente los requerimientos de los socios y de los equipos de Scrum; priorizarlos adecuadamente en la lista de pendientes del producto y comunicar las actualizaciones a todas las personas relevantes implicadas en un proyecto Scrum.

## 10.3.2 Herramientas

### 10.3.2.1 Reuniones de revisión de la lista priorizada de pendientes del producto\*

El propietario del producto pudiera tener varias reuniones por separado con los socios relevantes, con el Scrum Master y con el equipo Scrum a fin de asegurarse de que cuente con la información suficiente para actualizar la lista priorizada de pendientes del producto en el proceso de su mantenimiento.

La intención de las reuniones de revisión de la lista priorizada de pendientes del producto es asegurarse de que se entiendan las historias de usuario y los criterios de aceptación y de que el propietario del producto las redacte adecuadamente reflejando los requerimientos y prioridades actuales del socio (cliente); que todos en el Equipo Scrum entiendan las historias de usuario, y que las historias de usuario de mayor prioridad sean bien refinadas de tal forma que el equipo Scrum las pueda estimar adecuadamente y avocarse a ellas.

Las reuniones de revisión de la lista priorizada de pendientes del producto garantizan también que se eliminen historias de usuario irrelevantes y que se incorporen las solicitudes de cambio aprobadas o los riesgos identificados en la lista priorizada de pendientes del producto.

### 10.3.2.2 Técnicas de comunicación

Scrum promueve la comunicación precisa y eficaz primordialmente mediante la co-ubicación de equipos Scrum. Scrum favorece también las interacciones informales cara a cara en vez de la comunicación formal por escrito. Cuando un equipo Scrum debe estar distribuido geográficamente, el Scrum Master debe garantizar que existan técnicas eficaces de comunicación para que los equipos se puedan auto organizar y trabajar eficientemente.

### 10.3.2.3 Otras técnicas de mantenimiento de la lista priorizada de pendientes del producto

Otras herramientas de mantenimiento de la lista priorizada de pendientes del producto incluyen muchas de las mismas herramientas implementadas en los siguientes procesos:

- *Desarrollo de épica(s)*—Descrito en la sección 8.4.2.

- *Creación de la lista priorizada de pendientes del producto*—Descrito en la sección 8.5.2.
- *Realizar la planificación del lanzamiento*—Descrito en la sección 8.6.2.
- *Creación de historias de usuario*—Descrito en la sección 9.1.2.
- *Aprobación, estimación y asignación de historias de usuarios*—Descrito en la sección 9.2.2.
- *Creación de tareas*—Descrito en la sección 9.3.2.
- *Estimación de tareas*—Descrito en la sección 9.4.2.

### 10.3.3 Salidas

#### 10.3.3.1 Lista priorizada de pendientes del producto actualizada\*

Descrita en la sección 8.5.3.1.

La lista priorizada de pendientes del producto se puede actualizar con nuevas historias de usuario, nuevas solicitudes de cambio, nuevos riesgos identificados o historias de usuario actualizadas o bien, la re-priorización de las historias de usuario ya existentes.

#### 10.3.3.2 Cronograma de planificación del lanzamiento actualizado

Descrito en la sección 8.6.3.1.

El cronograma de planificación del lanzamiento (del inglés: *Release Planning Schedule*) puede actualizarse a fin de que refleje el impacto de las historias de usuario nuevas o actualizadas en la lista priorizada de pendientes del producto.

## 10.4 Diagrama de flujo de datos de fase

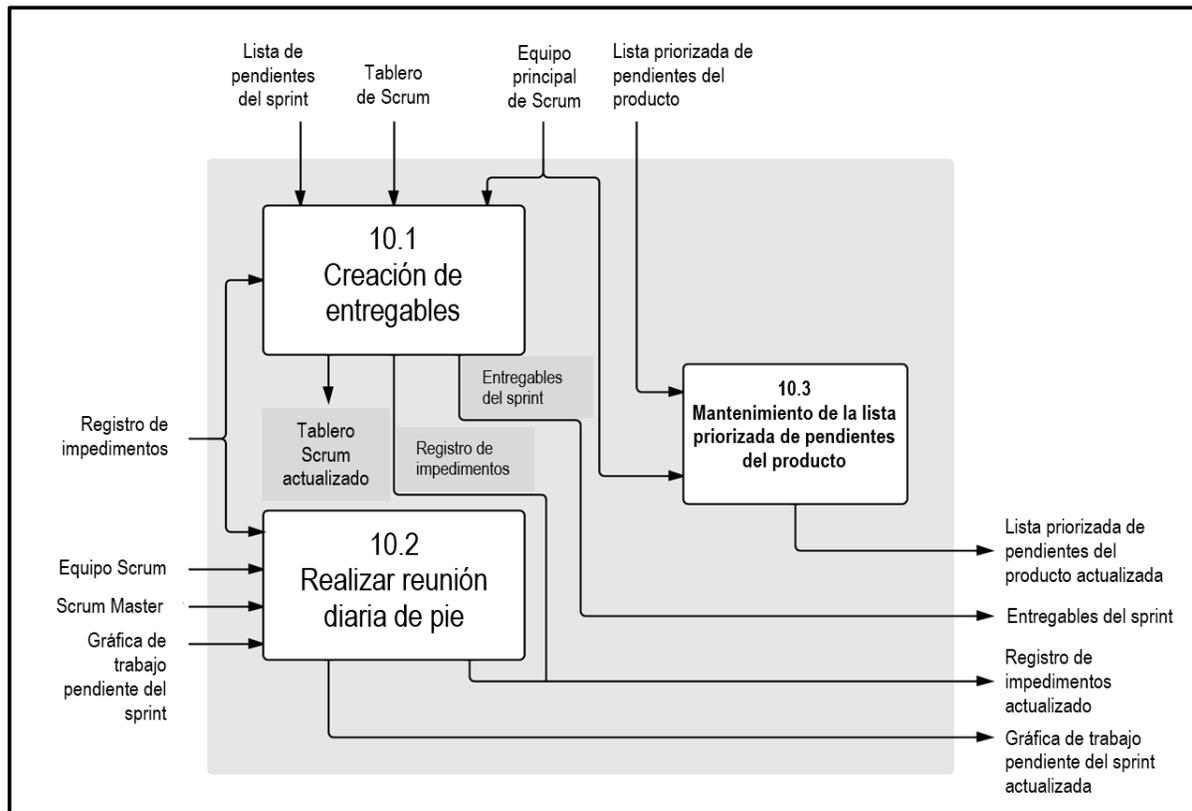


Figura 10-10: Fase de implementación—Diagrama de flujo de datos

## 11. REVISIÓN Y RETROSPECTIVA

La fase de revisión y retrospectiva se ocupa de la revisión de los entregables y del trabajo que se ha realizado y determina las formas para mejorar las prácticas y métodos implementados para realizar el trabajo del proyecto. En las grandes organizaciones, el proceso de revisión y retrospectiva puede también incluir el convocar a reuniones de Scrum de Scrums.

La *revisión y retrospectiva*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se les entregará a los socios;
- Proyectos de cualquier tamaño o complejidad.

El término “producto” en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Para facilitar la mejor aplicación del marco de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como “obligatorias” u “opcionales”. Las entradas, herramientas y salidas indicadas con asteriscos (\*) son obligatorias, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el equipo Scrum y aquellas personas que empiezan a aprender sobre el marco y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los propietarios del producto, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™*, no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está escrito desde la perspectiva de un equipo Scrum que trabaja en un sprint para producir entregables que se pueden enviar como parte de un proyecto más amplio. Sin embargo, la información que se describe aplica de igual forma a proyectos enteros, programas y carteras. Hay información adicional relacionada con el uso de Scrum para proyectos, programas y portfolios que está disponible en los capítulos 2 al 7, y que cubren los principios y aspectos de Scrum.

La figura 11-1 brinda una visión general de los procesos de la fase *Revisión y Retrospectiva*, los cuales son los siguientes:

**11.1 Convocar a un Scrum de Scrums**—En este proceso, los representantes del equipo Scrum convocan una reunión de Scrum de Scrums en intervalos predeterminados o cuando sea necesario a fin de colaborar y dar seguimiento a sus respectivos progresos, impedimentos y dependencias entre los equipos. Esto es relevante únicamente para grandes proyectos que involucran a múltiples equipos de Scrum.

**11.2 Demostración y validación del sprint**—En este proceso, el equipo Scrum demuestra los entregables del sprint al propietario del producto y a los socios relevantes durante una reunión de revisión del sprint. El propósito de esta reunión es lograr la aprobación y aceptación del propietario del producto respecto al producto o servicio.

**11.3 Retrospectiva de sprint**—En este proceso, el Scrum Master y el equipo Scrum se reúnen para discutir las lecciones aprendidas durante el sprint. Dicha información se documenta como lecciones aprendidas que pudieran implementarse en futuros sprints. Generalmente, como consecuencia de esta reunión, pudieran resultar mejoras accionables aceptadas (conocidas en inglés como: *Agreed Actionable Improvements*) o recomendaciones actualizadas del cuerpo de asesoramiento de Scrum.

11.1 Convocar a un Scrum de Scrums	11.2 Demostración y validación del Sprint	11.3 Retrospectiva del Sprint
<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Scrum Master o representantes del equipo Scrum*</li> <li>2. Jefe Scrum Master</li> <li>3. Jefe propietario del producto</li> <li>4. Orden del día</li> <li>5. Registro de impedimentos</li> <li>6. Dependencias</li> <li>7. Salidas de la Retrospectiva del Sprint</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reunión de Scrum de Scrums*</li> <li>2. Cuatro preguntas por equipo*</li> <li>3. Videoconferencia</li> <li>4. Sala de juntas</li> <li>5. Experiencia del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Mejor coordinación del equipo*</li> <li>2. Problemas resueltos</li> <li>3. Registro de impedimentos actualizado</li> <li>4. Dependencias actualizadas</li> </ol>	<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Equipo principal de Scrum*</li> <li>2. Entregables del sprint*</li> <li>3. Lista de pendientes del Sprint*</li> <li>4. Criterios de terminado*</li> <li>5. Criterio de aceptación de las historias del usuario*</li> <li>6. Socio(s)</li> <li>7. Cronograma de planificación del lanzamiento</li> <li>8. Riesgos identificados</li> <li>9. Dependencias</li> <li>10. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reuniones de revisión del Sprint*</li> <li>2. Análisis del valor ganado</li> <li>3. Experiencia del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Entregables aceptados*</li> <li>2. Entregables rechazados</li> <li>3. Riesgos actualizados</li> <li>4. Resultados del análisis del valor ganado</li> <li>5. Cronograma de planificación del lanzamiento actualizado</li> <li>6. Dependencias actualizadas</li> </ol>	<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Scrum Master*</li> <li>2. Equipo Scrum*</li> <li>3. Salidas de la demostración y validación del Sprint*</li> <li>4. Propietario del producto</li> <li>5. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reunión de retrospectiva del Sprint*</li> <li>2. ECVP</li> <li>3. Lancha rápida (<i>Speed Boat</i>)</li> <li>4. Parámetros y técnicas de medición</li> <li>5. Experiencia del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Mejoras accionables aceptadas *</li> <li>2. Elementos de acción asignados y fechas de entrega</li> <li>3. Elementos no funcionales propuestos para la lista priorizada de pendientes del producto</li> <li>4. Registro de la retrospectiva del sprint</li> <li>5. Lecciones aprendidas del equipo de Scrum</li> <li>6. Recomendaciones actualizadas del cuerpo de asesoramiento de Scrum</li> </ol>

Figura 11-1: Resumen de la revisión y retrospectiva

Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.

La figura 11-2 muestra las entradas, herramientas y salidas obligatorias para los procesos en la fase *revisión y retrospectiva*.

11.1 Convocar a un Scrum de Scrums	11.2 Demostración y validación del sprint	11.3 Retrospectiva del sprint
<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Scrum Master o representantes del equipo Scrum*</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reunión de Scrum de Scrums*</li> <li>2. Cuatro preguntas por equipo*</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Mejor coordinación del equipo*</li> </ol>	<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Equipo principal de Scrum*</li> <li>2. Entregables del sprint*</li> <li>3. Lista de pendientes del Sprint*</li> <li>4. Criterios de terminado*</li> <li>5. Criterio de aceptación de las historias del usuario*</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reuniones de revisión del Sprint*</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Entregables aceptados*</li> </ol>	<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Scrum Master*</li> <li>2. Equipo Scrum*</li> <li>3. Salidas de la demostración y validación del Sprint*</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reunión de retrospectiva del Sprint*</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Mejoras accionables aceptadas*</li> </ol>

Figura 11-2: Resumen de la revisión y retrospectiva (esenciales)

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

## 11.1 Convocar a un Scrum de Scrums

La figura 11-3 muestra todas las entradas, las herramientas y las salidas para el proceso para convocar a un Scrum de Scrums.

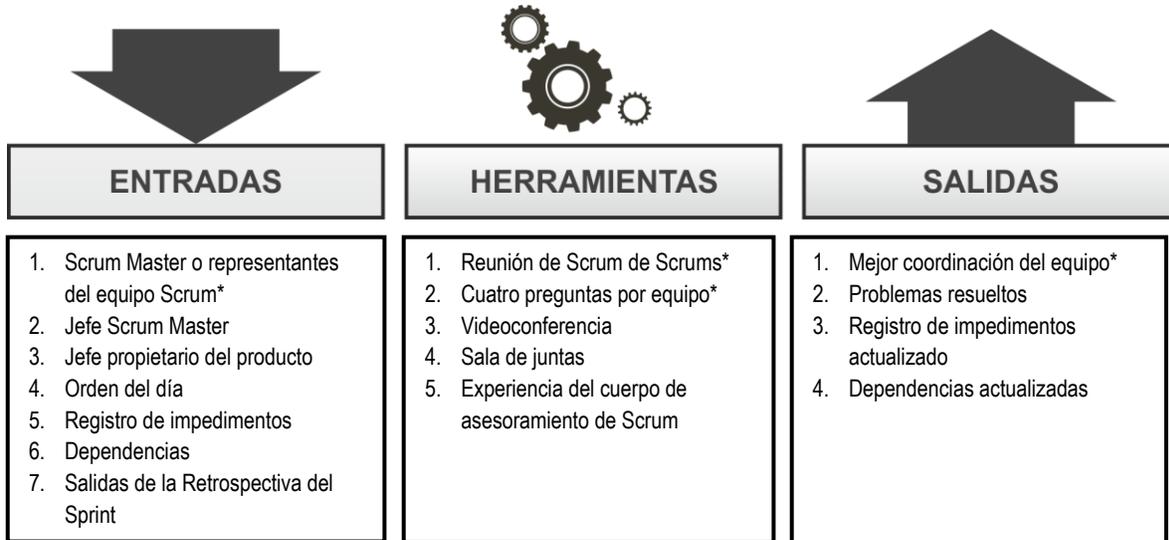


Figura 11-3: Convocar a un Scrum de Scrums—Entradas, herramientas y salidas

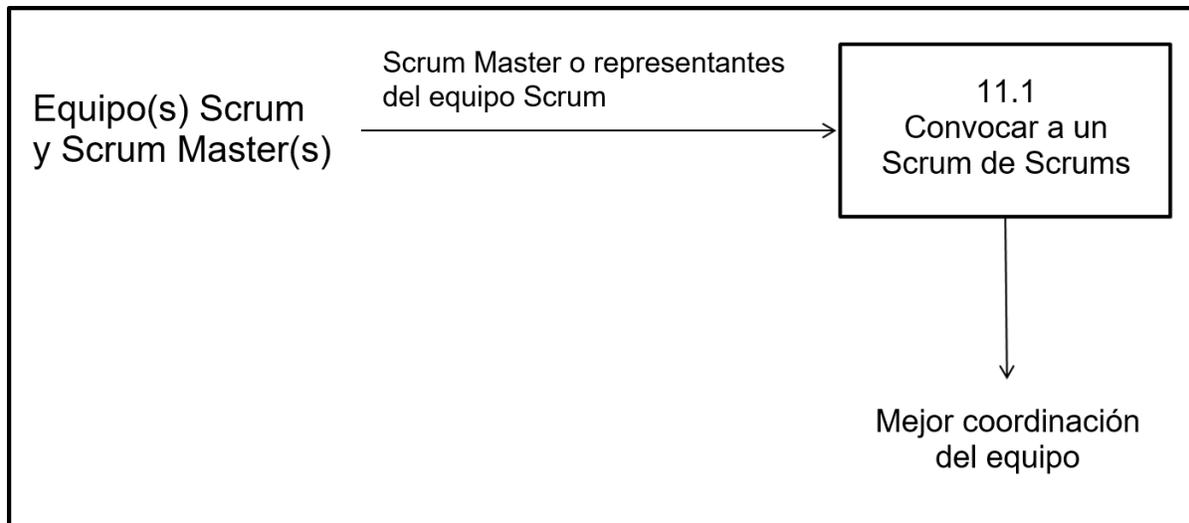


Figura 11-4: Convocar a un Scrum de Scrums—Diagrama de flujo de datos

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

## 11.1.1 Entradas

### 11.1.1.1 Scrum Master o representantes del equipo Scrum\*

Normalmente, uno de los miembros de cada equipo Scrum representará a su equipo en las reuniones de Scrum de Scrums. En la mayoría de los casos lo hace el Scrum Master, aunque en ocasiones pudiera representarlos alguien más. El equipo pudiera nombrar a una sola persona para que los represente en cada reunión o bien, el representante pudiera cambiar con el paso del tiempo basándose en quién pueda desempeñar mejor tal función dependiendo de las situaciones o circunstancias actuales. Cada persona que participa en esta reunión debe contar con un entendimiento técnico para poder identificar instancias en las cuales los equipos pudieran causar retrasos o impedimentos entre sí.

### 11.1.1.2 Jefe Scrum Master

Descrito en la sección 8.2.1.6.

### 11.1.1.3 Jefe propietario(s) del producto

Descrito en la sección 8.1.1.5

### 11.1.1.4 Orden del día

El propósito fundamental de una reunión de Scrum de Scrums es informar sobre el progreso entre los varios equipos. El jefe Scrum Master (o cualquier Scrum Master que conduzca la reunión) puede comunicar el orden del día con anticipación a la reunión. Esto le permite a cada equipo considerar los puntos en el orden del día y prepararse para la reunión. Cualquier impedimento que enfrente un equipo que pudiera también afectar a los otros, debe indicarse a fin de tratarlo durante la reunión de Scrum de Scrums. Asimismo, si el equipo se percata de un problema a gran escala, de un cambio, o de un riesgo que pudiera afectar a los demás equipos, tal situación debe comunicarse en la reunión de Scrum de Scrums.

### 11.1.1.5 Registro de impedimentos

Descrito en la sección 10.1.1.4.

### 11.1.1.6 Dependencias

Descritas en la sección 9.3.3.3.

### 11.1.1.7 Salidas de la retrospectiva del sprint

Las salidas del proceso de retrospectiva del sprint pudieran tener problemas que pudieran afectar a múltiples equipos Scrum y pudieran implementarse como entrada para reuniones eficaces de Scrum de Scrums.

## 11.1.2 Herramientas

### 11.1.2.1 Reunión de Scrum de Scrums

Son reuniones, de preferencia breves (pero por lo general no tienen un bloque de tiempo asignado a fin de permitir que se comparta más información entre los equipos), donde un representante de cada equipo Scrum se reúne para compartir el estado de sus respectivos equipos. La reunión de Scrum de Scrums se lleva a cabo en intervalos predeterminados o cuando así lo requieran los equipos Scrums para facilitar el intercambio de información entre los distintos equipos de Scrum. Los problemas, dependencias y riesgos que afectan a los varios equipos Scrum pueden monitorearse de cerca, lo cual ayuda a los distintos equipos que trabajan en un amplio proyecto a coordinarse mejor e integrar su trabajo. Es responsabilidad del jefe Scrum Master (u otro Scrum Master que facilite la reunión de Scrum de Scrums) asegurarse de que todos los representantes cuenten con un ambiente conductivo para un intercambio abierto y honesto de información, incluyendo retroalimentación a los demás representantes de los equipos. Para proyectos más grandes, la participación de una cantidad considerable de equipos, pueden convocarse varios niveles de tales reuniones para compartir el estado de sus respectivos equipos.

Las reuniones de Scrum de Scrums se describen más a detalle en la sección 3.7.2.

### 11.1.2.2 Cuatro preguntas por equipo\*

Cada representante del equipo Scrum brindará actualizaciones de su equipo. Estas actualizaciones se ofrecen generalmente en forma de respuestas a cuatro preguntas específicas:

- 1) ¿En qué ha estado trabajando mi equipo desde la última reunión?
- 2) ¿Qué va a hacer mi equipo hasta la próxima reunión?
- 3) ¿Qué esperaban otros equipos que mi equipo hiciera y que aún no se ha hecho?
- 4) ¿Qué planea hacer nuestro equipo que pudiera afectar a otros equipos?

Las respuestas a estas cuatro preguntas brindan información que permite a cada equipo entender claramente la situación de trabajo de todos los demás equipos.

### 11.1.2.3 Videoconferencia

Descrita en la sección 10.2.2.4

Es muy probable que la reunión Scrum de Scrums no sea cara a cara. Generalmente se necesita de una videoconferencia para los grandes proyectos donde hay mayor posibilidad de tener equipos distribuidos geográficamente.

#### **11.1.2.4 Sala de juntas**

Se recomienda tener una sala de juntas disponible para las reuniones de Scrum de Scrums donde todos los representantes del equipo Scrum se sientan cómodos.

#### **11.1.2.5 Experiencia del cuerpo de asesoramiento de Scrum**

También se describe en la sección 8.4.2.7.

En el proceso de convocar a un Scrum de Scrums, la experiencia del cuerpo de asesoramiento de Scrum pudiera relacionarse a las mejores prácticas documentadas sobre cómo llevar a cabo una reunión de Scrum de Scrums e incorporar las sugerencias de tales reuniones en el trabajo del proyecto para equipos individuales de Scrum. Pudiera haber también expertos en la materia que puedan ayudar al jefe Scrum Master a facilitar la reunión de Scrum de Scrums.

### **11.1.3 Salidas**

#### **11.1.3.1 Mejor coordinación del equipo**

La reunión de Scrum de Scrums facilita la coordinación de trabajo entre los varios equipos de Scrum. Eso resulta de gran importancia cuando existen tareas que implican independencias entre equipos. Las incompatibilidades y discrepancias entre el trabajo y los entregables de distintos equipos quedan rápidamente expuestas. Este foro brinda también a los equipos la oportunidad de mostrar sus logros y brindar retroalimentación a otros equipos. Por medio de las reuniones de Scrum de Scrums, existe una colaboración en toda la organización, a diferencia de cuando las personas trabajan en equipos cerrados preocupándose principalmente con sus responsabilidades individuales.

