



INTRODUCCIÓN

Metodologías ágiles

Albert Álvarez Carulla

05.03.2021

TheAlbert.dev

@thealbertdev



"Introducción a las metodologías ágiles" por Albert Álvarez Carulla se distribuye bajo una [Licencia Creative Commons Atribución-NoComercial-SinDerivadas 4.0 Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/)

Orígenes de Scrum

[1]



[2]



“The New New Product Development Game”
Harvard Business Review, 64(1), 137-146.

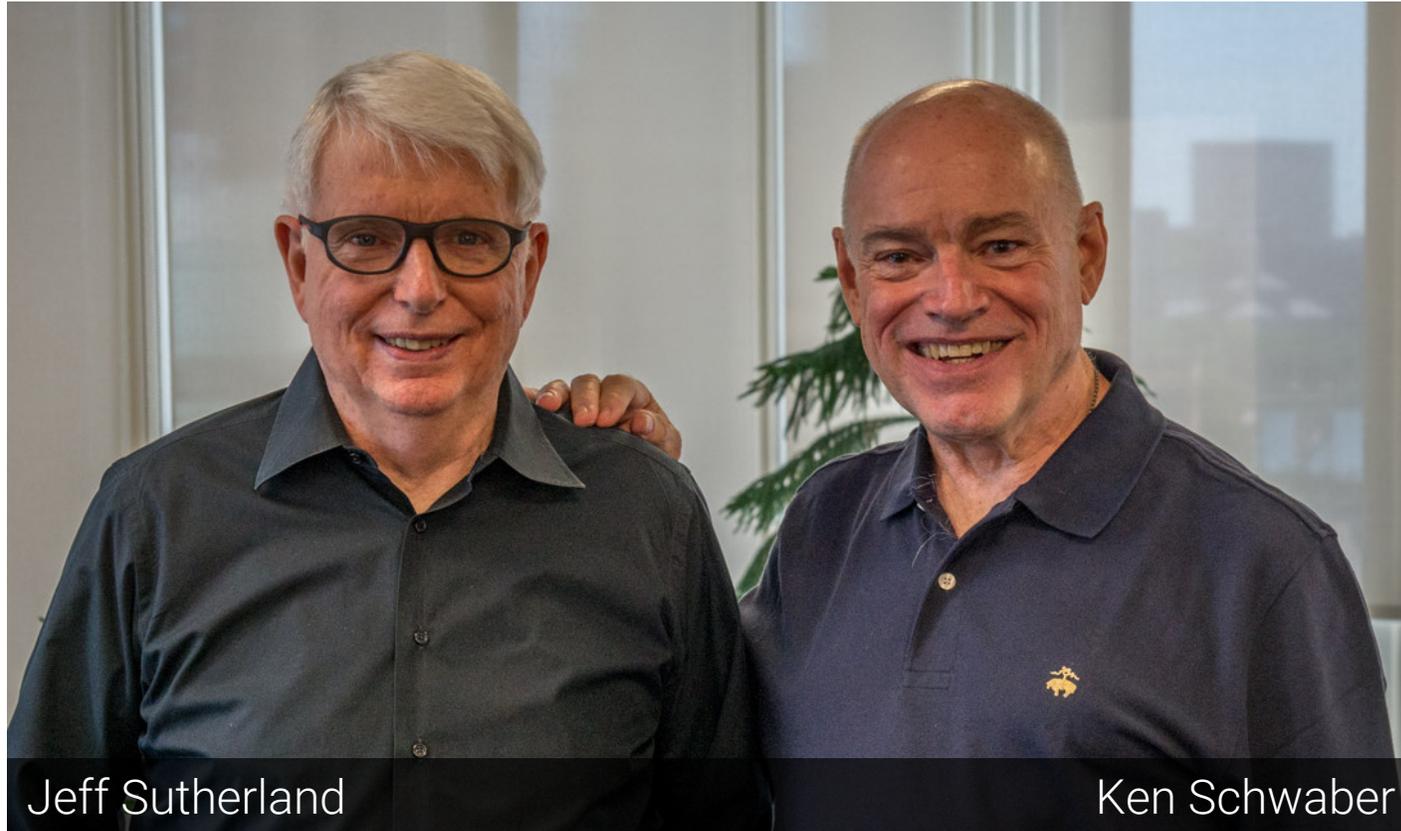
1986



https://www.youtube.com/watch?v=NAksCK6Pp_M

Orígenes de Scrum

[3]

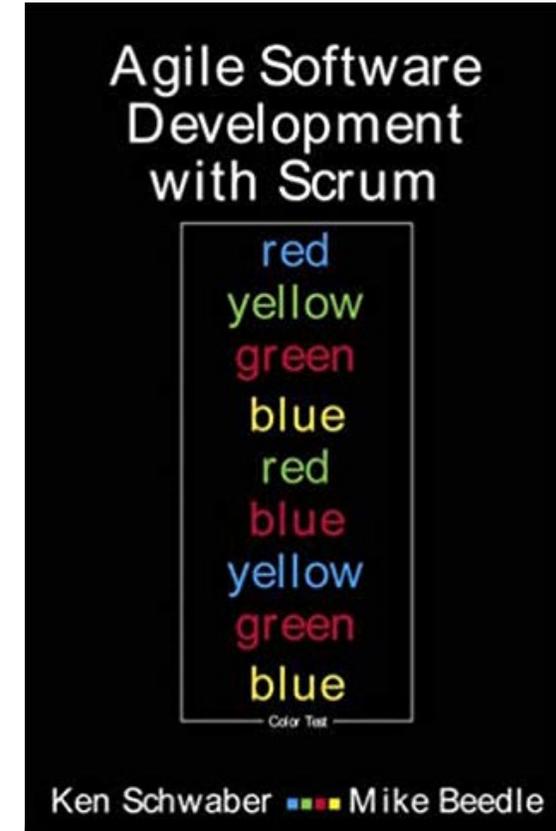


Jeff Sutherland

Ken Schwaber

“Scrum Development Process”
OOPSLA 95

1995



Schwaber, K., & Beedle, M. (2002). Agile software development with Scrum (Vol. 1). Upper Saddle River: Prentice Hall.

Contenidos

01. Tipos de gestión

- Métodos predictivos
- Métodos adaptativos

02. Valores Agile

- Lean, Agile, Scrum
- 3Ms
- El manifiesto Agile
- Los 12 principios
- Shu Ha Ri

03. Scrum

- El proceso
- Los roles
- Artefactos
- Reuniones

01.

Tipos de gestión

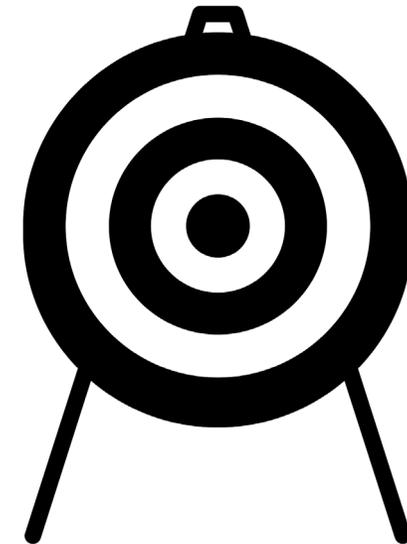
- Métodos predictivos
- Métodos adaptativos