#### **11.1.3.2 Problemas resueltos**

La reunión de Scrum de Scrums es un foro donde los miembros del equipo Scrum tienen la oportunidad de debatir de forma transparente los problemas que impactan su proyecto. La necesidad de entregar a tiempo cada sprint fuerza a los equipos a confrontar activamente tales problemas en vez de posponer su resolución. Esta discusión y resolución oportuna de los problemas durante las reuniones de Scrum de Scrum mejora la coordinación entre los distintos equipos de Scrum y reduce también la necesidad de rediseñar y volver a realizar el trabajo. También se mitigan los riesgos relacionados a las dependencias y tablas de tiempos de entrega.

Las reuniones de Scrum de Scrum se describen detalle en la sección 3.7.2.1.

### **11.1.3.3 Registro de impedimento actualizado**

Descrito en la sección 10.1.3.3.

### **11.1.3.4 Dependencias actualizadas**

Descritas en la sección 9.3.3.3.

## 11.2 Demostración y validación del sprint

La figura 11-5 muestra todas las entradas, las herramientas y las salidas para el proceso de demostración y validación del Sprint.



Figura 11-5: Demostración y validación del sprint—Entradas, herramientas y salidas

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

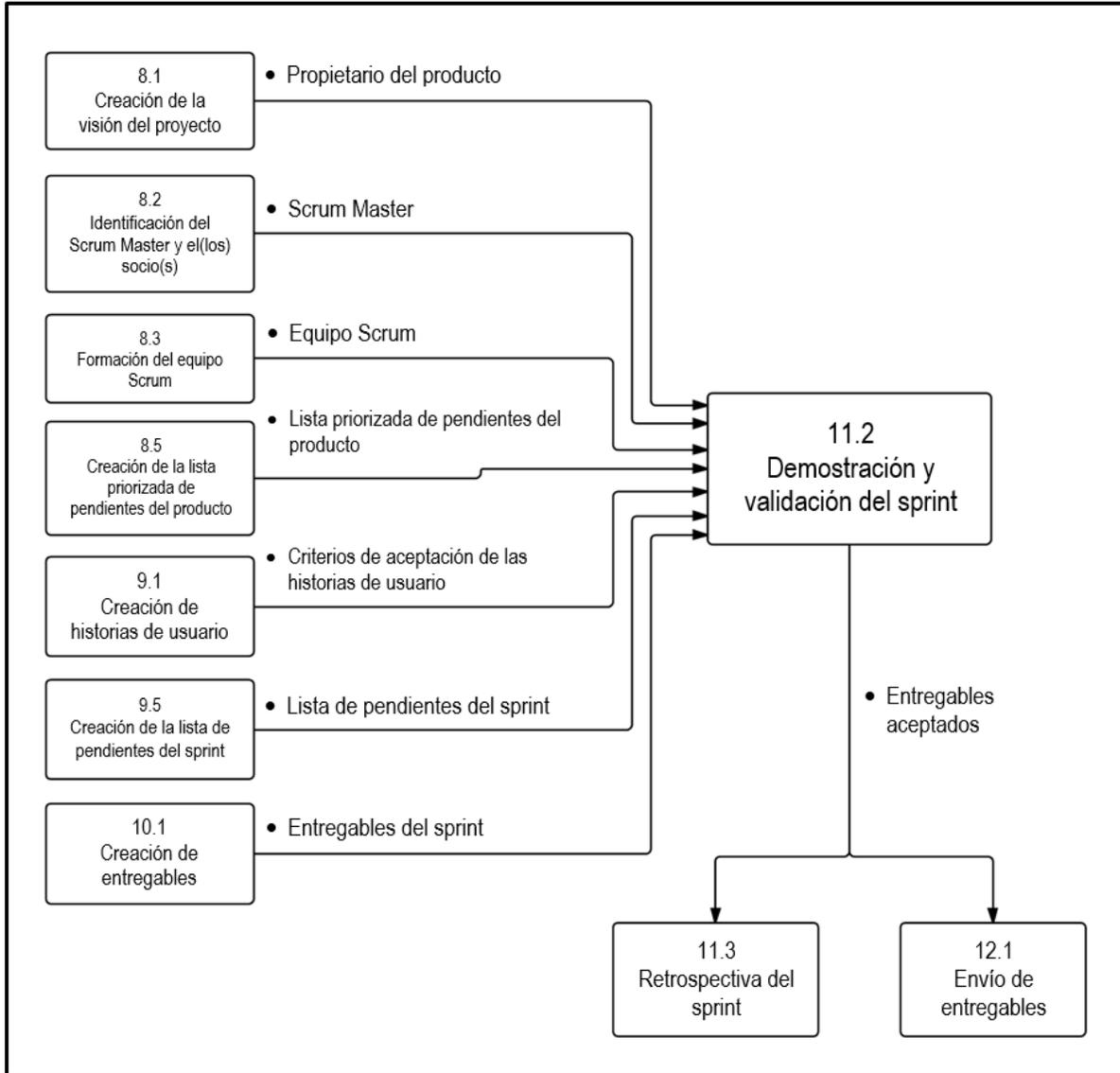


Figura 11-6: Demostración y validación del sprint—Diagrama de flujo de datos

## **11.2.1 Entradas**

### **11.2.1.1 Equipo principal de Scrum\***

Descrito en la sección 8.4.1.1.

### **11.2.1.2 Entregables del sprint\***

Descritos en la sección 10.1.3.1

### **11.2.1.3 Lista de pendientes del sprint\***

Descrita en la sección 9.5.3.1.

### **11.2.1.4 Criterios de terminado\***

Descritos en la sección 8.5.3.2.

### **11.2.1.5 Criterios de aceptación de las historias de usuario\***

Descritos en la sección 9.4.1.3.

### **11.2.1.6 Socio(s)**

Descrito en la sección 8.2.3.2.

### **11.2.1.7 Cronograma de planificación del lanzamiento**

Descrito en la sección 8.6.3.1.

### **11.2.1.8 Riesgos identificados**

Descrito en la sección 8.4.3.4.

### **11.2.1.9 Dependencias**

Descritas en la sección 9.3.3.3

### 11.2.1.10 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12

En el proceso de demostración y validación del sprint, las recomendaciones del cuerpo de asesoramiento de Scrum pueden incluir las mejores prácticas sobre cómo llevar a cabo las reuniones de revisión del sprint y evaluar los resultados del análisis del valor ganado. Pudiera haber también una orientación sobre como compartir experiencias con otras personas en el equipo principal de Scrum, así como demás equipos Scrum en el proyecto.

## 11.2.2 Herramientas

### 11.2.2.1 Reunión de revisión del sprint\*

Los miembros del equipo principal de Scrum y los socios relevantes participan en las reuniones de revisión del sprint para aceptar los entregables que cumplan con los criterios de aceptación de las historias de usuario y rechazar los entregables no aceptables. Tales reuniones se convocan al final de cada sprint. El equipo Scrum demuestra los logros del sprint, incluyendo las nuevas funcionalidades o los productos elaborados. Esto brinda una oportunidad para que el propietario del producto y el(los) socio(s) inspeccionen lo que se ha completado hasta el momento y determinen si deben realizarse cambios en el proyecto o en los procesos en sprints posteriores.

11

### 11.2.2.2 Análisis del valor ganado

Descrito en la sección 4.6.1

### 11.2.2.3 Experiencia del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.4.2.7.

En el proceso de demostración y validación del sprint, la experiencia del cuerpo de asesoramiento de Scrum podría relacionarse a las mejores prácticas documentadas sobre cómo llevar a cabo reuniones de revisión del sprint. Pudiera también haber expertos que pudieran brindar orientación sobre como facilitar de una mejor forma las reuniones de revisión del sprint.

## 11.2.3 Salidas

### 11.2.3.1 Entregables aceptados\*

Los entregables que cumplen con los criterios de aceptación de las historias de usuario son aceptados por el propietario del producto. El objetivo de un sprint es crear entregables potencialmente enviables o incrementos del producto que cumplan con los criterios de aceptación definidos por el cliente y el

propietario del producto. Estos se consideran entregables aceptados que pudieran ser entregados al cliente y así se desea. Se mantiene una lista de entregables aceptados y se actualiza después de cada reunión de revisión del sprint. Si un entregable no cumple con los criterios de aceptación definidos, no se considera aceptado y generalmente se llevará a un sprint posterior para corregir cualquier problema. Esto no es muy recomendable ya que el objetivo de cada sprint es que los entregables cumplan con los criterios de aceptación.

### **11.2.3.2 Entregables rechazados**

Si los entregables no cumplen con los criterios de aceptación, tales entregables se rechazan. Las historias de usuario asociados a tales entregables rechazados se agregan a la lista priorizada de pendientes del producto para que tales entregables puedan ser considerados como parte de un sprint posterior.

### **11.2.3.3 Riesgos actualizados**

Descrito en la sección 8.4.3.4.

### **11.2.3.4 Resultados del análisis del valor ganado**

Descrito en la sección 4.6.1.

### **11.2.3.5 Cronograma de la planificación del lanzamiento actualizado**

Descrito en la sección 10.3.3.2.

### **11.2.3.6 Dependencias actualizadas**

Descritas en la sección 9.3.3.3.

## 11.3 Retrospectiva del sprint

La figura 11-7 muestra todas las entradas, las herramientas y las salidas para el proceso retrospectiva de sprint.

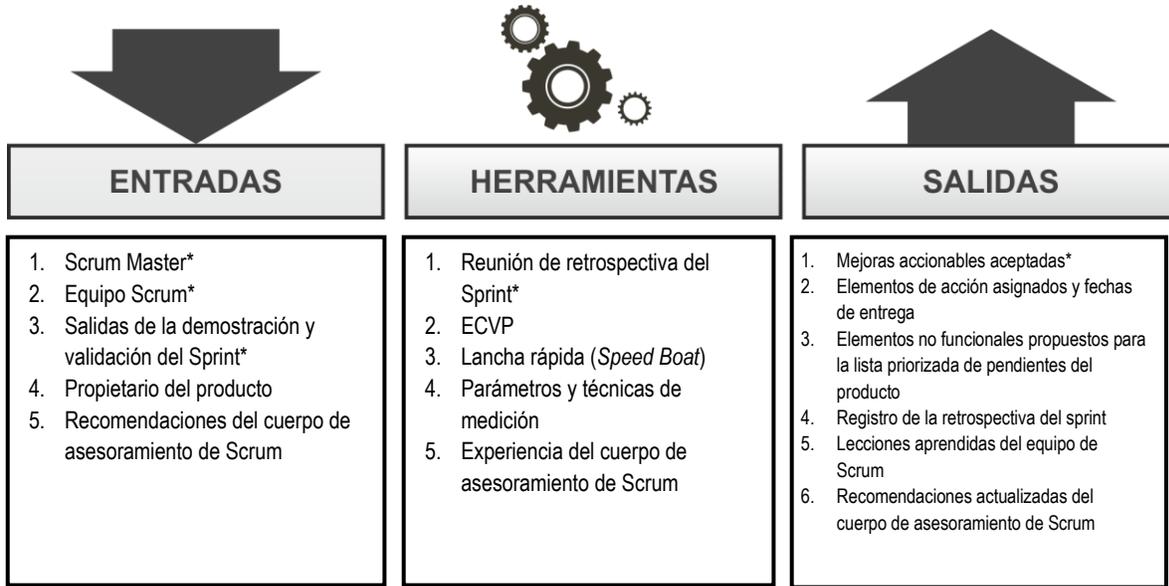


Figura 11-7: Retrospectiva del sprint—Entradas, herramientas y salidas

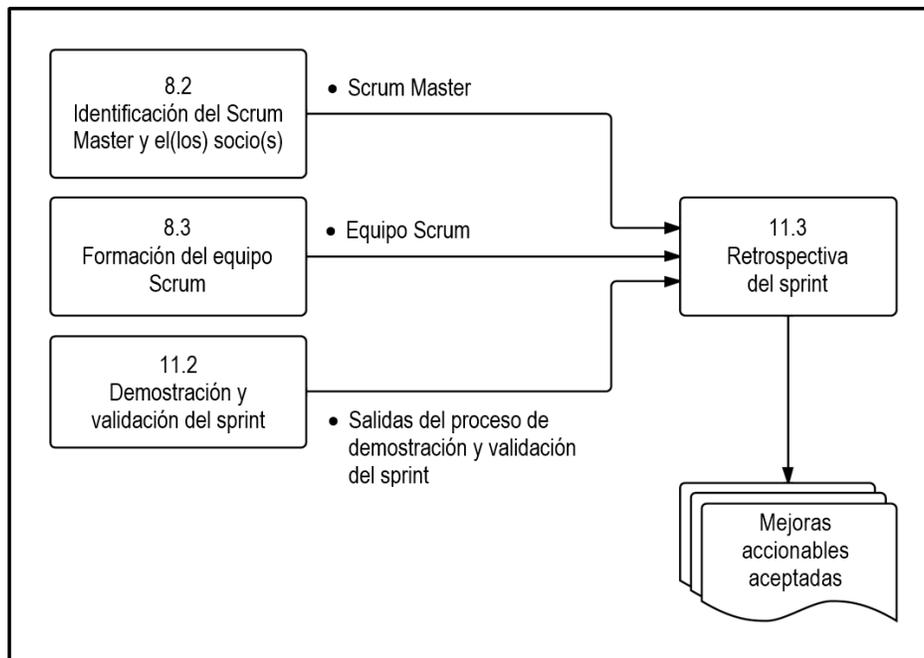


Figura 11-8: Retrospectiva del sprint—Diagrama de flujo de datos

Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.

## 11.3.1 Entradas

### 11.3.1.1 Scrum Master\*

Descrito en la sección 8.2.3.1.

### 11.3.1.2 Equipo Scrum\*

Descrito en la sección 8.3.3.1.

### 11.3.1.3 Salidas de la demostración y validación del sprint\*

Descrito en la sección 11.2.3.

Las salidas del proceso de demostración y validación del sprint brindan información útil al momento de llevar a cabo el proceso de retrospectiva del sprint.

### 11.3.1.4 Propietario del producto

Descrito en la sección 8.1.3.1.

### 11.3.1.5 Recomendaciones del cuerpo de asesoramiento de Scrum

El cuerpo de asesoramiento de Scrum puede brindar directrices para llevar a cabo las reuniones de retrospectiva del sprint, incluyendo sugerencias sobre herramientas a utilizarse, así como documentación o los entregables que se esperan de las reuniones.

## 11.3.2 Herramientas

### 11.3.2.1 Reunión de retrospectiva del sprint\*

La reunión de retrospectiva del sprint es un elemento importante del marco de “inspección-adaptación” de Scrum y es el último paso en un sprint. Todos los miembros del equipo Scrum asisten a la reunión, misma que organiza y modera el Scrum Master. Se recomienda que asista el propietario del producto, aunque no es obligatorio. Un integrante del equipo funge como secretario y documenta las discusiones y los elementos para acciones a futuro. Es esencial celebrar esta reunión en un entorno abierto y relajado a fin de fomentar la completa participación de todos los miembros del equipo. Las discusiones en la reunión de retrospectiva del sprint abarcan tanto lo que salió mal como lo que salió bien. Los objetivos primordiales de la reunión son identificar tres elementos específicos:

- 1) Las cosas que el equipo necesita seguir haciendo: mejores prácticas

- 2) Las cosas que el equipo necesita empezar a hacer: mejoras en el proceso
- 3) Las cosas que el equipo necesita dejar de hacer: problemas de proceso y embotellamiento

Estas áreas se analizan y se crea una lista de mejoras accionables aceptadas.

### 11.3.2.2 Explorador—Comprador—Vacacionista—Prisionero (ECVP)

Es un ejercicio que se puede llevar a cabo al inicio de la reunión de retrospectiva del sprint para entender la mentalidad de los participantes y establecer el tono de la reunión. Se les pide a los asistentes que indiquen de forma anónima lo que mejor representa su punto de vista en la reunión.

- Explorador—Quiere participar y aprender todo lo que se discutió en la retrospectiva
- Comprador—Quiere escuchar todo y elegir lo que entiende de la retrospectiva
- Vacacionista—Quiere relajarse y ser turista en la retrospectiva
- Prisionero—Quiere estar en otro lugar y asiste a la retrospectiva por ser obligatorio

El Scrum Master reúne las respuestas; prepara y comparte la información con el grupo.

### 11.3.2.3 Lancha rápida (del inglés: *Speed Boat*)

La “lancha rápida” (del inglés: *Speed Boat*) es una técnica que se puede utilizar para llevar a cabo la reunión de retrospectiva del sprint. Los miembros juegan el papel de la tripulación de la lancha. La lancha debe llegar a una isla: simbólicamente la visión del proyecto. Los asistentes utilizan notas adhesivas para llevar un registro de motores y anclas. Los “motores” ayudan a llegar a la isla, mientras que las “anclas” son cosas que están obstaculizando la llegada. Este ejercicio tiene un bloque de tiempo asignado de unos cuantos minutos. Una vez que se documentan todos los elementos, se reúne la información, se discute y se prioriza por medio de un proceso de votación. Se reconocen los motores y se planifican las acciones de mitigación para las anclas dependiendo de la prioridad.

### 11.3.2.4 Parámetros y técnicas de medición

Se pueden utilizar varios indicadores para medir y contrastar el desempeño del equipo en el actual sprint en comparación al rendimiento de sprints anteriores. Algunos ejemplos de dichos parámetros incluyen:

- Velocidad del equipo—Número de puntos de historia en un determinado sprint
- Tasa de terminado satisfactorio—Porcentaje de puntos de historia que se han terminado en comparación a los que se han asignado
- Eficacia de la estimación—Número o porcentaje de discrepancia entre el tiempo estimado y el tiempo actual en las tareas e historias de usuario.
- Clasificación de retroalimentación de revisiones—Se puede solicitar la retroalimentación de los socios utilizando clasificaciones cuantitativas o cualitativas, brindando una medida del desempeño del equipo.

- Clasificación moral del equipo—Resultados de una auto evaluación de la moral de los miembros del equipo.
- Retroalimentación de los compañeros—Se pueden implementar mecanismos de retroalimentación de 360 grados para solicitar una crítica constitutiva e información sobre el rendimiento del equipo.
- Progreso para la liberación o lanzamiento—El valor empresarial que se brinda en cada lanzamiento, así como el valor representado por el actual proceso encaminado a un lanzamiento. Esto contribuye a la motivación del equipo y al nivel de satisfacción de trabajo.

### 11.3.2.5 Experiencia del cuerpo de asesoramiento de Scrum

También se describe en la sección 8.4.2.7.

En el proceso de retrospectiva del sprint, la experiencia del cuerpo de asesoramiento de Scrum podría relacionarse a las mejores prácticas sobre cómo llevar a cabo reuniones de retrospectiva del sprint. Pudiera también haber expertos que pudieran brindar orientación sobre cómo utilizar las herramientas en el proceso de retrospectiva del sprint para entregar mejoras accionables aceptadas para los futuros sprints.

## 11.3.3 Salidas

### 11.3.3.1 Mejoras accionables aceptadas\*

Las mejoras accionables aceptadas (del inglés: *Agreed Actionable Improvements*) son los resultados primarios del proceso de retrospectiva del sprint. Forman parte de la lista de elementos accionables que ha elaborado el equipo para hacer frente a los problemas y mejorar los procesos a fin de mejorar también su desempeño en futuros sprints.

### 11.3.3.2 Elementos de acción asignados y fechas de entrega

Una vez que se han elaborado y refinado los elementos de acción asignados y las fechas límite, el equipo Scrum puede considerar los puntos de acción para implementar las mejoras. Cada elemento de acción contará con una fecha límite de conclusión.

### 11.3.3.3 Elementos no funcionales propuestos para la lista priorizada de pendientes del producto

Cuando se desarrolla la lista inicial priorizada de pendientes del producto, esta se basa en historias de usuario y en funcionalidades requeridas. Por lo general, los requerimientos no funcionales pudieran no estar completamente definidos en las primeras etapas del proyecto y pudieran surgir durante la revisión del sprint o en las reuniones de retrospectiva del sprint. Estos elementos deben añadirse a la lista priorizada de pendientes del producto a medida que se descubren. Algunos ejemplos de requerimientos no funcionales son: tiempos de respuesta, limitaciones en la capacidad y problemas en materia de seguridad.

#### 11.3.3.4 Registro de la retrospectiva del sprint

Los registros de la retrospectiva del sprint son registros de las opiniones, discusiones y elementos accionables planteados en la reunión de retrospectiva del sprint. El Scrum Master puede facilitar la creación de dicho registro con la aportación de los miembros del equipo principal de Scrum. La recopilación de todos los registros de la retrospectiva del sprint se convierte en el diario del proyecto y detalla los éxitos del mismo, los problemas y las resoluciones. Los registros son documentos públicos disponibles a todos en la organización.

#### 11.3.3.5 Lecciones aprendidas del equipo Scrum

Se espera que el equipo Scrum (equipo auto organizado y empoderado) aprenda de los errores cometidos durante el sprint y que estas lecciones aprendidas ayuden a mejorar su desempeño en futuros sprints. Estas lecciones aprendidas también pueden documentarse en las recomendaciones del cuerpo de asesoramiento de Scrum para compartirse con otros equipos de Scrum.

Puede haber varias lecciones positivas aprendidas como parte de un sprint. Estas lecciones positivas aprendidas son parte clave de la retrospectiva y deben compartirse adecuadamente dentro del equipo y con el cuerpo de asesoramiento de Scrum conforme el equipo avanza hacia una auto-mejora continua.

#### 11.3.3.6 Recomendaciones del cuerpo de asesoramiento de Scrum actualizadas

Como resultado de la reunión de retrospectiva del sprint, se pueden hacer sugerencias para revisar o mejorar las recomendaciones del cuerpo de asesoramiento de Scrum. Si el cuerpo de asesoramiento acepta las sugerencias, estas se incorporarán como actualizaciones a la documentación del cuerpo de asesoramiento de Scrum.