Tipos de métodos de gestión

01. Tipos de gestión | Métodos predictivos



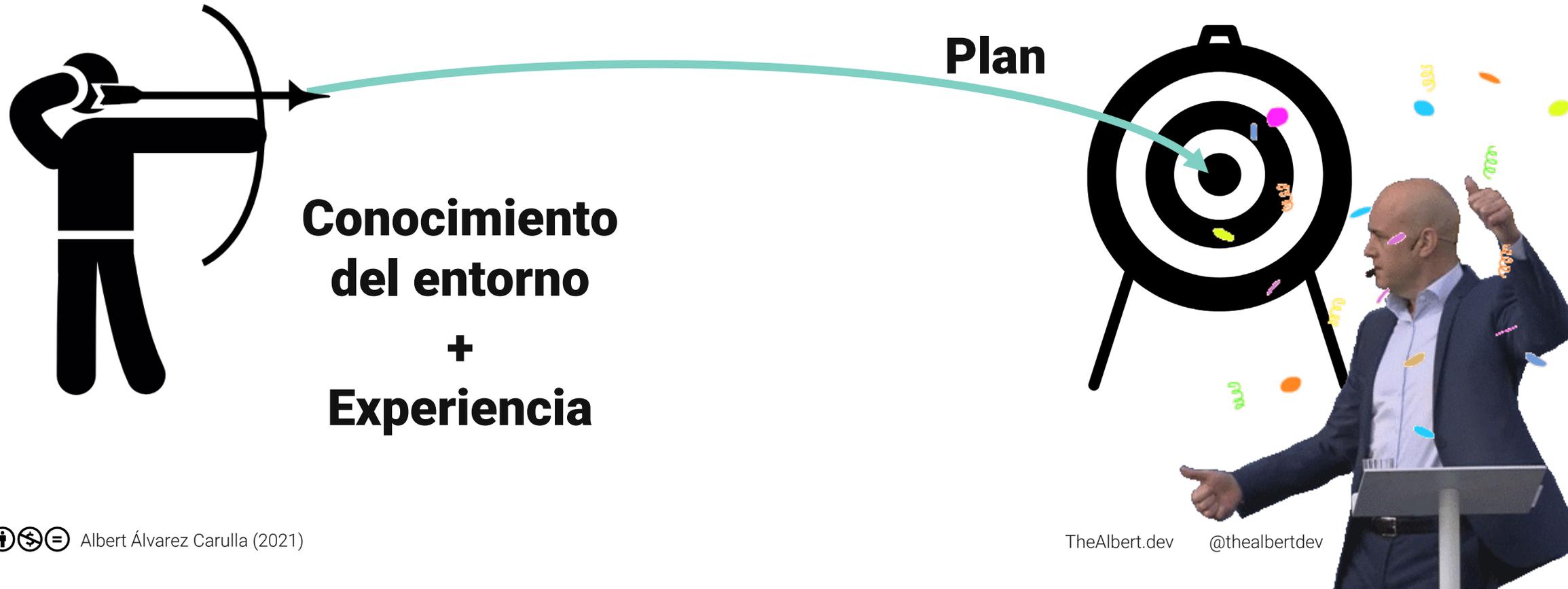
Conocimiento del entorno



Tipos de métodos de gestión

Métodos de gestión predictivos

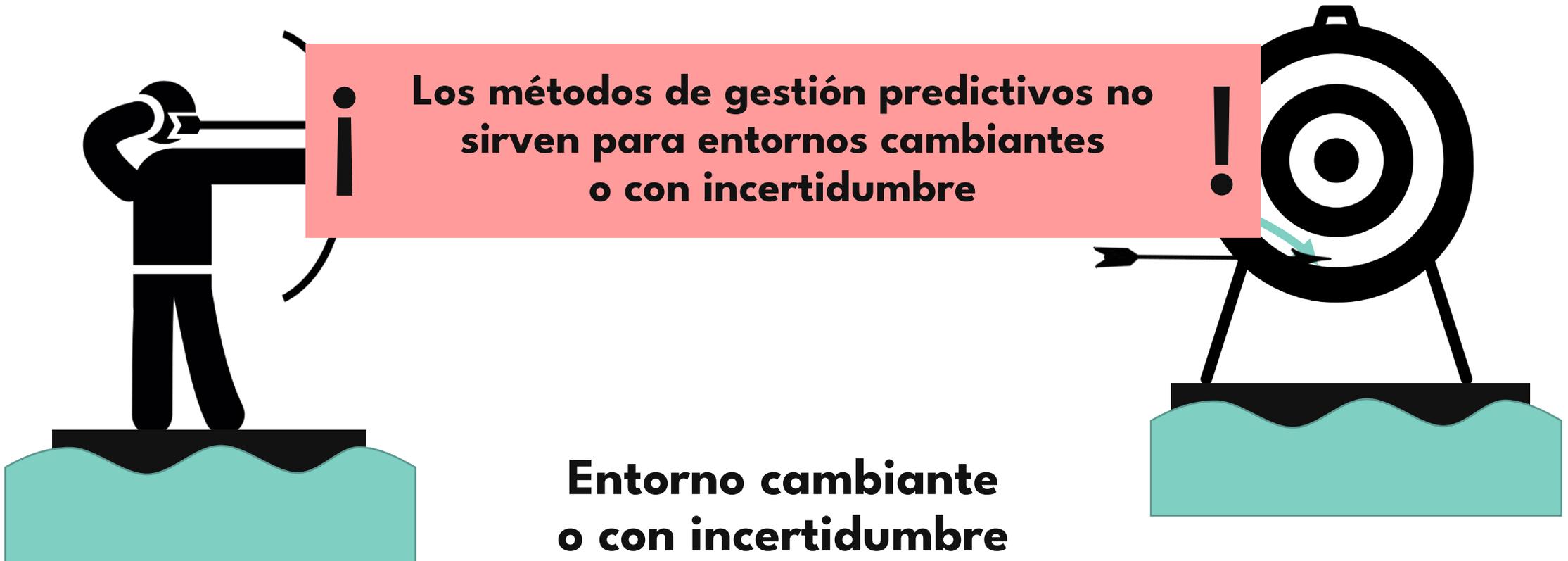
Ejemplo: Desarrollo/modelo en cascada



Tipos de métodos de gestión

Métodos de gestión predictivos

01. Tipos de gestión | Métodos predictivos



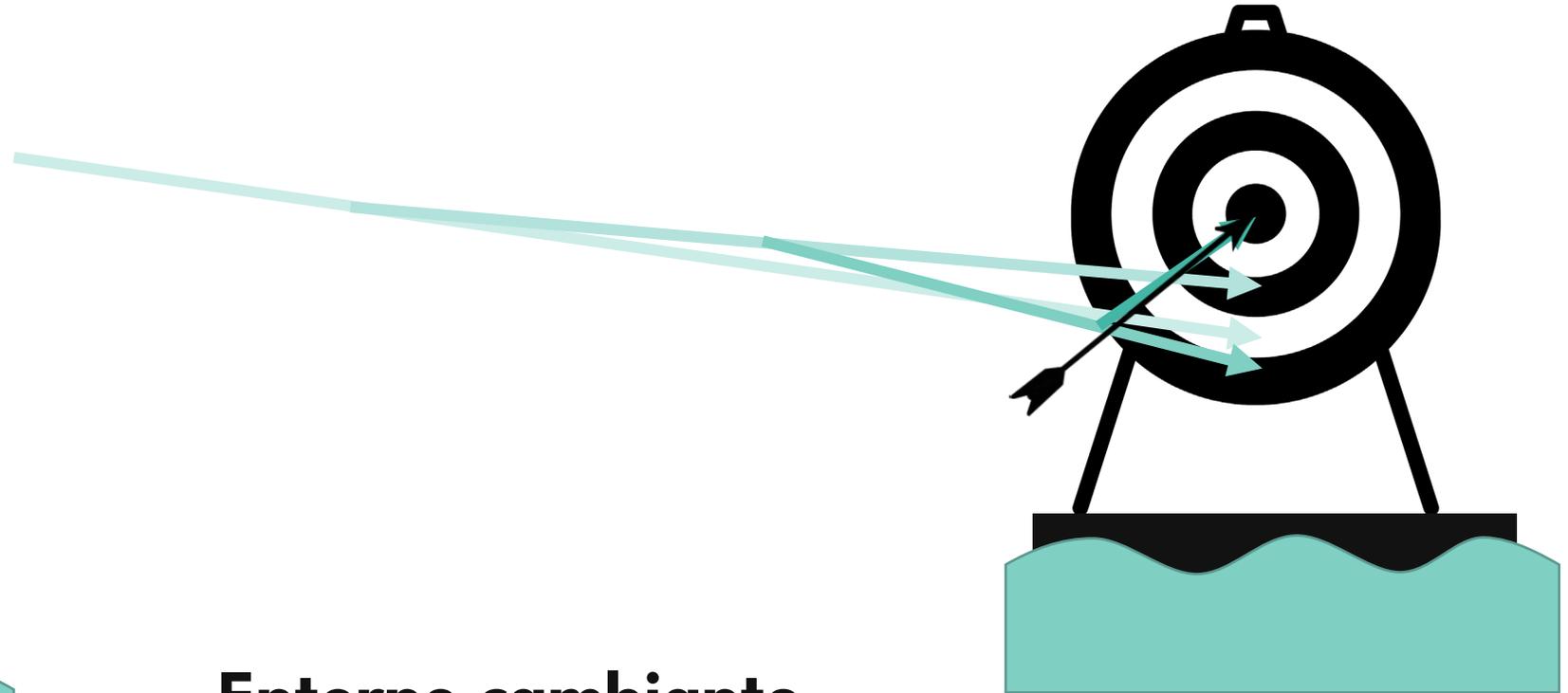
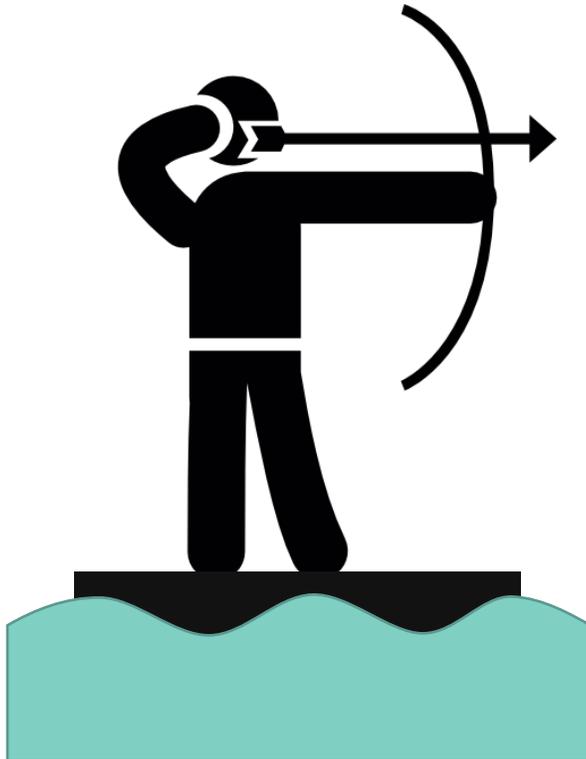
! Los métodos de gestión predictivos no sirven para entornos cambiantes o con incertidumbre **!**