## 11.4 Diagrama de flujo de datos de fase

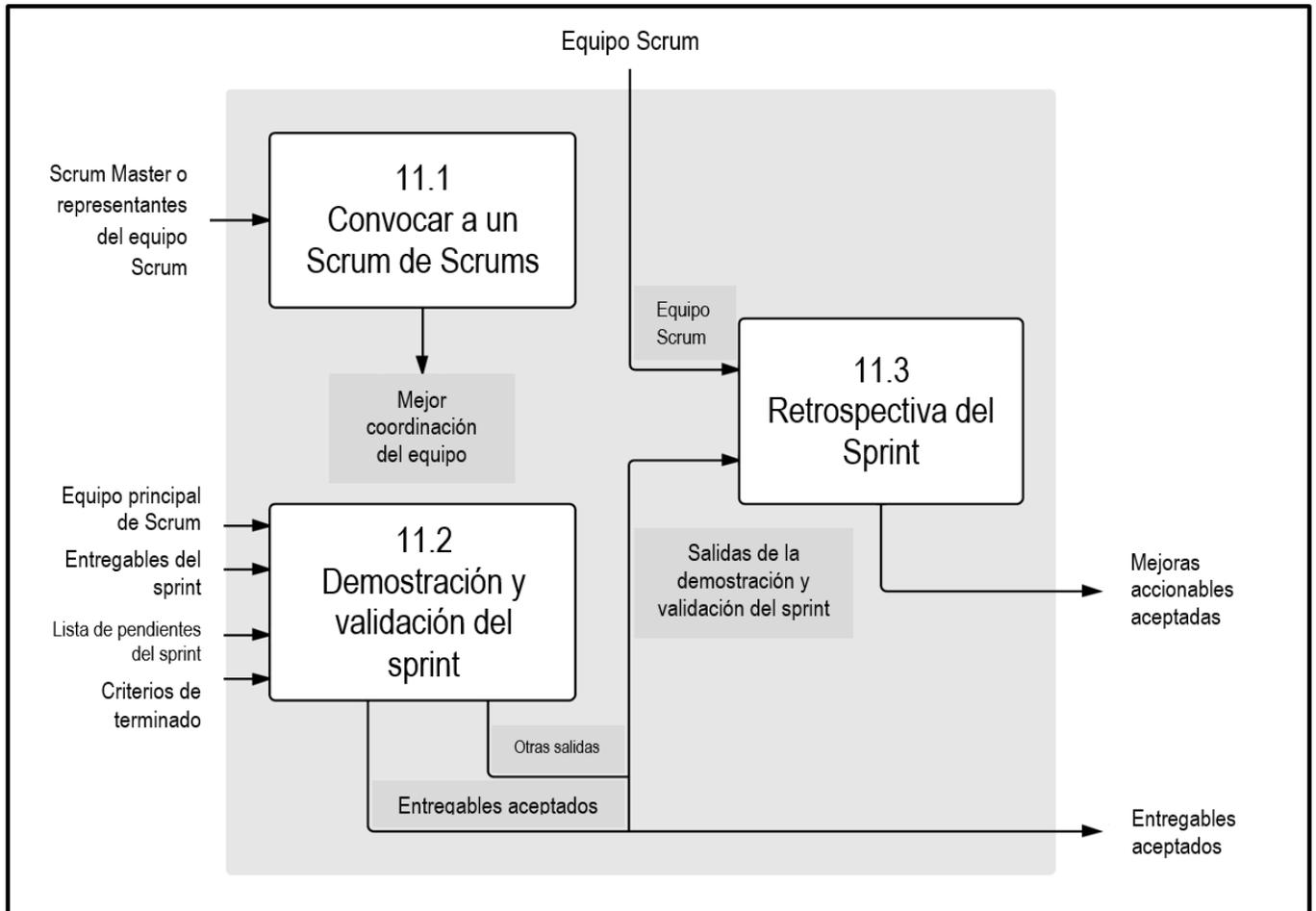


Figura 11-9: Fase de revisión y retrospectiva—Diagrama de flujo de datos

## 12. LANZAMIENTO

La fase de lanzamiento hace énfasis en la entrega al cliente de los entregables aceptados, así como en la identificación, documentación, e internalización de las lecciones aprendidas durante el proyecto.

El *lanzamiento*, tal como se define en la *Guía para el conocimiento de Scrum (Guía SBOK™)*, es aplicable a los siguientes:

- Carteras, programas y/o proyectos en cualquier industria;
- Productos, servicios o cualquier otro resultado que se les entregará a los socios;
- Proyectos de cualquier tamaño o complejidad.

El término “producto” en la *Guía SBOK™* puede referirse a un producto, servicio, o cualquier otro entregable. Scrum se puede aplicar de manera efectiva a cualquier proyecto en cualquier industria: desde proyectos o equipos pequeños con tan sólo seis miembros, hasta proyectos grandes y complejos con cientos de miembros por equipo.

Para facilitar la mejor aplicación del marco de Scrum, en este capítulo se identifican las entradas, herramientas y salidas de cada proceso, ya sea como “obligatorias” u “opcionales”. Las entradas, herramientas y salidas indicadas con asteriscos (\*) son obligatorias, mientras que las que no tienen asteriscos son opcionales.

Se recomienda que el equipo Scrum y aquellas personas que empiezan a aprender sobre el marco y los procesos de Scrum, se enfoquen principalmente en las entradas, herramientas y salidas obligatorias; mientras que los propietarios del producto, Scrum Masters y demás practicantes con experiencia en Scrum, se deberían de esforzar por alcanzar un conocimiento más a fondo sobre la información ofrecida en este capítulo. También es importante saber que, aunque todos los procesos se definen de forma única en la *Guía SBOK™*, no necesariamente se llevan a cabo de forma secuencial o por separado. En ocasiones, puede ser más apropiado combinar algunos procesos, dependiendo de los requisitos específicos de cada proyecto.

Este capítulo está escrito desde la perspectiva de un equipo Scrum que trabaja en un sprint para producir entregables que se pueden enviar como parte de un proyecto más amplio. Sin embargo, la información que se describe aplica de igual forma a proyectos enteros, programas y carteras. Hay información adicional relacionada con el uso de Scrum para proyectos, programas y portfolios que está disponible en los capítulos 2 al 7, y que cubren los principios y aspectos de Scrum.

La figura 12-1 brinda una visión general de los procesos de la fase de lanzamiento, los cuales son los siguientes:

**12.1 Envío de entregables**—En este proceso se hace la entrega o la transición de los entregables aceptados a los socios relevantes. La conclusión satisfactoria del sprint se documenta en un acuerdo formal de entregables funcionales.

**12.2 Retrospectiva del proyecto**—En este proceso, en el cual se concluye el proyecto, los socios de la organización y los miembros del equipo principal de Scrum se reúnen para hacer una retrospectiva del proyecto e identificar, documentar e internalizar las lecciones que se aprendieron. Generalmente, dichas lecciones permiten documentar las mejoras accionables aceptadas e implementarlas en futuros proyectos.

12.1 Envío de entregables	12.2 Retrospectiva del Proyecto
<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Propietario del producto*</li> <li>2. Socios(s)*</li> <li>3. Entregables aceptados*</li> <li>4. Cronograma de planificación del lanzamiento*</li> <li>5. Scrum Master</li> <li>6. Equipo Scrum</li> <li>7. Criterios de aceptación de las historias del usuario</li> <li>8. Plan de pilotaje</li> <li>9. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Métodos de desplazamiento organizacional*</li> <li>2. Plan de comunicación</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Acuerdo de entregables funcionales*</li> <li>2. Entregables funcionales</li> <li>3. Lanzamientos del producto</li> </ol>	<p><b>ENTRADAS</b></p> <ol style="list-style-type: none"> <li>1. Equipo principal de Scrum*</li> <li>2. Jefe Scrum Master</li> <li>3. Jefe propietario del producto</li> <li>4. Socio(s)</li> <li>5. Recomendaciones del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>HERRAMIENTAS</b></p> <ol style="list-style-type: none"> <li>1. Reunión de la retrospectiva del proyecto*</li> <li>2. Otras herramientas para la retrospectiva del proyecto</li> <li>3. Experiencia del cuerpo de asesoramiento de Scrum</li> </ol> <p><b>SALIDAS</b></p> <ol style="list-style-type: none"> <li>1. Mejoras accionables aceptadas*</li> <li>2. Elementos de acción asignados y fechas límite*</li> <li>3. Elementos no funcionales propuestos para la lista priorizada de productos del programa y la lista priorizada de pendientes del producto</li> <li>4. Recomendaciones actualizadas del cuerpo de asesoramiento de Scrum</li> </ol>

**Figura 12-1: Resumen del lanzamiento**

*Nota: Los asteriscos (\*) denotan una entrada, herramientas o salida "obligatoria" para el proceso correspondiente.*

La figura 12-2 muestra las entradas obligatorias, herramientas y salidas para los procesos en fase de lanzamiento.



Figura 12-2: Resumen del lanzamiento (Esenciales)

*Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.*

## 12.1 Envío de entregables

La figura 12-3 muestra todas las entradas, las herramientas y las salidas para el proceso de envío de entregables.

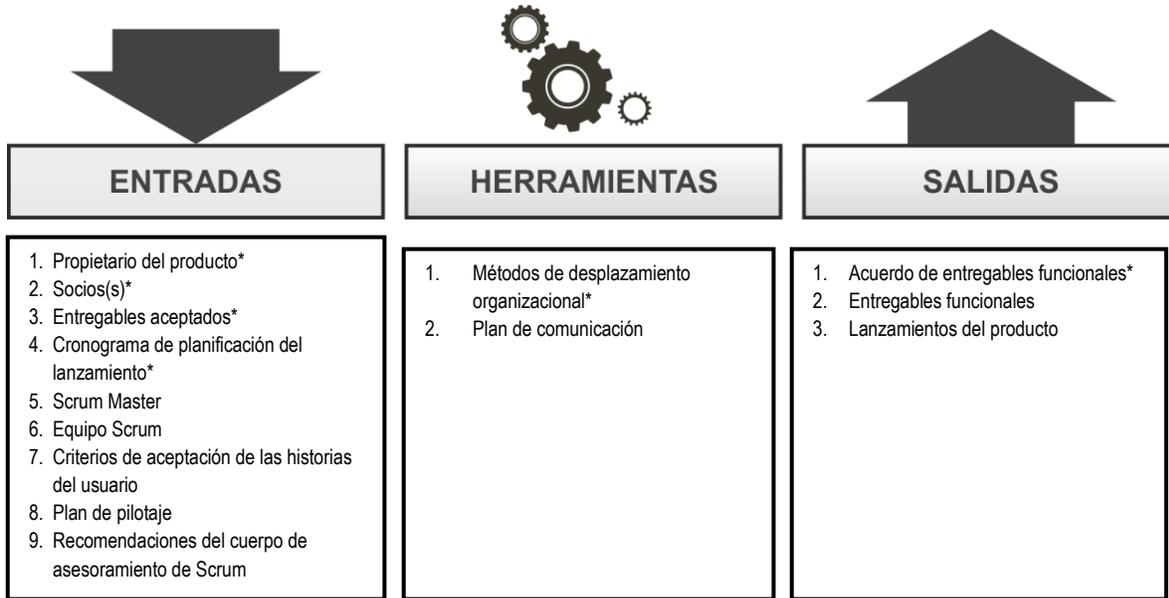


Figura 12-3: Envío de entregables—Entradas, Herramientas y Salidas

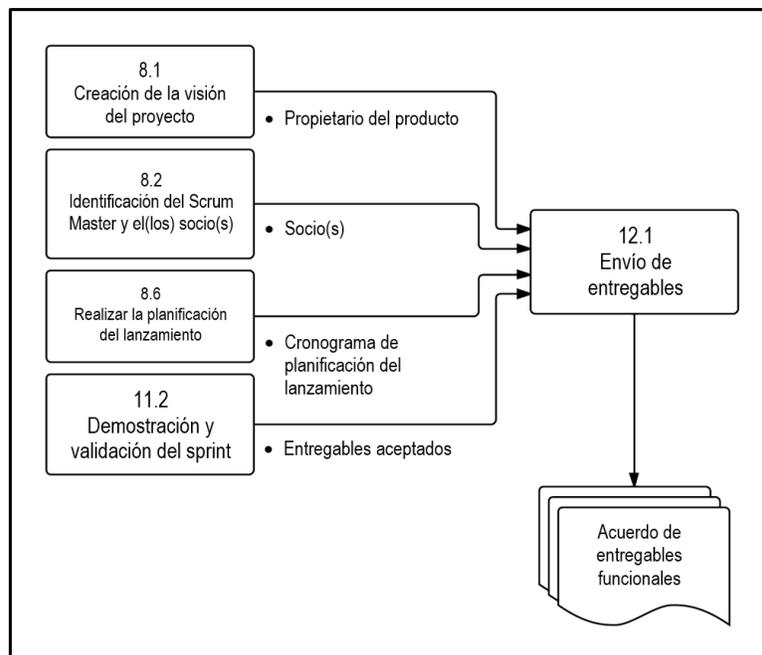


Figura 12-4: Envío de entregables—Diagrama de flujo de datos

Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.

## 12.1.1 Entradas

### 12.1.1.1 Propietario del producto\*

Descrito en la sección 8.1.3.1.

### 12.1.1.2 Socio(s)\*

Descrito en la sección 8.2.3.2.

### 12.1.1.3 Entregables aceptados\*

Descrito en la sección 11.2.3.1.

### 12.1.1.4 Cronograma de planificación del lanzamiento

Descrito en la sección 8.6.3.1.

### 12.1.1.5 Scrum Master

Descrito en la sección 8.2.3.1.

### 12.1.1.6 Equipo Scrum

Descrito en la sección 8.3.3.1.

### 12.1.1.7 Criterios de aceptación de las historias de usuario

Descrito en la sección 9.1.3.2.

### 12.1.1.8 Plan de pilotaje

El plan pilotaje (conocido en inglés como: *Piloting Plan*) es una entrada opcional que se puede utilizar para trazar un desplazamiento piloto a detalle. El alcance y los objetivos del desplazamiento, la base de usuarios seleccionada para la implementación, un cronograma del desplazamiento, los planes de transición, la preparación requerida del usuario, los criterios de evaluación para el desplazamiento y otros elementos clave relacionados al desplazamiento, se identifican en el plan de pilotaje y se comparten con los socios.

### 12.1.1.9 Recomendaciones del cuerpo de asesoramiento de Scrum

Descrito en la sección 8.1.1.12.

En el proceso de envío de entregables, el cuerpo de asesoramiento de Scrum puede brindar recomendaciones y directrices para el desplazamiento de productos. Estas son las mejores prácticas que se deben tomar en cuenta al implementar un producto al cliente con el fin de maximizar el valor entregado.

## 12.1.2 Herramientas

### 12.1.2.1 Métodos de desplazamiento organizacional\*

Los mecanismos de desplazamiento de cada organización tienden a ser diferentes basados en su respectiva industria, en los usuarios meta y el posicionamiento. Dependiendo del producto a entregarse, el desplazamiento puede ser remoto o puede incluir el envío físico o la transición de un artículo. Debido a que la implementación tiende a implicar un alto nivel de riesgo, las organizaciones suelen tener mecanismos de implementación bien definidos y establecidos, con procesos detallados para garantizar el cumplimiento de todas las normas aplicables y medidas de garantía de calidad. Estas podrían incluir aprobaciones finales de los representantes específicos de gestión, mecanismos de aprobación de usuario y directrices para la funcionalidad mínima de un lanzamiento.

### 12.1.2.2 Plan de comunicación

En muchos proyectos existe un plan de comunicación. El plan especifica los registros que se deben crear y mantener durante todo el proyecto. Se utiliza una variedad de métodos para transmitir a los socios información importante sobre el proyecto. El plan de comunicación define los métodos e indica quién es responsable de las distintas actividades de comunicación. A medida que se evalúan los entregables, el estado de las actividades de examinación se comunica mediante el plan de comunicación según lo determinado por el propietario del producto y el patrocinador. Un mecanismo común de comunicación es una representación visual que muestra información importante y en un formato fácil de interpretar, publicado en un lugar accesible y que se mantiene al día con la información más actualizada.

## 12.1.3 Salidas

### 12.1.3.1 Acuerdo de entregables funcionales\*

Los entregables que cumplen con los criterios de aceptación, reciben el cierre formal del negocio y la aprobación formal por parte del cliente o del patrocinador. Obtener la aceptación formal del cliente es fundamental para el reconocimiento de los ingresos. La responsabilidad de obtener dicha aceptación se define en las políticas de la empresa y no es necesariamente la responsabilidad del propietario del producto.

### 12.1.3.2 Entregables funcionales

Esta salida es el entregable final que se puede enviar (del inglés: *shippable deliverable*) para el cual fue sancionado el proyecto. A medida que se crean nuevos incrementos de los productos, estos se integran continuamente a los incrementos anteriores, por lo que hay un producto potencialmente entregable disponible en todo momento a lo largo del proyecto.

### 12.1.3.3 Lanzamientos del producto

Los lanzamientos de productos deben incluir lo siguiente:

- Contenido del lanzamiento—Esto consiste en información esencial sobre los entregables que pudiera ayudar al equipo de atención al cliente.
- Notas de lanzamiento—Las notas de lanzamiento deben incluir criterios de envío externos de cara al mercado para los productos que serán entregados.

## 12.2 Retrospectiva del proyecto

La figura 12-5 muestra todas las entradas, las herramientas y las salidas para el proceso de retrospectiva del proyecto.

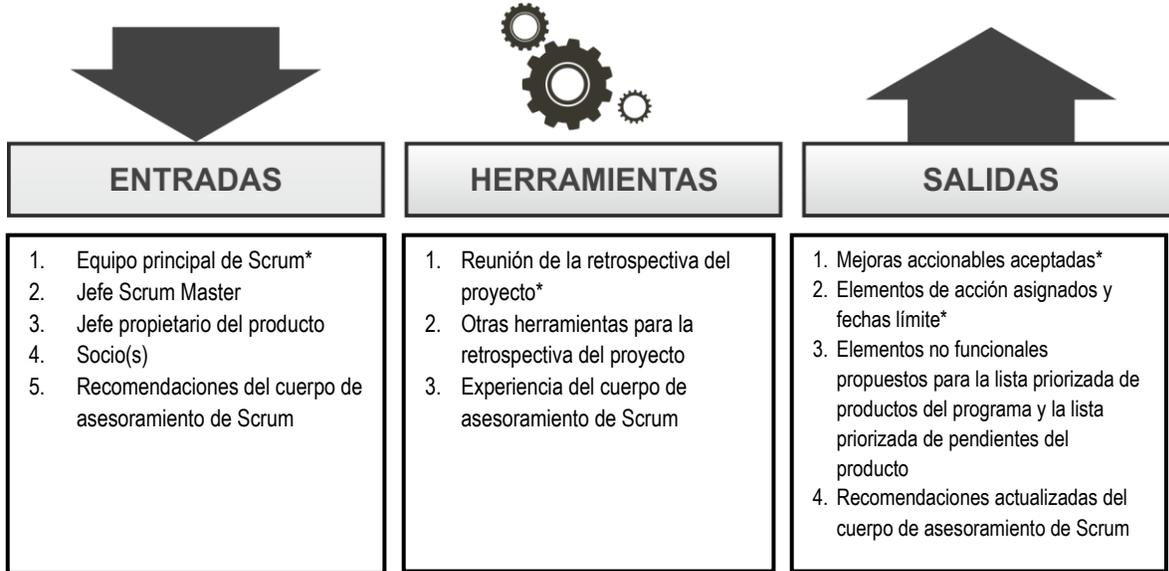


Figura 12-5: Retrospectiva del proyecto—Entrada, herramientas y salidas

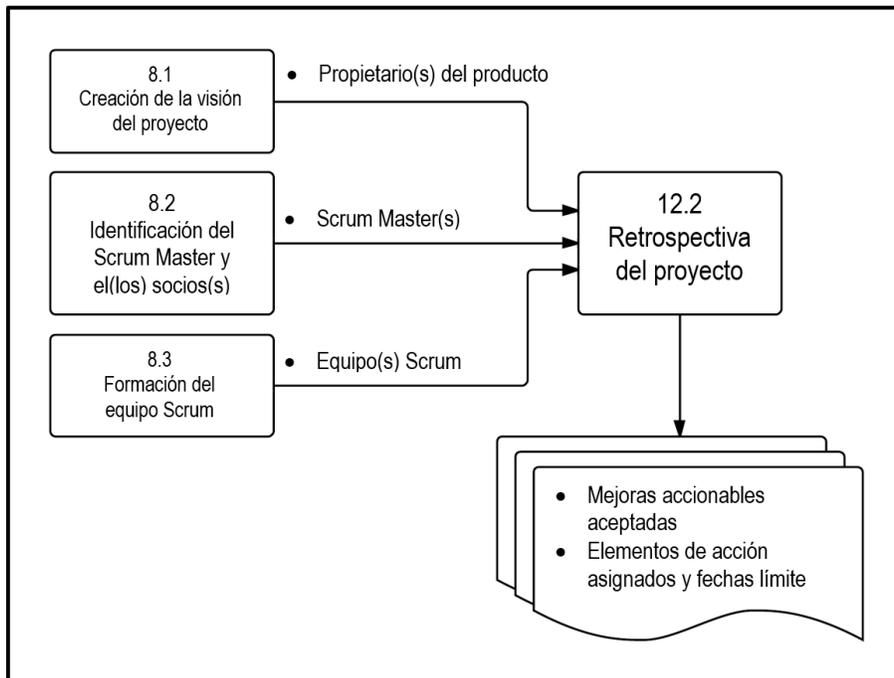


Figura 12-6: Retrospectiva del proyecto—Diagrama de flujo de datos

Nota: Los asteriscos (\*) denotan una entrada, herramienta o salida "obligatoria" para el proceso correspondiente.

## **12.2.1 Entradas**

### **12.2.1.1 Equipo(s) principal de Scrum\***

Descrito en la sección 8.4.1.1.

### **12.2.1.2 Jefe Scrum Master**

Descrito en la sección 8.2.1.6.

### **12.2.1.3 Jefe propietario del producto**

Descrito en la sección 8.1.1.5.

### **12.2.1.4 Socio(s)**

Descrito en la sección 8.2.3.2.

### **12.2.1.5 Recomendaciones del cuerpo de asesoramiento de Scrum**

Descrito en la sección 8.1.1.12.

En el proceso de retrospectiva del proyecto, las recomendaciones del cuerpo de asesoramiento de Scrum pueden incluir un repositorio de plantillas internas que apoyan los proyectos futuros y brindan orientación para la realización de la reunión de retrospectiva del proyecto. La orientación que se brinda puede estar relacionada a procedimientos administrativos, auditorías, evaluaciones y criterios de transición del proyecto. Generalmente incluyen también la forma en la que la organización habrá de mantener el conocimiento base de las lecciones aprendidas y la información de todos los proyectos.

## **12.2.2 Herramientas**

### **12.2.2.1 Reunión de retrospectiva del sprint\***

La reunión de retrospectiva del sprint es una reunión para determinar las formas en las que la colaboración y eficacia del equipo puede mejorarse en futuros proyectos. También se analizan las oportunidades positivas, negativas y potenciales para mejorar. Esta reunión no tiene un bloque de tiempo asignado y se puede llevar a cabo en forma presencial o en formato virtual. Entre los asistentes se encuentran: el equipo del proyecto, el jefe Scrum Master, el jefe propietario del producto y el (los) socio(s). Durante la reunión, se documentan las lecciones aprendidas y los participantes buscan oportunidades para mejorar los procesos y atender las ineficiencias.

### 12.2.2.2 Otras herramientas para la retrospectiva del proyecto

Algunas de las herramientas utilizadas en el proceso de retrospectiva del sprint también se pueden utilizar en este proceso. A continuación, se ofrecen ejemplos:

- Ejercicios de Explorador—Comprador—Vacacionista—Prisionero (ECVP)
- Lancha rápida (*Speed Boat*)
- Parámetros y técnicas de medición

### 12.2.2.3 Experiencia del cuerpo de asesoramiento de Scrum

Analizado en la sección 8.4.2.7.

En el proceso de retrospectiva del proyecto, la responsabilidad principal del cuerpo de asesoramiento de Scrum es asegurar que las lecciones aprendidas en cada proyecto no se hayan perdido y que estén integradas en la organización.

Asimismo, un cuerpo de asesoramiento puede aportar su experiencia en diversos ámbitos, entre ellos: la calidad, recursos humanos y Scrum, que pudiera ser útil en el proceso de retrospectiva del proyecto. Además, el cuerpo de asesoramiento de Scrum puede ofrecer sugerencias respecto a cómo se debe llevar a cabo la reunión de retrospectiva del proyecto.

## 12.2.3 Salidas

### 12.2.3.1 Mejoras accionables aceptadas\*

Descrito en la sección 11.3.3.1.

### 12.2.3.2 Elementos de acción asignados y fechas límite\*

Descrito en la sección 11.3.3.2.

### 12.2.3.3 Elementos no funcionales propuestos para la lista priorizada de productos del programa y la lista priorizada de pendientes del producto

Cuando se desarrolla inicialmente la lista de pendientes del producto del programa (*Program Product Backlog*) o la lista priorizada de pendientes del producto (*Prioritized Product Backlog*), estas se basan en las historias de usuario y en las funcionalidades requeridas. Por lo general los requerimientos no funcionales pudieran no estar completamente definidos en las primeras etapas del proyecto y pudieran surgir durante la revisión del sprint, la retrospectiva del sprint o en las reuniones de retrospectiva del proyecto. Dichos elementos deben agregarse a la lista de pendientes del producto del programa (para el programa) y en la lista priorizada de pendientes del producto (para el proyecto) conforme se van encontrando. Algunos

ejemplos de requerimientos no funcionales son: tiempos de respuesta, limitaciones en la capacidad y asuntos en materia de seguridad.

### 12.2.3.4 Recomendaciones actualizadas del cuerpo de asesoramiento de Scrum

Descritas en las secciones 8.1.1.12 y 11.3.3.5

## 12.3 Diagrama de flujo de datos de fase

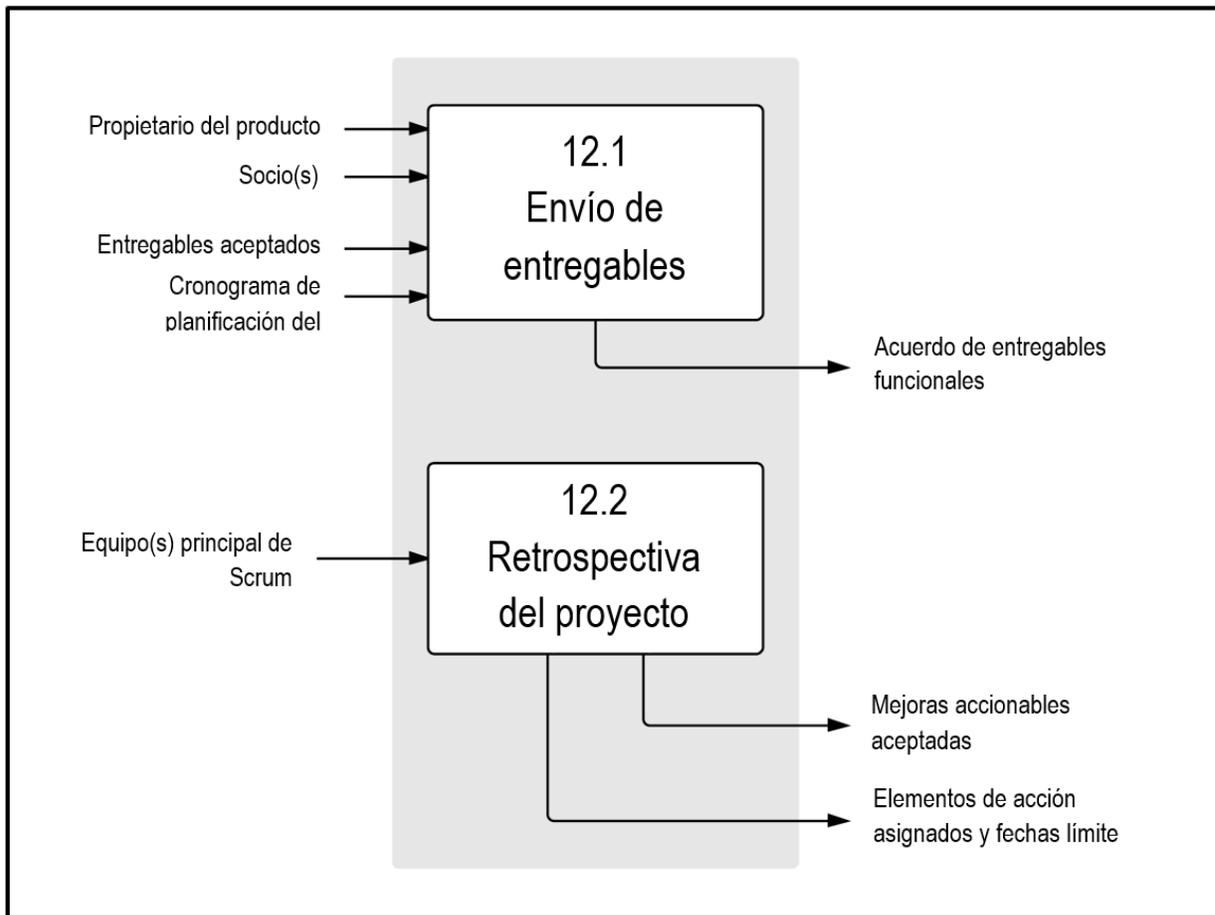


Figura 12-7: Fase de lanzamiento—Diagrama de flujo de datos

# APÉNDICE A. RESUMEN DE ÁGIL

## A.1 Introducción

Este apéndice tiene la intención de familiarizar al lector con el concepto de desarrollo Ágil y las diversas metodologías ágiles.

En las siguientes secciones se incluyen:

**A.2 Resumen**—Esta sección analiza la definición y los factores detrás del surgimiento de Ágil.

**A.3 Manifiesto Ágil**—En esta sección se presenta el *Manifiesto Ágil*, sus principios, y la *Declaración de Interdependencia* para proporcionar el contexto histórico de Ágil.

**A.4 Métodos Ágil**—Esta sección ofrece una breve descripción de las metodologías ágiles específicas tales como:

- **Lean Kanban**
- **Programación extrema** (*Extreme Programming*)
- **Métodos Crystal** (*Crystal Methods*)
- **Métodos de desarrollo de sistemas dinámicos** (*Dynamic Systems Development Methods*)
- **Desarrollo orientado a funcionalidades** (*Feature Driven Development*)
- **Desarrollo guiado por pruebas** (*Test Driven Development*)
- **Desarrollo adaptativo de software** (*Adaptive Software Development*)
- **Proceso unificado Ágil** (*Agile Unified Process*)
- **Desarrollo guiado por el dominio** (*Domain Driven Development*)

## A.2 Resumen

La palabra “ágil” generalmente hace referencia a la capacidad de moverse o responder con rapidez y facilidad; ser ágil. En cualquier tipo de disciplina de administración, Ágil es una calidad, y, por lo tanto, es algo bueno que se debe buscar. Específicamente, la administración ágil de proyectos implica adaptarse durante la creación de un producto, servicio u otro resultado.

Es importante entender que, aunque el desarrollo de métodos ágiles es altamente adaptativo, también es necesario considerar la estabilidad en su proceso de adaptación.

### A.2.1 El surgimiento de Ágil

Los rápidos cambios en la tecnología, las demandas y expectativas del mercado han creado mayores desafíos en el desarrollo de productos y servicios mediante el uso de modelos tradicionales de gestión de proyectos. Esto abrió el camino para la conceptualización e implementación de métodos y valores ágiles en muchas organizaciones. Los modelos de desarrollo ágil atienden las deficiencias asociadas con los modelos tradicionales de gestión de proyectos para satisfacer las crecientes demandas ambientales y expectativas que enfrentan las organizaciones. Dado a que los modelos tradicionales de gestión de proyectos en general hacen hincapié en una amplia planificación anticipada y que se ajustan al plan una vez que se establece, tales modelos no tuvieron éxito al intentar cumplir la realidad de los frecuentes cambios ambientales.

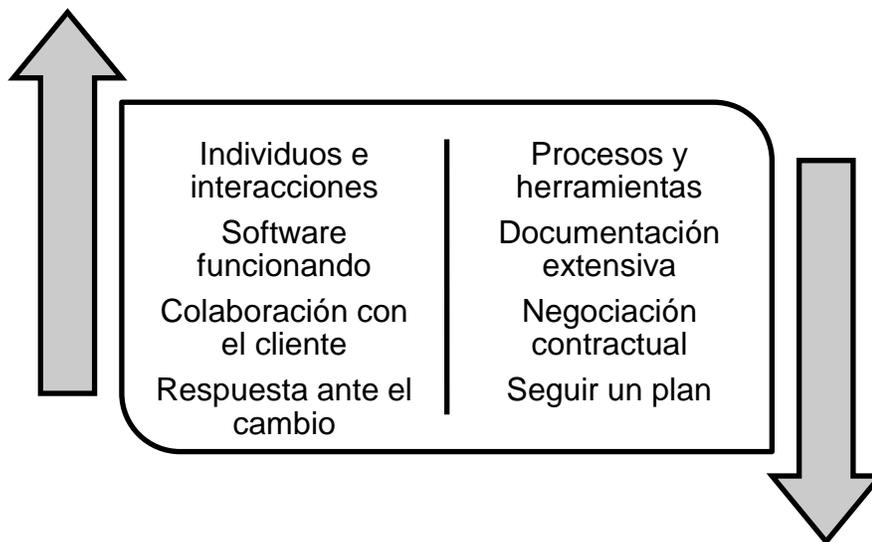
Ágil depende de la planificación adaptativa y del desarrollo y la entrega iterativa. Se centra principalmente en el valor de las personas al hacer eficazmente el trabajo. Aunque las metodologías adaptativas e incrementales han existido desde los años cincuenta, únicamente las metodologías que conforman el Manifiesto Ágil generalmente se consideran verdaderamente “ágiles”.

## A.3 El Manifiesto Ágil

En febrero del 2001, un grupo de 17 gurús de la informática, desarrolladores de software y administradores, se reunieron para discutir los métodos de desarrollo de software ligero. Formaron la Alianza Ágil (*Agile Alliance*) y las discusiones en estas reuniones resultaron después en un Manifiesto para el desarrollo ágil de software (*Manifiesto for Agile Software Development*). El manifiesto fue escrito por Fowler y Highsmith (2001) y suscrito después por todos los participantes a fin de establecer los lineamientos básicos de cualquier metodología ágil.

El propósito del Manifiesto Ágil se plasmó de la siguiente forma:

*Estamos descubriendo formas mejores de desarrollar software tanto por nuestra propia experiencia como ayudando a terceros. A través de este trabajo hemos aprendido a valorar:*



*Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda.*

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

El permiso para su reproducción fue proporcionado por los autores mencionados mediante aviso en:  
<http://agilemanifesto.org/>

Las cuatro contrapartes del Manifiesto Ágil se elaboran de la siguiente forma:

**1. Individuos e interacciones sobre procesos y herramientas**

Aunque los procesos y las herramientas ayudan a terminar con éxito un proyecto, son las personas quienes asumen, participan e implementan un proyecto y determinan cuáles procesos y herramientas utilizar. Por lo tanto, los actores clave en cualquier proyecto son las personas, por ello, el énfasis debe estar en ellos y en sus interacciones en vez de los complicados procesos y herramientas.

**2. Software funcionando sobre documentación extensiva**

Aunque la documentación es necesaria y útil para cualquier proyecto, muchos equipos se centran en la recopilación y el registro de descripciones cualitativas y cuantitativas de los entregables, cuando el valor real que se le entrega al cliente es en forma de un software funcional. Por lo tanto, en vez de la documentación detallada, el enfoque ágil está en la entrega de un software de buen funcionamiento en incrementos a lo largo del ciclo de vida del producto.

**3. Colaboración con el cliente sobre negociación contractual**

Tradicionalmente a los clientes se les ha visto como participantes externos, involucrados principalmente al inicio y al final del ciclo de vida del producto y cuya relación se basaba en contratos y en su cumplimiento. Ágil cree en un enfoque de valor compartido en el cual los clientes se consideran colaboradores. El equipo de desarrollo y el cliente trabajan unidos para evolucionar y desarrollar el producto.

**4. Responder ante el cambio sobre seguir un plan**

En mercado actual —donde los requerimientos del cliente, las tecnologías disponibles y los patrones empresariales cambian constantemente—, es fundamental abordar el desarrollo de productos en una forma adaptativa que permita la incorporación de cambio y rápidos ciclos de vida de desarrollo de producto en vez de enfatizar el seguimiento de planes formados probablemente con información obsoleta.

### A.3.1 Principios de Manifiesto Ágil

Los 12 principios del Manifiesto Ágil de Fowler y Highsmith (2001) son los siguientes:

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
2. Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
3. Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.
4. Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.
5. Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.
6. El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.
7. El software funcionando es la medida principal de progreso.
8. Los procesos ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.
9. La atención continua a la excelencia técnica y al buen diseño mejora la agilidad.
10. La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.
11. Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.
12. A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.

### A.3.2 Declaración de interdependencia

La Declaración de Interdependencia de gestión de proyectos ágiles fue redactada a principios del 2005 por un grupo de 15 líderes de proyecto como suplemento del Manifiesto Ágil. Enumera seis valores administrativos necesarios para reforzar la mentalidad del desarrollo ágil, particularmente en la gestión de proyectos complejos o inciertos.

La declaración destaca que los equipos del proyecto, los clientes y demás socios son interdependientes y están conectados, lo cual se debe reconocer a fin de lograr el éxito. Los valores por si mismos son también interdependientes.

*Nosotros...*

**Aumentamos el retorno sobre la inversión** enfocándonos en la aportación continua de valor.

**Entregamos resultados fiables** mediante la participación de clientes en las interacciones frecuentes, así como la propiedad compartida.

**Esperamos la incertidumbre** y la gestionamos a través de iteraciones, con anticipación y adaptación.

**Damos rienda suelta a la creatividad y la innovación** reconociendo que las personas son la fuente última de valor, y creando un ambiente donde pueden ser la diferencia.

**Aumentamos el rendimiento** mediante la orientación del grupo a los resultados y la responsabilidad compartida para la efectividad del equipo.

**Mejoramos la eficacia y fiabilidad** a través de estrategias, procesos y prácticas específicas.

Anderson. D., Augustine, S., Avery, C., Cockburn, A., Cohn, M., et al. 2005

## A.4 Métodos Ágiles

En la década de los noventa y a principios del 2000, se originó y obtuvo fuerza una serie de metodologías ágiles. Aunque difieren en una variedad de aspectos, lo que tienen en común se deriva de su apego al Manifiesto Ágil.

Los siguientes métodos ágiles se describen brevemente a continuación:

- **Lean Kanban**
- **Programación extrema** (*Extreme Programming*)
- **Métodos Crystal** (*Crystal Methods*)
- **Métodos de desarrollo de sistemas dinámicos** (*Dynamic Systems Development Methods*)
- **Desarrollo orientado a funcionalidades** (*Feature Driven Development*)
- **Desarrollo guiado por pruebas** (*Test Driven Development*)
- **Desarrollo adaptativo de software** (*Adaptive Software Development*)
- **Proceso unificado Ágil** (*Agile Unified Process*)
- **Desarrollo guiado por el dominio** (*Domain Driven Development*)

### A.4.1 Lean Kanban

El concepto de Lean optimiza el sistema de una organización para producir resultados valiosos sobre la base de sus recursos, necesidades y alternativas, mientras se reducen las pérdidas. Las pérdidas (*waste*) pudieran ser por la fabricación de algo equivocado, la imposibilidad de aprender o de prácticas que impidan el proceso. Debido a que estos factores tienen una naturaleza dinámica, una organización Lean evalúa la totalidad de su sistema y refina constantemente sus procesos. El fundamento de Lean es la reducción de la duración de cada ciclo (cada interacción), lo cual lleva a un aumento en la productividad mediante la reducción de retrasos y ayuda a detectar errores en las primeras etapas, reduciendo en consecuencia la cantidad total del trabajo necesario para finalizar una tarea. Los principios del software Lean se han implementado con éxito en el desarrollo de software.

*Kanban* literalmente significa “cartel” o “letrero”, e implica el uso de ayuda visual para dar seguimiento a la producción. El concepto fue introducido por Taiichi Ohno, considerado como el padre de los Sistemas de Producción Toyota (TPS, por sus siglas en inglés). El uso de ayuda visual es eficaz y se ha convertido en una práctica común. Algunos ejemplos incluyen: tarjetas de tarea, tableros de Scrum y gráficas de trabajo terminado (*Burndown Charts*). Dichos métodos generaron atención debido a su práctica en Toyota, empresa líder en gestión de procesos. Lean Kanban integra el uso de métodos de visualización según lo prescrito por Kanban aunado a los principios de Lean, creando así un sistema visual de gestión de proceso evolutivo incremental.

## A.4.2 Programación extrema

La programación extrema (conocida en inglés como: *Extreme Programming*), creada en la Chrysler Corporation, obtuvo impulso en la década de 1990. La programación extrema, o XP, por sus siglas en inglés, evita el aumento radical del costo de software cambiante con el paso del tiempo. Las características claves del XP incluyen el desarrollo incremental, horarios flexibles, códigos de prueba automatizados, comunicación verbal, diseño en evolución constante, así como la colaboración de cerca a corto y largo plazo se deriva de todos los que participan.

La programación extrema valora la comunicación, la retroalimentación, la simplicidad y el valor. Los distintos roles en el enfoque de XP incluyen: el cliente, desarrollador, seguidor (*tracker*) y el *coach*. Prescribe varias prácticas de codificación, de desarrollo y empresariales, así como eventos y artefactos para lograr un desarrollo eficaz y eficiente. El *extreme programming* ha sido adoptado extensamente debido a sus prácticas de ingeniería bien definidas.

## A.4.3 Métodos Crystal

Las metodologías Crystal para el desarrollo de software fueron introducidas por Alistair Cockburn a principios de la década de 1990. La intención de los métodos Crystal es centrarse en las personas; ser ligeros y fáciles de adaptar. Debido a que las personas son primordiales, el proceso de desarrollo y las herramientas no son fijas, sino que se ajustan a los requerimientos y características específicas del proyecto. Se utiliza el espectro de colores para decidir sobre la variante de un proyecto. Los factores tales como la comodidad, el dinero a discreción, dinero esencial y la vida, juegan un papel importante para determinar el “peso” de la metodología, lo cual se representa en varios colores del espectro. La familia Crystal se divide en: *Crystal Clear* (claro como el cristal), *Crystal Yellow* (cristal amarillo), *Crystal Orange* (cristal naranja), *Crystal Orange Web* (cristal naranja web), *Crystal Red* (cristal rojo), *Crystal Maroon* (cristal marrón), *Crystal Diamond* (cristal diamante) y *Crystal Sapphire* (cristal zafiro).

Todos los métodos Crystal tienen cuatro roles: patrocinador ejecutivo (*executive sponsor*), diseñador líder (*lead designer*), desarrolladores y usuarios experimentados. Los métodos Crystal recomiendan varias estrategias y técnicas para lograr agilidad. Un ciclo de proyecto Crystal consiste de la implementación del acta constitutiva (*chartering*), ciclo de entrega y cierre (*wrap-up*).

## A.4.4 Métodos de desarrollo de sistemas dinámicos

El marco del sistema de desarrollo de sistemas dinámicos (DSMS, por sus siglas en inglés) fue publicado inicialmente en 1995 y lo administra el Consorcio DSMS. El DSMS fija la calidad y el esfuerzo en términos de costo y tiempo al principio y ajusta los entregables del proyecto para cumplir con los criterios fijos mediante la priorización de los entregables en las categorías: “Debe tener” (*Must have*), “Debería tener” (*Should have*), “Podría tener” (*Could have*) y “No tendrá” (*Won't have*), con el uso de la técnica de

priorización MoSCoW. El DSMS es un método orientado en sistemas con seis fases distintas: pre-proyecto; viabilidad; fundamentos; exploración e ingeniería; desplazamiento y evaluación de beneficios.

En el 2007 se introdujo una nueva versión del sistema de desarrollo de sistemas dinámicos, conocida como DSMS Atern, misma que se enfoca tanto en la priorización de entregables como en el usuario consistente o colaboración del cliente. La nueva versión se inspira en Arctic Tern, haciéndola un marco de desarrollo de software centrado en el desarrollador para la entrega de características del proyecto a tiempo y dentro de los límites del presupuesto y en control de calidad.

#### **A.4.5 Desarrollo orientado a funcionalidades**

El desarrollo orientado en funcionalidades (FDD, por sus siglas en inglés) fue introducido por Jeff De Luca en 1997 y opera bajo el principio de concluir un proyecto mediante su fragmentación en pequeñas funciones valoradas por el cliente que puedan presentarse en menos de dos semanas. El desarrollo orientado en funcionalidades tiene dos principios fundamentales: el desarrollo de software es una actividad humana y es una funcionalidad valorada por el cliente.

El FDD define seis roles principales: Gerente del proyecto, arquitecto en jefe, gerente de desarrollo, programadores jefes, propietarios de clase y expertos del dominio, aunados a una serie de funciones complementarias. El proceso FDD es iterativo y consiste en el desarrollo de un modelo general; en crear una lista de características y después planificar, diseñar y crear con base en la característica.

#### **A.4.6 Desarrollo guiado por pruebas**

Conocido también como “desarrollo primero por pruebas” (del inglés: *Test-First Development*), el desarrollo guiado por pruebas fue introducido por Kent Beck, uno de los creadores de la programación extrema. El desarrollo guiado por pruebas es un método de desarrollo de software que implica redactar primero códigos de prueba automáticos y desarrollar la cantidad mínima de código necesario para avanzar después hacia la siguiente prueba. La totalidad del proyecto se divide en pequeñas características valoradas por el cliente que deben desarrollarse en el ciclo de desarrollo más breve posible. Las pruebas se redactan con base en los requerimientos y especificaciones del cliente. Las pruebas diseñadas en la etapa anterior se utilizan para diseñar y redactar el código de producción.

El desarrollo guiado por pruebas (TDD, por sus siglas en inglés), se puede clasificar en dos niveles: Aceptación de TDD (ATDD, en inglés), lo cual requiere de una prueba distinta de aceptación, y Desarrollador TDD (DTDD en inglés) que implica la redacción de una sola prueba de desarrollo. El TDD se ha popularizado debido a las numerosas ventajas que ofrece como los resultados confiables, la retroalimentación constante y la reducción de tiempo de depuración (*debugging*).

### A.4.7 Desarrollo adaptativo de software

El desarrollo adaptativo de software (ASD, por sus siglas en inglés) surgió a partir del rápido trabajo de desarrollo de aplicaciones por parte de Jim Highsmith y Sam Bayer. Los aspectos más destacados del ASD son la constante adaptación de procesos al trabajo con el que se cuenta, el suministro de soluciones a los problemas que surgen en los grandes proyectos, así como el desarrollo iterativo e incremental con prototipos continuos.

Al ser un método de desarrollo impulsado por el riesgo y tolerante al cambio, el desarrollo adaptativo de software indica que un plan no puede aceptar incertidumbres y riesgos, ya que esto sería indicativo de un plan deficiente y fallido. El desarrollo adaptativo de software se basa en características y se guía por metas. La primera fase en este tipo de desarrollo es la especulación (a diferencia de la planificación), seguida de las fases de colaboración y aprendizaje.

### A.4.8 Proceso unificado ágil

El proceso unificado ágil (AUP, por sus siglas en inglés) evolucionó del proceso unificado racional de IBM (del inglés: *IBM's Rational Unified Process*). El proceso unificado Ágil, desarrollado por Scott Ambler, combina técnicas ágiles probadas y examinadas por la industria tales como el desarrollo guiado por pruebas (TDD), modelos ágiles, gestión ágil de cambios y refactorización de base de datos, a fin de brindar un producto funcional de la mejor calidad.

El proceso unificado ágil modela sus procesos y técnicas con base en los valores de las herramientas de la simplicidad, agilidad, personalización, auto-organización e independencia y se enfoca en actividades de alto valor. Los principios y valores del proceso unificado ágil se ponen en acción en las fases de: incepción (inicio), elaboración, construcción y transición.

### A.4.9 Desarrollo guiado por el dominio

El desarrollo guiado por el dominio (DDD, por sus siglas en inglés) es un método de desarrollo ágil diseñado para administrar diseños complejos con la implementación vinculada a un modelo evolutivo. Fue conceptualizado en el 2004 por Eric Evans y gira en torno al diseño de un dominio central. La palabra “dominio” se define como un área de actividad en la cual el usuario aplica un programa o funcionalidad. Muchas de estas áreas se procesan en lotes y se diseña un modelo. El modelo consiste en un sistema de abstracciones que se pueden utilizar para diseñar un proyecto general y resolver los problemas relacionadas a los dominios en lote. Los valores centrales del DDD incluyen: el diseño orientado en el dominio, diseño guiado por el modelo, el lenguaje ubicuo y el contexto limitado.

En el DDD, se establece un lenguaje ubicuo y se modela el dominio. Después sigue el diseño, el desarrollo y la evaluación. La refinación y refactorización del modelo del dominio se lleva a cabo hasta que sea satisfactorio.

## APÉNDICE B. AUTORES Y REVISORES DE LA GUÍA SBOK™

Este apéndice enumera los nombres de las personas que han contribuido al desarrollo y la producción de la *Guía SBOK™*.

SCRUMstudy™ le agradece a todas estas personas por su apoyo continuo y reconoce sus contribuciones para el desarrollo de la *Guía de SBOK™*.

### B.1 Autor principal

Tridibesh Satpathy

### B.2 Los co-autores y expertos en la materia

R-A Alves

Winfried Hackmann

Quincy D. Jordan

Gaynell Malone

J. Drew Nations

Buddy Peacock

Karen Lyncook

Jaimie M. Rush

Elizabeth Lynne Warren

Ruth Kim

Mehul Doshi

Gaurav Garg

Ajey Grandhem

Sayan Guha

Vinay Jagannath

Deepak Ramaswamy

Ahmed Touseefullah Siddiqui

## B.3 Revisores y Equipo de Edición

Corey T. Bailey  
Sohini Banerjee  
Vince Belanger  
Bobbie Green  
Magaline D. Harvey  
Ravneet Kaur  
Robert Lamb  
Mimi LaRaque  
Melissa Lauro  
Richard Mather  
Lachlan McGurk  
Madhuresh Kumar Mishra  
Neha Mishra  
Yogaraj Mudalgi  
Jose Nunez  
Obi Nwaojigba  
Bryan Lee Perez  
James Pruitt  
Charles J. Quansah  
Frank Quinteros  
Nadra Rafee  
Tommie L. Sherrill  
Barbara Siefken  
Sandra A. Strech  
Frances Mary Jo Tessler  
Chrys Thorsen  
Mike Tomaszewski  
Ron Villmow

## REFERENCIAS

Anderson, D., Augustine, S., Avery, C., Cockburn, A., Cohn, M., DeCarlo, D., Fitzgerald, D., Highsmith, J., Jepsen, O., Lindstrom, L., Little, T., McDonald, K., Pixton, P., Smith, P., and Wysocki, R. (2005) "Declaration of Interdependence," accessed Septiembre 2013, <http://www.pmdoi.org/>.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001) "Manifiesto for Agile Software Development," obtenido el Septiembre 2013, <http://agilemanifiesto.org/>.

Fellers, G. (1994) *Why Things Go Wrong: Deming Philosophy In A Dozen Ten-Minute Sessions*. Gretna, LA: Pelican Publishing.

Greenleaf, R. K. (1977) *Servant Leadership: A Journey into the Nature of Legitimate Power and Greatness*. Mahwah, NJ: Paulist Press.

Kano, N., Seraku, N., Takahashi, F., and Tsuji, S. (1984) "Attractive Quality and Must Be Quality." *Quality*, 14 (2): 39–48.

Leffingwell, D. and Widrig, D. (2003) *Managing Software Requirements: A Use Case Approach, 2nd ed.* Boston: Addison-Wesley.

Maslow, A. H. (1943) "A Theory of Human Motivation." *Psychological Review*, 50 (4): 370–396.

McGregor, D. (1960) *The Human Side of Enterprise*. New York: McGraw-Hill.

Patton, J. (2005) "It's All in How You Slice." *Better Software*, Enero: 16–40.

Spears, L. C. (2010) "Character and Servant Leadership: Ten Characteristics of Effective, Caring Leaders." *The Journal of Virtues & Leadership*, 1 (1): 25–30.

Takeuchi, H. and Nonaka, I. (1986) "The New New Product Development Game." *Harvard Business Review*, January–February: 137–146.

## GLOSARIO

### **Método de los 100 puntos**

El método de los 100 puntos fue desarrollado en el 2003 por Dean Leffingwell y Don Widrig. Dicho método implica otorgar 100 puntos al cliente a fin de que los pueda utilizar para votar por las características que consideren más importantes.

### **Entregables aceptados (*Accepted Deliverables*)**

Los entregables que cumplen con los criterios de aceptación de las historias de usuario son aceptados por el propietario del producto. Se consideran entregables aceptados que pueden distribuirse al cliente si así lo desea.

### **Adaptación**

La adaptación sucede cuando el equipo principal de Scrum y el (los) socio(s) (*stakeholders*) aprende(n) por medio de la transparencia e inspección, adaptando después lo aprendido para mejorar el trabajo que realizan.

### **Estimación por afinidad (*Affinity Estimation*)**

La estimación por afinidad (del inglés: *Affinity Estimation*) es una técnica que se utiliza para estimar rápidamente un gran número de historias de usuarios con el uso de categorías. Dichas categorías pueden ser pequeñas, medianas o grandes, o bien, pueden estar enumeradas utilizando valores de punto de la historia (*point story values*) para indicar el tamaño relativo. Algunos de los beneficios claves de este método son que el proceso es muy transparente, visible para todos y fácil de llevar a cabo.

### **Mejoras accionables aceptadas (*Agreed Actionable Improvements*)**

Las mejoras accionables aceptadas (del inglés: *Agreed Actionable Improvements*) son los resultados primarios del proceso de retrospectiva del sprint. Forman parte de la lista de elementos accionables que ha elaborado el equipo para hacer frente a los problemas y mejorar los procesos a fin de mejorar también su desempeño en futuros sprints.

### **Aprobación, estimación y asignación de historias de usuarios**

En este proceso, el propietario del producto aprueba las historias de usuario para un sprint. Después, el Scrum Master y el equipo Scrum calculan el esfuerzo necesario para desarrollar la funcionalidad descrita en cada historia de usuario. Por último, el equipo Scrum se aboca a entregar los requerimientos del cliente en forma de historias de usuario autorizadas, estimadas y asignadas.

### **Solicitudes de cambio aprobadas**

Las solicitudes de cambios aprobadas son los cambios que han sido aprobados para incluirse en la lista priorizada de pendientes del producto. En ocasiones, las solicitudes de cambio aprobadas pueden presentarlos los gerentes del programa o de la cartera, y serían entradas que se añadirán a la lista de cambios autorizados del proyecto para su implementación en futuros sprints.

### **Aprobación, estimación y asignación de historias de usuarios**

Las historias de usuario (entradas en este proceso), tienen estimaciones de alto nivel de los procesos de creación de la lista priorizada de pendientes del producto y de creación de historias de usuario. Dichas estimaciones son utilizadas por el propietario del producto para aprobar las historias de usuario para el sprint. Una vez aprobadas, el equipo estima dicha historias utilizando diversas técnicas de evaluación. Después de dicha evaluación, el equipo se avoca a un subconjunto de historias de usuario aprobadas y estimadas que consideran que se pueden terminar en el siguiente sprint. Estas historias de usuario aprobadas, estimadas y asignadas formarán parte de la lista de pendientes del sprint.

### **Líder asertivo**

Los líderes asertivos confrontan los problemas y demuestran confianza para establecer autoridad con respeto.

### **Elementos de acción asignados y fechas de entrega**

Una vez que se han elaborado y refinado los elementos de acción asignados y las fechas límite, el equipo Scrum puede considerar los puntos de acción para implementar las mejoras. Cada elemento de acción contará con una fecha límite de conclusión.

### **Líder autocrático**

Los líderes autocráticos toman decisiones por su cuenta, dándoles poco o nada de tiempo a los miembros del equipo antes de tomar una decisión. Este estilo de liderazgo se debe utilizar solamente en raras ocasiones.

### **Herramientas automatizadas de software**

Las herramientas automatizadas de software son aquellas herramientas de software que se utilizan para la planificación, recopilación de información y distribución.

**Mejor coordinación del equipo**

La reunión de Scrum de Scrums facilita la coordinación del trabajo entre los varios equipos Scrum. Toma especial importancia cuando existen tareas que involucran dependencias entre equipos. Esto permite exponer rápidamente las incompatibilidades y discrepancias entre el trabajo y los entregables de los distintos equipos. Este foro brinda también a los equipos la oportunidad de mostrar sus logros y dar su retroalimentación a los demás equipos.

**Lluvia de ideas**

Son sesiones donde los socios y los miembros del equipo principal de Scrum comparten abiertamente ideas por medio de discusiones y sesiones de intercambio de conocimientos, generalmente dirigidas por un facilitador.

**Justificación del negocio**

La justificación del negocio demuestra las razones para emprender un proyecto. Responde a la pregunta: ¿Por qué es necesario este proyecto? La justificación del negocio guía todas las decisiones relacionadas a un proyecto.

**Necesidades del negocio**

Las necesidades del negocio son aquellos resultados del negocio que se espera que cumpla el proyecto, tal como se documenta en la declaración de visión del proyecto.

**Requerimientos del negocio**

Los requerimientos del negocio definen lo que debe ser entregado para cumplir con las necesidades del negocio y brindarles valor a los socios. La suma de toda la información obtenida mediante las distintas herramientas tales como entrevistas con el usuario o el cliente, cuestionarios, sesiones de diseño de aplicación conjunta (JAD, por sus siglas en inglés), análisis de brecha, análisis FODA y demás sesiones, ayudan a obtener una mejor perspectiva sobre los requerimientos empresariales y en la creación de la lista priorizada de pendientes del producto.

**Solicitud de cambio**

Las peticiones para realizar cambios generalmente se presentan como solicitudes de cambio. Dichas solicitudes permanecen como no aprobadas hasta que se autorizan formalmente.

**Jefe propietario del producto (*Chief Product Owner*)**

En los grandes proyectos, el jefe propietario del producto prepara y mantiene la lista priorizada de pendientes del producto para el proyecto. Él o ella coordinan el trabajo entre los propietarios del producto de los equipos Scrum. Los propietarios del producto, a su vez, se encargan de administrar sus respectivas partes en la lista priorizada de pendientes del producto.

**Jefe Scrum Master**

En los grandes proyectos, el jefe Scrum Master es responsable de moderar las reuniones de Scrum de Scrums (SoS) y eliminar los impedimentos que afectan a los varios equipos.

**Líder de apoyo y entrenamiento (*Coaching/Supportive Leader*)**

Los líderes de apoyo y entrenamiento dan instrucciones; después apoyan y monitorean a los miembros del equipo al escuchar, ayudar, alentar y presentar una perspectiva positiva en tiempos de incertidumbre.

**Colaboración**

En Scrum, la colaboración describe el trabajo conjunto del equipo principal de Scrum con los socios a fin de crear y validar los entregables del proyecto para cumplir con las metas trazadas en la visión del proyecto. La colaboración se produce cuando los equipos trabajan en conjunto para aprender de los demás y aprovechar ese conocimiento para producir algo mejor.

**Plan de colaboración**

La colaboración es un elemento de suma importancia en Scrum. El plan de colaboración delinea la forma en la que participan y colaboran entre sí las personas encargadas de la toma de decisiones, los socios y los miembros del equipo.

**Co-ubicación (*Collocation*)**

Co-ubicación (del inglés: *collocation*) es tener a todos los miembros del equipo principal de Scrum ubicados en el mismo lugar de trabajo, aprovechando las ventajas de una mejor coordinación, resolución de problemas, intercambio de conocimiento y aprendizaje.

**Plan de comunicación**

Este plan especifica los registros que se deben crear y mantenerse durante todo el proyecto. Se utiliza una variedad de métodos para transmitir a los socios información importante sobre el proyecto. El plan de comunicación define los métodos e indica quién es responsable de las distintas actividades de comunicación.

**Misión de la empresa**

La misión de la empresa brinda el marco para formular las estrategias de una empresa u organización que guía la toma de decisiones en general.

**Visión de la empresa**

Comprender la visión de la empresa ayuda a que el proyecto mantenga su enfoque en los objetivos de la organización y en el futuro potencial de la empresa. La visión de la empresa puede ayudar y guiar al propietario del producto para crear la declaración de visión del proyecto.

**Realizar reunión diaria de pie (*Conduct Daily Standup*)**

Es un proceso en donde se realiza diariamente una reunión muy enfocada con un bloque de tiempo asignado. A esta reunión se le conoce como reunión diaria de pie (*Daily Standup Meeting*), y es un foro que les permite a los miembros del equipo Scrum ponerse al día sobre sus progresos y sobre cualquier impedimento que pudieran estar enfrentando.

**Realizar planificación del lanzamiento**

En este proceso, el equipo principal de Scrum revisa las historias de usuario de alto nivel en la lista priorizada de pendientes del producto para desarrollar un programa de planificación del lanzamiento, que esencialmente es un programa de implementación por fases que se puede compartir con el (los) socio(s). Durante este proceso, también se determina la duración de los sprints.

**Gestión de conflictos**

Los miembros del equipo utilizan técnicas de gestión de conflictos para administrar cualquier conflicto que surja durante un proyecto Scrum. Las fuentes del conflicto generalmente incluyen los horarios, prioridades, recursos, la jerarquía de informes, problemas técnicos, procedimientos, personalidades y costos.

**Mejora continua**

La mejora continua es un enfoque de Scrum en el cual el equipo aprende de las experiencias y de la participación de los socios para mantener constantemente actualizada la lista priorizada de pendientes del producto con cualquier cambio en los requisitos.

**Justificación continua de valor**

La justificación continua de valor describe la evaluación con regularidad del valor del negocio para determinar si aún existe la justificación o viabilidad de la ejecución del proyecto.

**Convocar Scrum de Scrums**

En este proceso, el (los) socio(s) o los representantes del equipo Scrum convocan a reuniones de Scrum de Scrums en intervalos predeterminados o cuando sea necesario para colaborar y dar seguimiento a sus respectivos progresos, impedimentos y dependencias entre equipos.

**Creación de entregables**

La creación de entregables es el proceso en el cual el equipo Scrum trabaja en las tareas de la lista de pendientes del sprint para crear entregables del sprint.

**Creación de la lista priorizada de pendientes del producto**

En este proceso se elaboran y se refinan las épicas dándoles después prioridad para crear una lista priorizada de pendientes del producto para el proyecto. Durante este proceso se establecen también los criterios de terminado.

**Creación de la visión del proyecto**

En este proceso, se revisa el caso de negocio del proyecto para crear la declaración de visión del proyecto que servirá de inspiración y brindará un enfoque para la totalidad del proyecto. Durante este proceso se identifica al propietario del producto.

**Creación de la lista de pendientes del sprint**

En este proceso, el equipo principal de Scrum lleva a cabo reuniones de planificación del sprint donde el grupo crea una lista de pendientes del sprint y se incluyen todas las tareas a realizar durante el mismo.

**Creación de tareas**

En este proceso, las historias aprobadas, estimadas y asignadas se dividen en tareas específicas y se compilan en una lista de tareas. Para tales efectos, generalmente se lleva a cabo una reunión de planificación de tareas.

**Creación de historias de usuario**

En este proceso se crean las historias de usuario y sus respectivos criterios de aceptación. Las historias de usuario generalmente las redacta el propietario del producto y están diseñadas para garantizar que los requerimientos del cliente estén claramente representados y puedan ser plenamente comprendidos por todos los socios.

## Diagrama de flujo acumulativo (DFA)

El diagrama de flujo acumulativo (DFA) es una herramienta útil para la elaboración de informes y el seguimiento de los resultados del proyecto. Proporciona una representación sencilla y visual del progreso del proyecto en un punto de tiempo determinado. Se utiliza generalmente para proporcionar un estado de mayor nivel de la totalidad del proyecto y no para actualizaciones individuales diarias de Sprints.

## Cliente

El cliente es un individuo u organización que adquiere el producto del proyecto, servicio u otro resultado. En cualquier organización, dependiendo del proyecto, puede haber clientes internos (dentro de la misma organización) o clientes externos (fuera de la organización).

## Priorización basada en el valor que se entrega al cliente

La priorización basada en el valor que se entrega al cliente le da importancia primordial al cliente y se esfuerza primero en implementar las historias de usuario con más alto valor. Dichas historias de usuario de alto valor se identifican y se colocan en la parte superior de la lista priorizada de pendientes del producto.

## Reunión diaria de pie (*Daily Standup Meeting*)

La reunión diaria de pie es una breve reunión diaria. Tiene un bloque de tiempo asignado de 15 minutos. Los miembros del equipo se reúnen para dar un informe sobre sus progresos y contestar las siguientes tres preguntas:

1. ¿Qué terminé ayer?
2. ¿Qué terminaré hoy?
3. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

## Segmentación (*Decomposition*)

La segmentación (del inglés: *decomposition*) es una herramienta donde las tareas de alto nivel se dividen en niveles inferiores y más detallados. Los miembros del equipo Scrum segmentan las historias de usuario en tareas. Las historias de usuario de la lista priorizada de pendientes del producto deben estar suficientemente segmentadas a un nivel que le brinde al equipo Scrum la información adecuada para crear entregables a partir de las tareas mencionadas en la lista de tareas.

## Líder delegador

Los líderes delegadores participan en la mayoría de la toma de decisiones; sin embargo, delegan parte de las responsabilidades de planificación a los miembros del equipo, particularmente si son competentes para manejar tareas. Este estilo de liderazgo es apropiado en situaciones en las que el líder está en sintonía con los detalles específicos de proyectos y cuando el tiempo es limitado.

**Demostración y validación del sprint**

En este proceso, el equipo Scrum demuestra los entregables del Sprint al propietario del producto y a los socios relevantes durante una reunión de revisión del sprint.

**Determinación de dependencia**

Una vez que el equipo Scrum ha seleccionado las historias de usuario para un determinado sprint, deben considerar las dependencias, incluyendo aquellas relacionadas a la disponibilidad de personal, así como cualquier dependencia técnica. Documentar adecuadamente las dependencias ayuda a los equipos Scrum a determinar el orden relativo en el cual deben ejecutarse las tareas para crear los entregables del sprint. Las dependencias destacan también la relación e interacción entre las tareas dentro del equipo Scrum que trabaja en un determinado Sprint y con otros equipos Scrum en el proyecto.

**Patrones de diseño**

Los patrones de diseño proporcionan una manera formal de registrar una resolución a un problema de diseño en un campo específico de especialización. Dichos patrones registran tanto el proceso que se utiliza como la resolución, misma que puede reutilizarse después para mejorar la toma de decisiones y la productividad.

**Desarrollo de épicas**

En este proceso, la declaración de visión del proyecto funge como la base para el desarrollo de épicas. Se pueden llevar a cabo reuniones de grupos de usuario para el desarrollo de épicas.

**Contrato de desarrollo en fases**

Este contrato facilita la disponibilidad de fondos cada mes o cada trimestre después de concluir satisfactoriamente un lanzamiento. Incentiva tanto al cliente como al proveedor y garantiza que el riesgo monetario del cliente esté limitado a un periodo específico, ya que los lanzamientos fracasados no reciben financiamiento.

**Líder directivo**

Los líderes directivos instruyen a los miembros del equipo sobre las tareas que se requieren y sobre cómo y cuándo deben llevarse a cabo.

## Dependencias discrecionales

Las dependencias discrecionales son aquellas que se colocan en el flujo de trabajo por decisión propia. Normalmente, el equipo Scrum determina cuáles son las dependencias discrecionales con base en experiencias anteriores o las mejores prácticas en un campo o dominio específico.

## Criterios de terminado

Los criterios de terminado son el conjunto de reglas que aplican a todas las historias de usuario. Una definición clara del “terminado” es crítica, ya que elimina la ambigüedad de los requisitos y ayuda a que el equipo se apegue a las normas obligatorias de calidad. Esta clara definición se utiliza para crear los criterios de terminado (una de las salidas del proceso de Creación de la lista priorizada de pendientes del producto). Una historia de usuario se considera terminada cuando se demuestra al propietario del producto para su aprobación; después la califica con base en los criterios de terminado y los criterios de aceptación de la historia de usuario.

## Análisis del valor ganado (*Earned Value Analysis*)

El análisis del valor ganado analiza el verdadero desempeño del proyecto en comparación al rendimiento previsto en un punto previsto. Mide las variaciones actuales en el cronograma del proyecto, así como los costos de desempeño y prevé el costo final con base en el actual desempeño determinado.

## Lista de tareas de esfuerzo estimado

La lista de tareas del esfuerzo estimado es una lista de tareas relacionadas a las historias de usuario que se incluyen en un sprint. El esfuerzo estimado se expresa en términos de los criterios de estimación acordados por el equipo. El equipo Scrum utiliza lista de tareas del esfuerzo estimado durante las reuniones de planificación del sprint a fin de crear la lista del sprint y la gráfica de trabajo pendiente (*Burndown Chart*) del sprint.

## Control del proceso empírico

Un modelo de control de proceso empírico ayuda a tomar decisiones basadas en la observación y en la experimentación en vez de la planificación inicial detallada. Se base en las tres principales ideas de transparencia, inspección y adaptación.

## Épica(s)

Las épicas se escriben en las etapas iniciales del proyecto cuando la mayoría de las historias de usuario son funcionalidades de alto nivel o descripciones del producto y cuando se definen ampliamente los requerimientos y descripciones del mismo. Son historias de usuario amplias y sin refinar que se incluyen en la lista priorizada de pendientes del producto.

## **Rango de estimación**

Las estimaciones de los proyectos deben presentarse en rangos. Las cifras exactas pueden dar la impresión de ser muy específicas cuando en realidad tal vez no lo sean. De hecho, se entiende que las estimaciones, por su definición, no son precisamente exactas. Los rangos de estimación deben basarse en el nivel de confianza del equipo en cada estimación que se haga.

## **Proceso de estimación de tareas**

En este proceso, el equipo principal de Scrum, en un taller de estimación de tareas, calcula el esfuerzo necesario para lograr cada tarea en la lista. El resultado de este proceso es una lista de tareas de esfuerzo estimado.

## **Criterios de estimación**

El objetivo principal del uso de los criterios de estimación es mantener los tamaños relativos de estimación y minimizar la necesidad de volver a realizar el cálculo. Los criterios de estimación se pueden expresar de muchas formas. Dos ejemplos comunes son: puntos de historia y tiempo ideal.

## **Valor monetario esperado (*Expected Monetary Value*)**

Es una técnica de evaluación de riesgos donde el impacto financiero potencial de un riesgo se determina con base en su valor monetario esperado (VME). Dicho valor se calcula multiplicando el impacto monetario por la probabilidad de riesgo, según la aproximación del cliente.

## **Explorador—Comprador—Vacacionista—Prisionero (ECVP)**

Es un ejercicio que se puede llevar a cabo al inicio de la reunión de retrospectiva del Sprint para entender la mentalidad de los participantes y establecer el tono de la reunión. Se les pide a los asistentes que indiquen de forma anónima lo que mejor representa su punto de vista en la reunión.

## **Dependencias externas**

Las dependencias externas son aquellas que están relacionadas a las tareas, actividades o productos fuera del alcance del trabajo a realizar por el equipo Scrum, pero que son necesarias para concluir una tarea del proyecto o crear un entregable del proyecto. Las dependencias externas generalmente están fuera del alcance del equipo Scrum.

## **Puño de cinco**

El puño de cinco es un mecanismo simple y rápido para lograr el consenso en un grupo y guiar una discusión. Después del debate inicial sobre una determinada propuesta o decisión pendiente, se les pide a los miembros del equipo Scrum que voten en una escala del 1 al 5 utilizando sus dedos.

## **Reuniones de grupo de enfoque**

Los grupos de enfoque reúnen personas en una sesión guiada para brindar sus opiniones, percepciones o valoraciones sobre un producto, servicio o resultado esperado. Los miembros del grupo de enfoque tienen la libertad de hacerse preguntas unos a otros y obtener clarificaciones sobre temas o conceptos específicos. Por medio de este cuestionamiento, la crítica constructiva y la retroalimentación, los grupos de enfoque llevan a un producto de mejor calidad, contribuyendo al cumplimiento de las expectativas de los usuarios.

## **Formación de un equipo Scrum**

Durante este proceso se identifican a los miembros del equipo Scrum. Normalmente, el propietario del producto es el responsable principal de seleccionar a los miembros del equipo, aunque generalmente lo lleva a cabo en colaboración con el Scrum Master.

## **Etapas de formación**

La etapa de formación es la primera etapa en la formación de un equipo. Generalmente se considera una etapa divertida, ya que todo es novedoso y no se han encontrado dificultades con el proyecto.

## **Cuatro preguntas por equipo**

Conjunto de preguntas formuladas en cada reunión de Scrum de Scrums (SoS). Cada representante del equipo Scrum proporciona actualizaciones de su equipo, que generalmente se presentan en forma de respuesta a cuatro preguntas específicas:

1. ¿En qué ha trabajado mi equipo desde la última reunión?
2. ¿Qué va a hacer mi equipo hasta la próxima reunión?
3. ¿Qué esperaban los demás equipos que hiciera el nuestro y que no se ha hecho?
4. ¿Qué planes tiene hacer nuestro equipo que pudieran afectar a otros equipos?

## **Análisis de brecha**

El análisis de brecha es una técnica que se utiliza para comparar el estado actual con el estado deseado y determinar la forma de acortar la brecha entre ambos.

**Mantenimiento de la lista priorizada de pendientes del producto**

El mantenimiento de la lista priorizada de pendientes del producto es un proceso en el cual se mantiene y actualiza constantemente la lista priorizada de pendientes del producto.

**Proceso de identificación del Scrum Master y los socios**

En este proceso se identifican el Scrum Master y los socios mediante el uso de criterios de selección específicos.

**Impedimento**

Un impedimento es cualquier obstáculo o barrera que reduce la productividad del equipo Scrum.

**Fase de implementación**

La fase de implementación incluye los procesos relacionados con la ejecución de las tareas y actividades para crear un producto de un proyecto.

**Contrato de incentivos y sanciones**

Este contrato se basa en el acuerdo de que el proveedor será recompensado con un incentivo económico si los productos del proyecto se entregan a tiempo, pero incurrirá en sanciones económicas si la entrega se realiza tarde.

**Contrato de entrega incremental**

Este contrato incluye puntos de inspección en intervalos frecuentes. Ayuda a que el cliente o los socios (*stakeholders*) tomen decisiones periódicas sobre el desarrollo a lo largo de un proyecto en cada punto de inspección. El cliente puede aceptar el desarrollo del producto, optar por detener su desarrollo o solicitar modificaciones al mismo.

**Fichas (*Index Cards*)**

Las fichas, conocidas en inglés como *index cards*, se utilizan para dar seguimiento a la historias de usuario durante todo el proyecto. Esto aumenta la visibilidad y la transparencia, facilitando la detención oportuna de cualquier problema que pueda surgir.

## **Fase inicial**

Esta fase se compone de los procesos relacionados a la iniciación de un proyecto: Creación de la visión del producto, identificación del Scrum Master y socios, formación del equipo Scrum, desarrollo de épica(s), creación de la lista priorizada de pendientes del producto y realizar la planificación del lanzamiento.

## **Inspección**

La inspección es el monitoreo necesario para seguir el control del proceso empírico para garantizar que los entregables del proyecto se ajusten a los requisitos.

## **Dependencias internas**

Las dependencias internas son aquellas entre las tareas, productos o actividades que están bajo el control del equipo Scrum y dentro de un enfoque de trabajo a ser ejecutado por dicho equipo.

## **Tasa interna de retorno (TIR)**

La tasa interna de retorno (TIR), es una tasa de descuento sobre una inversión en la cual el actual valor de los flujos de efectivo se iguala al valor actual del flujo de salidas de efectivo a fin de evaluar la tasa de rentabilidad del proyecto. Al hacer un comparativo de los proyectos, generalmente resulta mejor el que cuenta con una elevada tasa interna de retorno.

## **Problemas**

Los problemas (en inglés: *issues*) generalmente son certezas bien definidas que actualmente se están sucediendo en el proyecto, por lo que no hay necesidad de realizar una evaluación de probabilidad como se hace con un riesgo.

## **Desarrollo iterativo**

El desarrollo iterativo es la entrega gradual de valor al cliente.

## **Sesiones JAD (*Joint Application Design*)**

Una sesión de diseño conjunto de aplicaciones (JAD, por sus siglas en inglés) es una técnica de recopilación de requisitos. Es un taller altamente estructurado que acelera el proceso de creación de la visión del proyecto, ya que permite a los socios y demás encargados de la toma de decisiones llegar a un consenso en cuanto el enfoque, los objetivos y demás especificaciones del proyecto.

**Contrato de empresa conjunta (*Joint Venture Contract*)**

Este tipo de contratos se utiliza generalmente cuando dos o más partes se asocian para realizar el trabajo de un proyecto. Ambas partes involucradas en el proyecto lograrán algún retorno sobre la inversión, ya que los ingresos o beneficios que sean generados, se habrán de compartir entre las partes.

**Análisis de Kano**

El análisis de Kano fue desarrollado en 1984 por Noriaki Kano y consiste en clasificar las características o requisitos en cuatro categorías con base en las preferencias del cliente:

1. Calidad atractiva (*Delighters*)
2. Calidad unidimensional (*Satisfiers*)
3. Calidad requerida (*Dissatisfiers*)
4. Calidad indiferente (*Indifferent*)

**Líder *Laissez Faire***

Es un estilo de liderazgo en el que el equipo se queda sin supervisión y donde el líder no interfiere con las actividades laborales diarias. Esto a menudo conduce a un estado de anarquía.

**Duración del sprint**

Basado en las varias entradas, incluyendo los requerimientos del negocio y la planificación del lanzamiento, el propietario del producto y el equipo Scrum deciden sobre la duración de los sprints para el proyecto. Una vez determinada, la duración del Sprint generalmente permanece fija para todo el proyecto.

La duración del Sprint es la duración de los sprints determinados para un proyecto.

**Riesgos**

Los riesgos incluyen cualquier evento incierto o no planificado que pudiera afectar positiva o negativamente el proyecto.

**Dependencias obligatorias**

Estas dependencias son tanto inherentes en la naturaleza del trabajo (como una limitación física) como pueden darse debido a obligaciones contractuales o por requerimientos legales.

## Estudio de mercado

El estudio de mercado es el proceso organizado de investigar, recopilar, cotejar y analizar la información relacionada a las preferencias del cliente respecto a los productos. Generalmente incluye extensos datos sobre tendencias del mercado, segmentación del mercado y procesos de comercialización.

## Criterios mínimos de aceptación

Los criterios mínimos de aceptación son declarados por la unidad empresarial. Después se convierten en parte de los criterios de aceptación para cualquier historia de usuario para dicha unidad empresarial. Cualquier funcionalidad definida por la unidad empresarial debe satisfacer dichos criterios mínimos de aceptación si busca ser aceptada por el propietario del producto.

## Riesgos mitigados

Los riesgos mitigados son aquellos que se tratan o mitigan con éxito por el equipo Scrum durante el proyecto.

## Dinero de Monopoly

El dinero de Monopoly es una técnica que consiste en darle al cliente dinero del juego Monopoly o "dinero falso" equivalente a la cantidad del presupuesto del proyecto, solicitando que lo distribuyan entre las historias de usuario que están a consideración. De esta forma, el cliente prioriza con base en lo que está dispuesto a pagar por cada historia de usuario.

## Priorización MoSCoW

El esquema de priorización MoSCoW obtiene su nombre de la versión en inglés de las frases: "Debe tener" (*Must have*), "Debería tener" (*Should have*), "Podría tener" (*Could have*) y "No tendrá" (*Won't have*). Las etiquetas están en orden de prioridad decreciente con historias de usuario con características de "Debería tener", siendo aquellas sin las que el producto no tendrá valor, e historias de usuarios con características de "gustaría que tuviera", siendo aquellas que, a pesar de que sería bueno tener, no se es necesario incluir.

## Valor presente neto (VPN)

El valor presente neto (VPN) es un método que se utiliza para determinar el valor actual neto de un futuro beneficio económico, considerando la inflación prevista o la tasa de interés.

**Rol no central**

Los roles no centrales son los que no son obligatoriamente necesarios para el proyecto Scrum. Estos pueden incluir a miembros de los equipos que estén interesados en el proyecto, que no tienen ningún papel formal en el equipo del proyecto, y pueden interactuar con el equipo, pero pueden no ser responsables del éxito del proyecto.

**Etapas de normalización**

La tercera etapa de la formación del equipo es cuando el equipo comienza a madurar, resolver sus diferencias internas y encontrar soluciones para trabajar juntos. Se considera un período de ajuste.

**Número de historias**

El término número de historias se refiere a la cantidad de historias de usuario que se entregan como parte de un solo sprint. Se puede expresar en términos de conteo simple o conteo ponderado.

**Oportunidades**

A los riesgos con posibilidades de causar un impacto positivo en el proyecto se les conoce como oportunidades.

**Costo de oportunidad**

El costo de oportunidad es el valor de la siguiente mejor opción de negocio o proyecto que fue descartada en favor del proyecto seleccionado.

**Métodos de desplazamiento organizacional**

Los mecanismos de desplazamiento de cada organización tienden a ser diferentes basados en su respectiva industria, los usuarios meta y el posicionamiento. Dependiendo del producto a entregarse, el desplazamiento puede ser remoto o puede incluir el envío físico o la transición de un artículo.

**Matriz de recurso organizacional (*Organizational Resource Matrix*)**

La matriz de recurso organizacional es una representación jerárquica de una combinación de una estructura organizacional funcional y una estructura organizacional del proyecto. Las organizacionales matrices reúnen a los miembros del equipo para un proyecto de distintos departamentos funcionales, tales como: tecnologías de la información, finanzas, comercialización, ventas y demás departamentos a fin de crear equipos interfuncionales (*cross-functional*).

## Comparación por pares

La comparación por pares es una técnica donde se prepara una lista de todas las historias de usuario en la lista priorizada de pendientes del producto. Después, cada historia de usuario se toma en forma individual y se compara con otras historias en la lista, una a la vez. Cada vez que se comparan dos historias de usuario, se toma una decisión en cuanto a cuál de las dos es más importante. Por medio de este proceso, se puede generar una lista priorizada de las historias de usuario.

## Análisis de Pareto

Esta técnica de evaluación de riesgos implica la clasificación de riesgos por magnitud. Ayuda al equipo Scrum a atender los riesgos por orden de impacto probable en un proyecto.

## Ciclo PDCA/PDSA

El ciclo de planificar, hacer, verificar y actuar (PDCA, por sus siglas en inglés), conocido también como Ciclo Deming o Shewhart, fue desarrollado por el doctor W. Edwards Deming, considerado como el padre del control de calidad moderno, y por el doctor Walter A. Shewhart. Deming modificó después el nombre de “planificar, hacer, verificar y actuar” a “planificar, hacer, estudiar y actuar” (PDSA, por sus siglas en inglés) ya que consideraba que el término “estudiar” hacía énfasis en el análisis en vez de simplemente inspección, como lo implica el término “verificar”. Tanto Scrum como el Ciclo Deming/Shewhart/PDCA son métodos iterativos enfocados en la mejora continua.

## Etapa de desempeño

Es la etapa final de la formación del equipo cuando este se encuentra más unido y opera a su nivel más alto en términos de rendimiento. Los miembros se han convertido en un equipo eficiente de profesionales que son consistentemente productivos.

## Prototipos (*Personas*)

Los prototipos de usuario (conocidos en inglés como *personas*) son personajes ficticios altamente detallados que representan a la mayoría de los usuarios, así como a otros socios que pudieran no utilizar directamente el producto al final. Los prototipos se crean para identificar las necesidades de la base de usuarios meta.

## Plan de pilotaje (*Piloting Plan*)

El plan pilotaje se puede utilizar para trazar un desplazamiento piloto a detalle. El alcance y los objetivos del desplazamiento, la base de usuarios seleccionada para la implementación, un cronograma del desplazamiento, los planes de transición, la preparación requerida del usuario, los criterios de evaluación para el desplazamiento y otros elementos clave relacionados al desplazamiento, se identifican en el plan de pilotaje y se comparten con los socios.

## **Fase de planificación y estimación**

La fase de planificación y estimación se compone de los procesos relacionados a la planificación y estimación de tareas, que incluyen: la creación de historias de usuario, aprobación, estimación y asignación de historias de usuario, creación de tareas, estimación de tareas y creación de la lista de pendientes del sprint.

### **Planificar para el valor (*Planning for Value*)**

La planificación para el valor es la justificación y confirmación del valor del proyecto. La responsabilidad de determinar cómo se crea valor recae en los socios (patrocinadores, clientes y/o usuarios), mientras que el equipo Scrum se concentra en lo que se habrá de desarrollar.

### **Póker de planificación**

El póker de planificación, conocido también como póker de estimación, es una técnica de estimación que equilibra el pensamiento del grupo y el pensamiento individual para estimar los tamaños relativos de las historias de usuario o el esfuerzo necesario para desarrollarlos.

### **Estimación por puntos de costo (*Points for Cost Estimating*)**

La estimación del costo se puede lograr mediante el uso de unidades relativas (por ejemplo: estimaciones de esfuerzo) en vez de unidades absolutas (costos reales incurridos). A fin de estimar el costo para implementar una historia de usuario, el equipo Scrum puede utilizar puntos de historia. Al hacerlo, el costo estimado para cada tarea sea en forma de puntos de historia en vez de unidades monetarias.

### **Cartera (*Portfolio*)**

Una cartera, conocida en inglés como *portfolio*, es un grupo de programas relacionados con el objetivo de entregar resultados del negocio según se define en la declaración de visión de la cartera. La lista priorizada de pendientes de la cartera incorpora la lista priorizada de pendientes del programa para todos los programas en la cartera.

### **Propietario de la cartera del producto (*Portfolio Product Owner*)**

El propietario de la cartera del producto (del inglés: *Portfolio Product Owner*) define los objetivos y prioridades estratégicas para la cartera.

### **Scrum Master de la cartera (*Portfolio Scrum Master*)**

El Scrum Master de la cartera (del inglés: *Portfolio Scrum Master*) resuelve problemas, elimina impedimentos, facilita y conduce las reuniones para la cartera.

## Priorización

La priorización puede definirse como la determinación del orden de las cosas y la separación de lo que se hará ahora de lo que se puede hacer después.

### Lista priorizada de pendientes del producto (*Prioritized Product Backlog*)

La lista priorizada de pendientes del producto (del inglés: *Prioritized Product Backlog*) es un solo documento de requisitos que define el ámbito del proyecto, proporcionando una lista de priorizada de las características del producto o servicio que habrá de entregar el proyecto.

### Cuadrícula de probabilidad e impacto (*Probability Impact Grid*)

Es una cuadrícula donde se evalúan los riesgos para establecer la probabilidad de ocurrencia y del impacto potencia en los objetivos del proyecto. Generalmente, se asigna una clasificación numérica tanto para la probabilidad como para el impacto en forma independiente. Ambos valores se multiplican para obtener una puntuación de la gravedad del riesgo, lo cual se puede utilizar para priorizar los riesgos.

### Árboles de probabilidad

Los eventos potenciales se representan en un diagrama con una rama para cada resultado posible de los acontecimientos. La probabilidad de cada resultado se indica en la rama apropiada, y estos valores se pueden utilizar para calcular el impacto general de la ocurrencia de riesgos en un proyecto.

## Producto

El término “producto” en la *Guía SBOK™*, puede referirse a un producto, servicio o cualquier otro entregable que brinde valor al cliente.

### Reunión de revisión de la lista priorizada de pendientes del producto

Una reunión de revisión de la lista priorizada de pendientes del producto (también conocida como Sesión de mantenimiento de la lista priorizada de pendientes del producto, conocida en inglés como: *Prioritized Product Backlog Grooming Session*), es una reunión formal que ayuda al equipo Scrum a revisar y a obtener un consenso sobre dicha lista.

### Propietario del producto (*Product Owner*)

El propietario del producto es la persona responsable de maximizar el valor del negocio para el proyecto. Es la persona responsable de articular los requerimientos del cliente y mantener la justificación del negocio para el proyecto.

**Programa**

Un programa es un grupo de proyectos relacionados con el objetivo de entregar resultados de un negocio, según se define en la declaración de visión del proyecto. La lista priorizada de pendientes del proyecto incorpora las listas priorizadas de pendiente del producto para todos los programas en el proyecto.

**Riesgos del programa y la cartera (*Program and Portfolio Risks*)**

Riesgos relacionados a una cartera o a un programa que también habrán de afectar los proyectos que forman parte de la respectiva cartera o programa.

**Propietario del producto del programa (*Program Product Owner*)**

El propietario del producto del programa define los objetivos estratégicos y las prioridades del programa.

**Scrum Master del programa (*Program Scrum Master*)**

El Scrum Master del programa resuelve problemas, elimina impedimentos, facilita y lleva a cabo reuniones para el programa.

**Proyecto**

Un proyecto es un negocio colaborativo para crear, ya sea nuevos proyectos o servicios, o para entregar resultados según se define en la declaración de visión del proyecto. Los proyectos generalmente se ven afectados por limitaciones en el tiempo, costos, enfoque, calidad, personas y capacidades organizacionales.

**Beneficios del proyecto**

Los beneficios del proyecto incluyen todas las mejores cuantificables de un producto, servicio o resultado que pudiera brindar durante la conclusión satisfactoria de un proyecto.

**Presupuesto del proyecto**

El presupuesto del proyecto es un documento financiero que incluye los costos del personal, materiales y demás gastos relacionados a un proyecto. El presupuesto del proyecto típicamente se autoriza por los patrocinadores a fin de garantizar que existen suficientes fondos disponibles.

**Acta constitutiva del proyecto (*Project Charter*)**

Un acta constitutiva de un proyecto es el documento oficial de los objetivos deseados y los resultados que se buscan en un proyecto. En muchas organizaciones, el acta constitutiva del proyecto es el documento que autoriza oficial y formalmente el proyecto, otorgando al equipo la autorización por escrito para dar inicio a los trabajos del proyecto.

## **Costos del proyecto**

Los costos del proyecto son las inversiones y demás costos de desarrollo en un proyecto.

## **Razonamiento del proyecto**

El razonamiento del proyecto incluye todos los factores que este requiere, ya sean positivos, negativos, elegidos o no (por ejemplo: capacidad inadecuada para cumplir con la demanda actual y la demanda prevista, la disminución en la satisfacción del cliente, baja utilidad, requerimientos legales, etc.).

## **Escalas de tiempo del proyecto (*Project Timescales*)**

Las escalas de tiempo reflejan la duración de un proyecto. Dichas escalas relacionadas a un caso de negocio incluyen también el tiempo durante el cual se obtendrán los beneficios del proyecto.

## **Reunión de visión del proyecto**

Una reunión de visión del proyecto es una reunión con el (los) socio(s), el propietario del producto del programa, el Scrum Master del programa y el jefe propietario del producto. Ayuda a identificar el contexto empresarial, los requerimientos del negocio y las expectativas de los socios a fin de desarrollar una declaración eficaz de la visión del proyecto.

## **Declaración de la visión del proyecto**

El resultado clave del proceso de creación de la visión del proyecto es una declaración bien estructurada de la declaración de la visión del proyecto. Una buena visión del proyecto explica las necesidades empresariales, así como lo que el proyecto busca cumplir en vez de cómo habrá de satisfacer la necesidad.

## **Elementos no funcionales propuestos para la lista priorizada de pendientes del producto (*Proposed Non-Functional Items for Product Backlog*)**

Los requerimientos no funcionales pudieran no estar completamente definidos en las primeras etapas del proyecto y pudieran surgir durante la revisión del sprint o en las reuniones de retrospectiva del sprint. Estos elementos deben añadirse a la lista priorizada de pendientes del producto a medida que se descubren.

## **Calidad**

La calidad se define como la capacidad que tiene el producto terminado como los entregables de cumplir con los criterios de aceptación y lograr el valor del negocio que espera el cliente.

## Garantía de calidad

La garantía de calidad es el proceso de evaluación y los estándares que rigen la gestión de calidad en un proyecto a fin de garantizar que continúen siendo relevantes. Las actividades relacionadas a la garantía de calidad se llevan a cabo como parte del trabajo.

## Control de calidad

El control de calidad es la ejecución de las actividades de calidad planeadas por el equipo Scrum en el proceso de creación de entregables que la potencialidad de enviarse. Incluye también el aprendizaje de cada serie de actividades realizadas a fin de lograr una mejora continua.

## Gestión de calidad

En Scrum, la gestión de calidad permite a los clientes conocer cualquier problema en forma anticipada en un proyecto. Ayuda a reconocer si un proyecto habrá de funcionarles o no. La gestión de calidad en Scrum se facilita mediante tres actividades interrelacionadas:

1. Planificación de calidad
2. Control de calidad
3. Garantía de calidad

## Planificación de calidad

La planificación de calidad es la identificación y definición del proyecto requerido de un Sprint y del proyecto, así como los criterios de aceptación, cualquier método de desarrollo a seguir y las responsabilidades clave de los miembros del equipo Scrum en relación a la calidad.

## Refactorización (*Refactoring*)

La refactorización es una herramienta específica para proyectos de software. El objetivo de esta técnica es mejorar el mantenimiento del código existente y hacerlo más sencillo, más conciso y más flexible. Refactorizar significa mejorar el diseño del código actual sin cambiar el comportamiento del mismo. Implica lo siguiente:

- La eliminación de código repetitivo y redundante
- Separar los métodos y las funciones en rutinas más pequeñas
- Definir claramente las variables y los nombres de los métodos
- Simplificar el diseño del código
- Hacer que el código sea más fácil de entender y modificar

**Entregables rechazados**

Los entregables rechazados son aquellos que no cumplen con los criterios de aceptación definidos. Después de cada reunión de revisión del sprint, se actualiza y se mantiene una lista de entregables rechazados incluyendo los entregables que no fueron aceptados.

**Clasificación relativa de priorización (*Relative Prioritization Ranking*)**

La clasificación relativa de priorización es una simple enumeración de las historias de usuario en el orden de prioridad. Es un método eficaz para determinar las historias deseadas para cada iteración o lanzamiento del producto o servicio.

**Tamaño relativo/Puntos de historia (*Relative Sizing/Story Points*)**

Además de utilizarse para la estimación de costos, los puntos de historia (*Story Points*), pueden emplearse también para calcular el tamaño total de una historia de usuario o de una característica. Este método asigna un punto de historia con base en una evaluación general del tamaño de la historia de usuario tomando en cuenta el riesgo, la cantidad requerida de esfuerzos y el nivel de complejidad.

**Contenido del lanzamiento**

Consiste en la información esencial sobre los entregables que puede ayudar al equipo de atención al cliente.

**Notas del lanzamiento (*Release Notes*)**

Las notas del lanzamiento deben incluir criterios de envío externos o de cara al mercado para el producto a entregarse.

**Cronograma de planificación del lanzamiento (*Release Planning Schedule*)**

Un cronograma de planificación del lanzamiento es una de las salidas clave del proceso para realizar la planificación del lanzamiento. Un cronograma de planificación del lanzamiento indica cuáles entregables van a ser puestos a disposición del cliente, junto con los intervalos planificados y las fechas para los lanzamientos. Podría no haber un cronograma de lanzamiento al final de cada iteración del sprint.

**Sesiones de planificación del lanzamiento (*Release Planning Sessions*)**

El principal objetivo de las sesiones de planificación del lanzamiento es crear un cronograma de planificación del lanzamiento y permitir al equipo Scrum contar con una descripción de los lanzamientos y del programa de entrega para los productos que desarrollan a fin de que se puedan alinear con las expectativas del propietario del producto y del (los) socio(s) relevante(s).

**Métodos de priorización del lanzamiento (*Release Prioritization Methods*)**

Los métodos de priorización del lanzamiento se utilizan para desarrollar el plan de lanzamiento. Dichos métodos son específicos en la industria y en la organización y generalmente se determinan por la alta gerencia en una organización.

**Problemas resueltos**

En las reuniones de Scrum de Scrums, los miembros del equipo Scrum tienen la oportunidad de discutir con transparencia los problemas que tienen un impacto en el proyecto. Esta discusión y resolución oportuna de los problemas en las reuniones de Scrum de Scrums mejora en buena medida la coordinación entre los distintos equipos de Scrum y reduce también la necesidad de rediseñar o volver a realizar el trabajo.

**Retrospectiva del proyecto**

En este proceso, que concluye el proyecto, los socios de la organización y los miembros del equipo principal de Scrum se reúnen para hacer una retrospectiva del proyecto e identificar, documentar e internalizar las lecciones aprendidas. A menudo, estas lecciones llevan a la documentación de mejoras accionables aceptadas, que se aplicarán en futuros proyectos.

**Reunión de retrospectiva del proyecto**

La reunión de retrospectiva del proyecto es una reunión para determinar las formas en la que se puede mejorar la colaboración del equipo y la eficacia en futuros proyectos. También se analizan las oportunidades positivas, negativas y potenciales. Esta reunión no tiene un bloque de tiempo asignado y se puede realizar en persona en forma virtual.

**Retrospectiva del Sprint**

En este proceso, el Scrum Master y el equipo Scrum se reúnen para analizar las lecciones aprendidas a lo largo del Sprint. Dichas lecciones se documentan y se pueden aplicar a futuros sprints.

**Registro de la retrospectiva del sprint**

Los registros de la retrospectiva del sprint son registros de las opiniones, discusiones y elementos accionables planteados en la reunión de retrospectiva del sprint. El Scrum Master puede facilitar la creación de dicho registro con la aportación de los miembros del equipo principal de Scrum.

**Reunión de retrospectiva del sprint**

La reunión de retrospectiva del sprint tiene un bloque de tiempo asignado de 4 horas para un sprint de un mes y se lleva a cabo como parte del proceso de retrospectiva del sprint. La duración puede escalar hacia arriba o hacia abajo en relación a la duración del sprint. Durante esta reunión, el equipo Scrum se reúne para revisar y reflexionar sobre el sprint anterior en términos de los procesos que fueron aplicados, las herramientas empleadas y los mecanismos de colaboración y comunicación, así como otros aspectos relevantes al proyecto.

**Retorno sobre la inversión (RSI)**

El retorno sobre la inversión, al utilizarse para la justificación de un proyecto, evalúa los ingresos netos esperados que se buscan obtener a partir de un proyecto. Se calcula deduciendo los costos esperados o la inversión en un proyecto de su ingreso previsto; después se divide (la utilidad neta) por los costos previstos a fin de obtener la tasa de retorno.

**Riesgo**

El riesgo se define como un evento incierto o un conjunto de eventos que pueden afectar los objetivos de un proyecto pudieran contribuir a su éxito o fracaso.

**Apetito de riesgo**

El apetito de riesgo es la cantidad de incertidumbre que está dispuesta a asumir un socio o una organización.

**Evaluación de riesgo**

La evaluación de riesgo es la evaluación y cálculo de los riesgos identificados.

**Actitud de riesgo**

En esencia, la actitud de riesgo de los socios determina cuánto riesgo considera aceptable el socio(s). Es un factor determinante cuando deciden tomar acciones para mitigar posibles riesgos adversos.

**Aversión al riesgo**

La aversión al riesgo es una de las categorías de la función de utilidad. Describe al socio que no está dispuesto a aceptar un riesgo sin importar el beneficio o la oportunidad anticipada.

**Estructura de distribución de riesgos**

En esta estructura, se agrupan los riesgos con base en sus categorías o modalidades. Por ejemplo, los riesgos se pueden clasificar como financieros, técnicos o relacionados a la seguridad.

**Gráfica de trabajo pendiente del riesgo (*Risk Burndown Chart*)**

Conocida en inglés como *Risk Burndown Chart*, es una gráfica que muestra la gravedad acumulativa del riesgo a lo largo del tiempo. La probabilidad de los varios riesgos se encima para mostrar el riesgo acumulativo en el eje vertical. La identificación y evaluación inicial de los riesgos, así como la creación de la gráfica de trabajo pendiente se realiza al principio del proyecto.

### **Listas de verificación de riesgos**

Las listas de verificación de riesgos incluyen puntos clave a considerarse al momento de identificar los riesgos, los riesgos comunes que se encuentran en proyectos Scrum o incluso categorías de riesgos que debe abordar el equipo.

### **Comunicación de riesgo**

La comunicación de riesgo implica la comunicación de los resultados de los primeros cuatro pasos de la gestión de riesgos a los socios apropiados y determinar su percepción respecto a eventos inciertos.

### **Identificación de riesgo**

La identificación de riesgo es un paso importante en la gestión de riesgos, que implica el uso de varias técnicas para identificar todos los posibles riesgos.

### **Reunión de riesgos**

Los riesgos se pueden priorizar más fácilmente por el propietario del producto convocando a una reunión del equipo principal de Scrum y, de manera opcional, invitar a la reunión a los socios relevantes.

### **Mitigación de riesgo**

La mitigación de riesgo es un paso importante en la gestión de riesgos que implica el desarrollo de una estrategia adecuada para hacer frente a un riesgo.

### **Neutral al riesgo**

Neutral al riesgo es una de las categorías de la función de utilidad que describe al socio que ni tiene aversión al riesgo, ni busca riesgos; cualquier decisión que no se ve afectada por el nivel de incertidumbre de los resultados. Cuando dos posibles escenarios llevan el mismo nivel de beneficio, el socio neutral al riesgo no se preocupará si uno de dichos casos es más riesgoso que el otro.

### **Priorización de riesgos**

La priorización de riesgos es un paso importante en la gestión de riesgos que implica la priorización de estos y que habrán de incluirse en una acción específica en la lista priorizada de pendientes del producto.

**Lista corta de riesgos (*Risk Prompt Lists*)**

Conocidas en inglés como *Risk Prompt List*, estas listas se utilizan para estimular el pensamiento respecto a la fuente de donde se pudieran originar los riesgos. Dichas listas para distintas industrias y tipos de proyectos están disponibles al público.

**Búsqueda de riesgos (*Risk Seeking*)**

La búsqueda de riesgos es una de las categorías de la función de utilidad que se refiere a que un socio esté dispuesto a aceptar el riesgo incluso si ofrece un aumento marginal de retorno o beneficio al proyecto.

**Umbral de riesgo (*Risk Threshold*)**

El umbral de riesgo es el nivel al cual el riesgo es aceptable para la organización del socio. Un riesgo caerá por encima o debajo del umbral de riesgo. Si está por debajo, el socio o la organización es más probable que acepte el riesgo.

**Tolerancia de riesgo**

La tolerancia de riesgo indica el grado, cantidad o volumen de riesgo que resistirán los socios.

**Despunte basado en riesgo (*Risk-Based Spike*)**

Los despuntes basados en riesgo son básicamente experimentos que implican una investigación o hacer un prototipo para entender mejor los riesgos potenciales. En un *spike*, se lleva a cabo un intenso ejercicio de dos a tres días (de preferencia al inicio de un proyecto, antes del proceso de desarrollo de épicas o de la creación de la lista priorizada de pendientes del producto) para ayudar al equipo a determinar las incertidumbres que pudieran afectar al proyecto.

**Alcance (*Scope*)**

El alcance de un proyecto es la suma total de todos los incrementos del producto y todo el trabajo necesario para desarrollar el producto final.

**Cuerpo de asesoramiento de Scrum (*Scrum Guidance Body*)**

El cuerpo de asesoramiento de Scrum (SGB, por sus siglas en inglés), es un rol opcional. Por lo general se compone de un grupo de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de objetivos relacionados a la calidad, regulaciones gubernamentales, seguridad y demás parámetros organizacionales clave.

### **Experiencia del cuerpo de asesoramiento de Scrum**

La experiencia del cuerpo de asesoramiento de Scrum se refiere a las reglas y regulaciones documentadas, directrices de desarrollo o estándares y mejores prácticas.

### **Scrum Master**

El Scrum Master es uno de los roles en el equipo principal de Scrum. Él o ella facilitan la creación de entregables del proyecto, gestiona riesgos, cambios e impedimentos durante el proceso de llevar a cabo la reunión diaria de pie, retrospectiva del sprint y demás procesos de Scrum.

### **Reunión de Scrum de Scrums**

La reunión de Scrum de Scrums es una reunión importante en el escalamiento de Scrum para grandes proyectos y al cual asisten los representantes de todos los equipos. Dicha reunión es generalmente facilitada por el jefe Scrum Master y su objetivo es enfocarse en áreas de coordinación e integración entre los distintos equipos de Scrum. Esta reunión se lleva a cabo en intervalos predeterminados o cuando lo requieran los equipos de Scrum.

### **Equipo Scrum**

El equipo Scrum es uno de los roles del equipo principal de Scrum. El equipo Scrum trabaja en la creación de entregables del proyecto y contribuye a la realización del valor del negocio para todos los socios y del proyecto.

### **Lecciones aprendidas del equipo Scrum**

Se espera que el equipo Scrum (equipo auto organizado y empoderado) aprenda de los errores cometidos durante el sprint y que estas lecciones aprendidas ayuden a mejorar su desempeño en futuros sprints.

### **Representantes del equipo Scrum**

Representante nombrado por el equipo para que lo represente en las reuniones de Scrum de Scrums, basado en quien puede desempeñar mejor el rol dependiendo de los actuales problemas y circunstancias.

### **Tablero de Scrum (*Scrumboard*)**

El tablero de Scrum es una herramienta utilizada por el equipo Scrum para planificar y dar seguimiento al proceso durante cada sprint. El tablero de Scrum contiene cuatro columnas para indicar el progreso de las tareas estimadas para el sprint: una columna “por hacer” (*To Do*) para las tareas que aún no inician; una columna “en progreso” (*In Progress*) para las tareas iniciadas, pero que no se han terminado; una columna de “prueba” (*Testing*) para tareas terminadas pero que están en proceso de prueba; y la columna de “terminado” (*Done*) para las tareas que se han terminado y examinado satisfactoriamente.

## **Auto organización**

Scrum cree que los empleados son trabajadores auto motivados y buscan aceptar mayores responsabilidades. Por lo tanto, pueden ofrecer mucho más valor cuando se organizan por cuenta propia.

## **Líderes serviciales (*Servant Leaders*)**

Los líderes serviciales escuchan, sienten empatía, compromiso, y tienen percepciones al compartir el poder y autoridad con los miembros de equipo. Los líderes serviciales son sirvientes que logran resultados enfocándose en las necesidades del equipo. Este estilo es la personificación del rol del Scrum Master.

## **Envío de entregables**

En este proceso, los entregables aceptados se envían o se pasan a los socios pertinentes. Un acuerdo formal de los entregables funcionales documenta la conclusión satisfactoria del sprint.

## **Esquemas simples**

Los esquemas simples implican etiquetar elementos como prioridad “1”, “2”, “3” o “alta”, “media” y “baja”, y así sucesivamente. Aunque se trata de un método sencillo y directo, puede llegar a ser problemático, ya que a menudo hay una tendencia en etiquetar todo como prioridad “1” o “alta”.

## **Matriz de requerimiento de habilidades (*Skills Requirement Matrix*)**

La matriz de requerimiento de habilidades, conocida también como marco de competencias, se utiliza para evaluar las carencias de cualificaciones y requisitos de formación para los miembros del equipo. Una matriz de requerimiento de habilidades traza las habilidades, capacidades y nivel de interés de los miembros del equipo en el uso de dichas habilidades y capacidades en un proyecto. Con el uso de esta matriz, la organización puede evaluar cualquier brecha de habilidades en los miembros del equipo e identificar a los empleados que requieren de mayor capacitación en un área o competencia en particular.

## **Lancha rápida (*Speed Boat*)**

“Lancha rápida”, (del inglés: *Speed Boat*) es una técnica que se puede utilizar para llevar a cabo la reunión de retrospectiva del sprint. Los miembros juegan el papel de la tripulación de la lancha. La lancha debe llegar a una isla: simbólicamente la visión del proyecto. Los asistentes utilizan notas adhesivas se utilizan para llevar un registro de motores y anclas. Los motores son cosas que ayudan a llegar a la isla, mientras que las anclas son cosas que están obstaculizando la llegada. Este ejercicio tiene un bloque de tiempo asignado de unos cuantos minutos.

**Patrocinador (*Sponsor*)**

El patrocinador es la persona o la organización que provee recursos y apoyos al proyecto. El patrocinador es también el socio a quien todos rinden cuentas al final.

**Sprint**

Un sprint es una iteración con un bloque de tiempo asignado de una a seis semanas de duración durante el cual el equipo Scrum crea y trabaja en los entregables del sprint.

**Lista de pendientes del sprint (*Sprint Backlog*)**

La lista de pendientes del sprint es una lista de tareas a ser ejecutadas por el equipo Scrum en el próximo sprint.

**Gráfica de trabajo pendiente del sprint (*Sprint Burndown Chart*)**

La gráfica de trabajo pendiente del sprint, conocida en inglés como *Sprint Burndown Chart*, muestra la cantidad de trabajo pendiente en el actual sprint.

**Entregables del sprint (*Sprint Deliverables*)**

Los entregables del sprint son los incrementos del producto o los entregables que se terminan al final de cada sprint.

**Reunión de planificación del sprint**

La reunión de planificación del sprint se lleva a cabo al inicio de un sprint como parte del proceso de creación de la lista de pendientes del sprint. Tiene un bloque de tiempo asignado de ocho horas por un sprint de un mes y se divide en dos partes: definición del objetivo y estimación de tareas.

**Reunión de revisión del sprint**

La reunión de revisión del sprint tiene un bloque de tiempo asignado de cuatro horas por un sprint de un mes y puede escalarse según la duración del sprint. Durante la reunión de revisión del sprint, el equipo Scrum presenta los entregables del actual sprint al propietario del producto, quien puede aceptar o rechazar los entregables.

**Herramientas de seguimiento del sprint**

Las herramientas de seguimiento del sprint se utilizan para rastrear el proceso de un sprint y para saber dónde está el equipo Scrum en términos de conclusión de tareas en la lista de pendientes del sprint. Se puede utilizar una variedad de herramientas para dar seguimiento al trabajo de un sprint, pero una de las más comunes es el tablero de Scrum, conocido también como tablero de tareas o gráfica de proceso.

**Velocidad del sprint (*Sprint Velocity*)**

La velocidad del sprint es la velocidad en la que el equipo puede completar el trabajo en un sprint. Por lo general se expresa en las mismas unidades que se utilizan para la estimación, normalmente puntos de historia o tiempo ideal.

**Socio(s) (*Stakeholder[s]*)**

Socio (del inglés: *stakeholder*) es un término colectivo que incluye a clientes, usuarios y patrocinadores que interactúan frecuentemente con el propietario del producto, con el Scrum Master y con el equipo Scrum para brindar opiniones y facilitar la creación del producto del proyecto, servicio u otros resultados.

**Etapas de enfrentamiento (*Storming Stage*)**

La etapa de enfrentamiento, conocida en inglés como *Storming Stage*, es la segunda etapa de formación del equipo donde este empieza a tratar de cumplir con el trabajo. Sin embargo, pueden ocurrir luchas por el poder resultando en caos o confusión entre los miembros del equipo.

**Mapeo de historias (*Story Mapping*)**

El mapeo de historias, conocido en inglés como *Story Mapping*, es una técnica para proporcionar un esquema visual del producto y sus componentes clave. El mapeo de historias, formulado por primera vez por Jeff Patton (2005), se utiliza comúnmente para ilustrar la ruta del producto. Los mapas de historia representan la secuencia de las iteraciones de desarrollo del producto y trazan las características que serán incluidas en el primer, segundo, tercero y subsecuentes lanzamientos.

**Ritmo sostenible (*Sustainable Pace*)**

El ritmo sostenible es el ritmo al cual el equipo puede trabajar y sostener cómodamente. Esto se traduce en una mayor satisfacción de los empleados, la estabilidad y el aumento de la precisión de la estimación; todo lo cual en última instancia conduce a una mayor satisfacción del cliente.

**Análisis FODA**

FODA es un enfoque estructurado para la planificación de un proyecto que ayuda a evaluar las fortalezas, debilidades, oportunidades y amenazas relacionadas a un proyecto. Este tipo de análisis ayuda a identificar tanto los factores internos como externos que pudieran impactar el proyecto.

**Clientes meta para el lanzamiento (*Target Customers for Release*)**

No todos los lanzamientos están dirigidos a todos los socios o usuarios. Los socios pueden optar por limitar ciertos lanzamientos a un subconjunto de usuarios. El plan de lanzamiento especifica los clientes meta para el lanzamiento.

**Taller de estimación de tareas**

El taller de estimación de tareas permite al equipo Scrum estimar el esfuerzo requerido para terminar una tarea o serie de tareas y estimar el trabajo necesario del personal para realizar las tareas dentro de un determinado sprint.

**Lista de tareas**

Lista completa que contiene todas las tareas a las que se ha abocado el equipo Scrum en el actual sprint. Contiene descripciones de cada tarea.

**Reunión de planificación de tareas**

En una reunión de planificación de tareas, el equipo Scrum se reúne para planear el trabajo que se hará en el sprint y revisar las historias de usuario asignadas en la parte superior de la lista priorizada de pendientes del producto. Para ayudar a garantizar que el grupo no se salga del tema, la reunión debe tener un bloque de tiempo asignado, con una duración estándar limitada a dos horas por semana de duración del sprint.

**Líder enfocado en las tareas (*Task-Oriented Leader*)**

Los líderes enfocados en las tareas hacen cumplir la realización de tareas y el apego a los plazos.

**Plan de formación de equipo (*Team Building Plan*)**

Debido a que un equipo Scrum es multifuncional, cada miembro debe participar activamente en todos los aspectos del proyecto. El Scrum Master debe identificar los posibles problemas que pudieran surgir con los miembros del equipo y tratar de resolverlos con diligencia en el plan de formación de equipo a fin de mantener un equipo eficaz.

**Calendario del equipo**

El calendario del equipo contiene información sobre la disponibilidad de los miembros, incluyendo información relacionada a las vacaciones de los empleados, licencias, ausencias temporales y días festivos.

## Experiencia del equipo

La experiencia del equipo es el conocimiento con el que cuentan los miembros del equipo Scrum para comprender las historias de usuario y las tareas en la lista de pendientes del sprint a fin de crear los entregables finales. La experiencia del equipo se utiliza para evaluar las entradas necesarias para ejecutar el trabajo previsto del proyecto.

## Deuda técnica

La deuda técnica (conocida también como deuda de diseño o deuda de código) es el trabajo al que los equipos dan menor prioridad; el trabajo que omiten o que no terminan a medida que trabajan en la creación de los principales entregables asociados al producto del proyecto. La deuda técnica se acumula y se debe saldar a futuro.

## Teoría X

En la Teoría X, los líderes suponen que los empleados están inherentemente desmotivados y evitarán el trabajo si es posible, lo cual justifica un estilo autoritario de gestión.

## Teoría Y

En la Teoría Y, los líderes asumen que los empleados son auto motivados y buscan aceptar una mayor responsabilidad. La Teoría Y implica un estilo más participativo de gestión.

## Amenazas

Las amenazas son riesgos que pudieran afectar el proyecto en forma negativa.

## Tres preguntas diarias

Las tres preguntas diarias se utilizan en las reuniones diarias de pie, organizadas por el Scrum Master, donde cada miembro del equipo Scrum brinda información en forma de respuesta a tres preguntas específicas:

4. ¿Qué terminé ayer?
5. ¿Qué terminaré hoy?
6. ¿Qué impedimentos u obstáculos (si los hubiera) estoy enfrentando en la actualidad?

**Asignación de bloque de tiempo (*Time-boxing*)**

La asignación de bloque de tiempo es la fijación de breves periodos para realizar el trabajo. Si el trabajo asumido permanece incompleto al final del bloque de tiempo, se traslada al subsecuente bloque. Los bloques de tiempo proporcionan la estructura necesaria para los proyectos Scrum, los cuales tienen un elemento de incertidumbre, son de naturaleza dinámica y son propensos a cambios frecuentes.

**Transparencia**

La transparencia permite que todas las facetas de cualquier proceso de Scrum puedan ser observadas por cualquiera. Compartir toda la información conduce a un ambiente de alta confianza.

**Solicitudes de cambio no aprobadas**

Las peticiones de cambio se presentan por lo general como solicitudes de cambio. Estas permanecen no aprobadas hasta que se autorizan formalmente.

**Lista actualizada de pendientes del producto del programa (*Updated Program Product Backlog*)**

Una lista actualizada de pendientes del producto del programa que se somete a una preparación periódica para incorporar cambios y agregar nuevos requerimientos.

**Usuario**

Los usuarios son los individuos o la organización que utiliza directamente el producto del proyecto, el servicio u otros resultados. Al igual que los clientes, para cualquier organización, puede haber tanto usuarios internos y como externos. En algunos casos, los clientes y los usuarios pueden ser los mismos.

**Reuniones de grupo de usuarios**

Las reuniones de grupo de usuarios incluyen a los socios relevantes, principalmente usuarios o clientes del producto. Ellos le proporcionan al equipo principal de Scrum la información de primera mano sobre las expectativas del usuario. Esto ayuda en la formulación de los criterios de aceptación para el producto y proporciona información valiosa para el desarrollo de épicas.

**Historias de usuario**

Las historias de usuario se adhieren a una estructura específica y predefinida y son una manera simplista de documentar los requisitos y la funcionalidad deseada del usuario final. Los requerimientos expresados en las historias de usuario son afirmaciones breves, simples y fáciles de entender, lo cual resulta en una mejor comunicación entre socios y mejores estimaciones por parte del equipo.

### **Criterios de aceptación de historias de usuario**

Cada historia de usuario tiene sus criterios de aceptación. Las historias de usuario son subjetivas, por lo que los criterios de aceptación proporcionan la objetividad requerida para que la historia de usuario sea considerada como terminada o no terminada durante la revisión del sprint, dando claridad al equipo sobre lo que se espera de una historia de usuario.

### **Talleres de historias de usuario**

Los talleres de historias de usuario se llevan a cabo como parte del proceso de desarrollo de épica(s). El Scrum Master facilita estas sesiones. Todo el equipo principal de Scrum participa y en ocasiones es conveniente incluir otros socios.

### **Experiencia en la redacción de historias de usuario**

El propietario del producto, con base en su interacción con los socios, en su experiencia y conocimiento del negocio, así como en las opiniones del equipo, desarrolla las historias de usuario que forman la primera lista priorizada de pendientes del producto para el proyecto.

### **Función de utilidad (*Utility Function*)**

La función de utilidad es un modelo utilizado para medir la preferencia del socio por el riesgo o su actitud hacia el riesgo. Esto define el nivel del socio para aceptar riesgos.

### **Mapa de flujo de valor (*Value Stream Mapping*)**

El mapa de flujo de valor utiliza diagramas de flujo para ilustrar el flujo de información necesario para completar un proceso y pudiera utilizarse para racionalizar un proceso ayudando a determinar los elementos que no aportan valor.

### **Vendedor**

Los vendedores son individuos externos u organizaciones que ofrecen productos y servicios que no están dentro de las competencias básicas de la organización del proyecto.

### **Voz del cliente (*Voice of the Customer*)**

A la voz del cliente se le puede conocer como los requerimientos explícitos e implícitos del cliente que se deben entender antes de diseñar un producto o servicio. El propietario del producto representa la voz del cliente.

**Salón de guerra (*War Room*)**

Salón de guerra, o *War Room* en inglés, es el término que comúnmente se utiliza para describir la ubicación donde trabajan todos los miembros del equipo Scrum. Normalmente, está diseñado de tal manera que los miembros del equipo pueden moverse libremente, trabajar y comunicarse fácilmente, ya que se encuentran ubicados en proximidades inmediatas.

**Técnica Wideband Delphi**

Wideband Delphi es una técnica de estimación basada en grupo para determinar la cantidad de trabajo necesario y el tiempo que tardará en completarse. Los individuos en el equipo proporcionan estimaciones anónimas para cada característica y las estimaciones iniciales se trazan en una gráfica. Posteriormente, el equipo analiza los factores que influyeron en sus estimaciones y proceden a una segunda ronda de estimación. Este proceso se repite hasta que las estimaciones de los individuos quedan cerca una de la otra y se puede llegar a un consenso para la estimación final.

**Entregables funcionales (*Working Deliverables*)**

Esta salida es el entregable final para enviar (del inglés: *shippable deliverable*) para el que fue sancionado el proyecto.

**Acuerdo de entregables funcionales**

Los entregables que cumplen con los criterios de aceptación, reciben el cierre formal del negocio y la aprobación formal por parte del cliente o del patrocinador.

# INDEX

## 1

100-point method, 166

## A

Acta constitutiva del proyecto, 141

Acuerdo de entregables funcionales, 253

**Adaptabilidad**, 4

Adaptación, 23

**Alta velocidad**, 5

**Ambiente de alta confianza**, 5

**Ambiente innovador**, 5

Análisis de brecha, 140

**Apetito de riesgo**, 116

Appropriation, 27

Aprobación, estimación y asignación de historias de usuario, 16, 177

Aprobación, estimación y asignación de historias de usuarios

Entradas, 188

Herramientas, 188

Salidas, 192

Aprobadas de cambio, 97

Articulation, 27

Asesoramiento de expertos RH, 146

**Asignación de un bloque de tiempo**, 10, 21

Aspectos de Scrum, 10

**Autocrático**, 59

**Auto-organización**, 9, 20

Awareness, 27

## B

**Brainstorming**, 118

## C

Calidad, 13

Calidad y valor del negocio, 83

Cambio, 13

Capacitación y costos de capacitación, 147

Cartera, 49

**caso de negocio**, 68

Caso de negocio del proyecto, 136

**Centrado en el cliente**, 4

Cientes meta para el lanzamiento, 175

**Colaboración**, 9, 20

**Colocated Teams**, 30

**Confirmar la realización de beneficios**, 69

Confirmar la realización de beneficios, 78

Contratos aplicables, 157

**Control del proceso empírico**, 9, 20, 21

Convocar a un Scrum de Scrums

Entradas, 233

Herramientas, 234

Salidas, 235

**Convocar a un Scrum de Scrums**, 229, 232

Convocar el Scrum de Scrums, 17

Costo actual, 75

Costo de los recursos, 147

Creación de entregables

Entradas, 212

Herramientas, 214

Salidas, 215

Creación de entregables, 17, 207, 210

Creación de historias de usuario

Herramientas, 183

Salidas, 185

Creación de historias de usuario, 16, 177, 180

Entradas, 182

Creación de la lista de pendientes del sprint

Entradas, 202

Herramientas, 203

Salidas, 204

Creación de la lista de pendientes del sprint, 17, 201

**Creación de la lista de pendientes del Sprint**, 177

Creación de la lista priorizada de pendientes del producto

Entradas, 165

outputs, 168

Creación de la lista priorizada de pendientes del producto, 16, 131, 163

Creación de la visión del proyecto

Entradas, 136

Salidas, 141

Creación de la visión del proyecto, 16, 131, 134

Creación de la visión del proyecto, 139

Creación de la visión del proyecto, 142

Creación de tareas

Entradas, 193

Herramientas, 194

- Salidas, 196
- Creación de tareas, 17, 177, 192
- Create Prioritized Product Backlog
  - Herramientas, 166
- Criterios de aceptación de historias de usuario, 185
- Criterios de estimación, 199
- Criterios de selección, 146
- Criterios de terminado, 169
- Cronograma de planificación del lanzamiento, 174
- Cronograma de planificación del lanzamiento actualizado, 226
- Cuatro preguntas por equipo, 234
- cuerpo de asesoramiento de Scrum (SGB)**, 11
- Cuestionarios, 160

## D

- Definición de terminado, 87
- Demostración y validación del sprint
  - Entradas, 239
  - Herramientas, 240
  - Salidas, 240
- Demostración y validación del sprint, 17, 229, 237
- Demostraciones, 78
- Dependencias**, 168, 196
- Dependencias discrecionales**, 195
- Dependencias externas**, 195
- Dependencias internas**, 196
- Dependencias obligatorias**, 195
- Desarrollador Scrum Certificado (SDC™), 6
- Desarrollo de épica(s), 16, 131, 154
  - Entradas, 155
  - Herramientas, 158
  - Salidas, 161
- Desarrollo iterativo**, 21, 34, 99
- Desarrollo Iterativo**, 10
- Desempeño, 56
- Disponibilidad y compromiso de las personas, 145
- Duración del sprint, 106, 174

## E

- El equipo principal de Scrum**, 100
- En cartera, 110
- Enfrentamiento, 56
- Entrega basada en el valor**, 63
- Entrega basada en valor, 64
- Entrega continua de valor**, 4
- Entregables aceptados, 240
- Entregables del sprint, 215
- Entregables efectivos**, 4

- Entregables rechazados, 223
- Entrevistas al usuario o cliente, 159
- Envío de entregables
  - Entradas, 252
  - Herramientas, 253
  - Salidas, 253
- Envío de entregables, 18, 249, 251
- Épica (s), 161
- Equilibrio, 98
- Equipo principal Scrum, 155
- Equipo Scrum, 46
- Equipo Scrum motivado, 220
- Escalabilidad de Scrum, 5
- Escalas de tiempo del proyecto**, 68
- Estilos de liderazgo, 59
- Estimación de tareas
  - Herramientas, 199
  - Salidas, 200
- Estimación de tareas, 17, 177, 197
- Estimaciones**, 168
- Estimado al completar, 75
- Estimado para completar, 75
- Evaluación de riesgos, 118
- Experiencia del cuerpo de asesoramiento de Scrum, 160, 191, 215, 235, 240, 245, 257
- Experiencia en la redacción de historias de usuario, 183
- Expert Advice from HR, 151
- Experto Ágil Certificado (AEC™), 6
- Experto Scrum Master Certificado (ESM™), 6

## F

- Fase de implementación—Diagrama de flujo de datos, 227
- Fase de lanzamiento—Diagrama de flujo de datos, 258
- Fase de planificación y estimación—Diagrama de flujo de datos, 205
- Fase de revisión y retrospectiva—Diagrama de flujo de datos, 247
- Formación, 56
- Formación de equipos Scrum, 16
- Formación de un equipo Scrum
  - Entradas, 150
  - Herramientas, 151
  - outputs, 152
- Formación de un equipo Scrum**, 131
- Formación de un equipo Scrum.*, 148

## G

- Ganar-Ganar, 57

Ganar-Perder, 58  
Gestión de conflictos, 57

## H

Historias de usuario, 188  
Historias de usuario, aprobadas, estimadas y asignadas actualizadas, 196

## I

Identificación de riesgos, 117  
Identificación del Scrum Master y el(los) socio(s), 16, 131  
Identificar al Scrum Master y al socio(s), 142  
Entradas, 144  
Herramientas, 146  
Salidas, 147  
Identified Scrum Team, 152  
Implementación, 17  
Índice de desempeño del programa, 75  
Información de proyectos previos, 158  
Inicio, 16, 130  
Inspección, 22  
Integración continua, 104  
Integración del cambio, 104

## J

Jefe propietario del producto, 43, 137  
Jefe Scrum Master, 45, 144  
Justificación del negocio, 12

## L

Lancha rápida, 244  
Lanzamiento, 18  
Lanzamientos del producto, 254  
Leyes y regulaciones, 157  
**Líder asertivo**, 59  
**Líder de apoyo y entrenamiento**, 59  
**Líder delegador**, 59  
**Líder directivo**, 59  
**Líder Laissez Faire**, 59  
**Líder servicial**, 59  
Lista de pendientes del producto del programa, 137  
Lista priorizada de pendientes del producto, 168  
Lista priorizada de pendientes del producto actualizada, 224  
**Listas de verificación de riesgos**, 117

## M

Mantenimiento de la lista priorizada de pendientes del producto  
Entradas, 223  
Herramientas, 225  
Salidas, 226  
Mantenimiento de la lista priorizada de pendientes del producto, 17, 207, 222  
Mantenimiento de la participación de los socios, 53  
Matriz de recursos organizacionales, 145  
Matriz de requerimientos de habilidades, 145  
**Mejora continua**, 4  
Mejoras accionables aceptadas, 245  
Métodos de estimación de historias de usuario, 184  
Métodos de priorización del lanzamiento, 174  
Modelo de dinámica de grupo de Tuckman, 56  
**Motivación**, 4

## N

Normalización, 56

## O

Organización, 10  
Otras técnicas de estimación, 190

## P

**Paired comparison**, 166  
**Patrocinador**, 41  
People Costs, 151  
Perder-Ganar, 58  
Perder-Perder, 58  
Plan de colaboración, 153  
Plan de desarrollo del equipo, 153  
Plan de pilotaje, 252  
Planificar para el valor, 72  
Porcentaje completado, 75  
Presupuesto al finalizar, 75  
Presupuesto del proyecto, 142  
**Priorización basada en el valor**, 21  
Priorización basada en el valor que se entrega al cliente, 103  
**Priorización basada en el valor que se entrega cliente**, 72  
**Priorización basada en valor**, 9, 30  
priorizada de pendientes del producto refinada, 175  
Procedimiento de gestión de riesgos, 117  
Proceso de aprobación de cambios, 97

**Proceso de desarrollo eficiente, 4**

Procesos de Scrum, 15

producto, 1

Programa, 49

Programas, 52

**Propietario del producto de la cartera, 51****Propietario del producto del programa, 52, 136**

Propietario del producto identificado, 141

Propietario del Producto Scrum Certificado (SPOC™), 6

Prototipos, 161

Proyecto, 49

Proyecto de prueba, 138

Prueba de concepto, 138

**R**

Realizar el plan de lanzamiento

Entradas, 172

Herramientas, 173

Salidas, 174

**Realizar el plan de lanzamiento, 131, 170**

Realizar la planificación del lanzamiento, 16

Realizar reunión diaria de pie

Entradas, 218

Herramientas, 219

Salidas, 220

Realizar reunión diaria de pie, 17, 207, 217

Recomendaciones de cuerpo de asesoramiento de

Scrum, 183

Recomendaciones del cuerpo de asesoramiento de

Scrum, 139, 158, 166, 173, 188, 213, 225, 243, 253, 256

*Requerida, 73*

Requerimientos de las personas, 145

Requerimientos de negocio, 165

Requisitos de recursos, 150

**Resolución de problemas de forma más rápida, 4**

Resource Costs, 152

**Responsabilidad colectiva, 5****Retroalimentación continua, 4****Retrospectiva de sprint, 229**

Retrospectiva de Sprint, 18

Retrospectiva del proyecto

Herramientas, 256

inputs, 256

Salidas, 257

Retrospectiva del proyecto, 18, 249, 255

Retrospectiva del sprint

Entradas, 243

Herramientas, 243

Salidas, 245

Retrospectiva del sprint, 242

Reunión de la visión del proyecto, 139, 141

**Reunión de planificación del sprint, 33****Reunión de retrospectiva del sprint, 33****Reunión de revisión del sprint, 33, 240****Reunión de riesgos, 119****Reunión diaria, 32**

Reunión diaria de pie, 219

Reuniones de estimación de tareas, 199

Reuniones de planificación de tareas, 194

Reuniones de planificación del sprint, 203

Reuniones del grupo de usuarios, 158

Revisión y retrospectiva, 17

Riesgo, 14

**Riesgo e incertidumbre, 168**

Riesgos del programa y cartera, 156

Riesgos identificados, 162

**Ritmo sostenible, 4**

Ritmo Sostenible, 90

**Roles centrales, 10****Roles no centrales, 11****S**

Scrum Guidance Body Expertise, 167

Scrum Guidance Body Recommendations, 198

Scrum Master, 44, 233

Scrum Master Certificado (SMC™), 6

**Scrum Master de la cartera, 51****Scrum Master del programa, 52, 136**

Scrum Master identificado, 147

Scrum Team Representatives, 233

Scrum Team Selection, 151

Scrum vs. Gestión de proyectos tradicional, 19

Segmentación, 195

Selección de personal, 48

Sesiones de diseño de aplicación conjunta, 139

Sesiones de planificación del lanzamiento, 173

**Socio(s), 11**

Socio(s) del programa, 137

Socio(s) identificado(s), 148

**Socios (s), 99**

Software, 214

Solicitudes de cambio, 156

Solicitudes de cambio no aprobadas, 156

**Sprint, 32**

Substitutos, 152

**T**

Tablero de Scrum, 212

Talleres de historias de usuario, 159

**Tamaño relativo/Puntos de historia**, 190

Técnicas de comunicación, 225

**Teoría X**, 62

**Teoría Y**, 62

Training and Training Costs, 151

**Transparencia**, 4, 21

## U

User Story Estimation Methods, 167

**Usuarios**, 41

## V

**Valor**, 168

Valor ganado, 75

Valor planeado, 75

Variancia al completar, 75

Varianza del costo, 75

**Vendedores**, 41

Videoconferencia, 220, 234

Visión de la compañía, 138

## W

**Wideband Delphi**, 190

## La guía esencial para entregar proyectos exitosos utilizando Scrum

La Guía *SBOK™* fue desarrollada como un medio para crear una guía necesaria para organizaciones y profesionales que desean implementar Scrum, así como para aquellos que ya lo están haciendo pero desean aun mejorar sus procesos existentes. Se basa en la experiencia extraída de miles de proyectos a través de una variedad de industrias y organizaciones. Las contribuciones de muchos expertos de Scrum y de profesionales de proyecto han sido consideradas en su desarrollo.

El enfoque de Scrum en la entrega del valor ayuda a los equipos Scrum a entregar resultados tan temprano en el proyecto como sea posible, mejorando la rentabilidad de la inversión (Return on Investment) para las empresas que utilizan Scrum como su marco de entrega de proyecto preferido. Por otra parte, el gestionar cambios es fácil mediante el uso de ciclos de desarrollo de producto cortos e iterativos y de interacción frecuente entre los clientes y el equipo Scrum.

La Guía *SBOK™* puede ser utilizada como una guía de referencia y conocimiento tanto por los desarrolladores de productos y servicios con conocimiento de Scrum, así como por individuos sin experiencia previa o conocimiento de Scrum u otra metodología de entrega de proyectos. El primer capítulo describe el propósito y el marco de la Guía *SBOK™* y proporciona una introducción a los conceptos claves de Scrum y un resumen de los principios, aspectos y procesos del mismo. El capítulo 2 expande en los seis principios de Scrum que constituyen la base en la que se basa el marco de Scrum. Los capítulos 3 al 7 elaboran en los cinco aspectos de Scrum que deben ser abordados a lo largo de cualquier proyecto, organización, justificación del negocio, calidad, cambio y riesgo. Los capítulos 8 al 12 cubren los 19 procesos de Scrum para llevar a cabo un proyecto Scrum. Estos procesos son parte de las 5 fases de Scrum de Iniciar; Planear y Estimar; Implementar, Revisión y Retrospectiva; y la Lanzamiento. Se describen los detalles sobre las entradas y salidas asociadas de cada proceso, así como las diversas herramientas que pueden utilizarse en cada una.

Aunque la Guía *SBOK™* es un libro de referencia muy completo de Scrum, su contenido se organiza como una referencia fácil y lectura agradable, cualquiera que sea el conocimiento previo del lector del Scrum.