**Entorno cambiante
o con incertidumbre**

Tipos de métodos de gestión

Métodos de gestión adaptativos

01. Tipos de gestión | Métodos adaptativos



**Entorno cambiante
o con incertidumbre**

Tipos de métodos de gestión

Métodos de gestión adaptativos

01. Tipos de gestión | Métodos adaptativos



**Entorno cambiante
o con incertidumbre**

“Un diagrama de Gantt no sirve para aterrizar un avión”

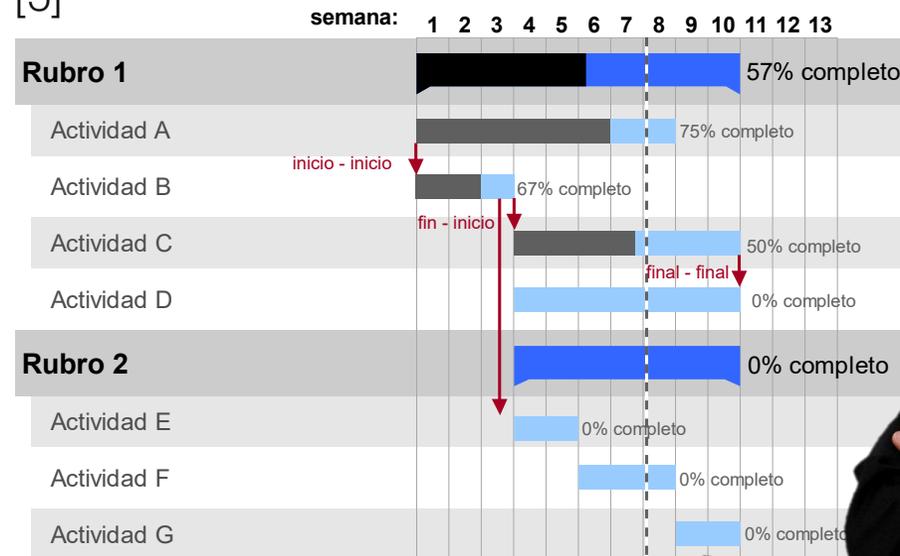
01. Tipos de gestión | Métodos adaptativos

[4]



Si lo logras, me llamas...

[5]



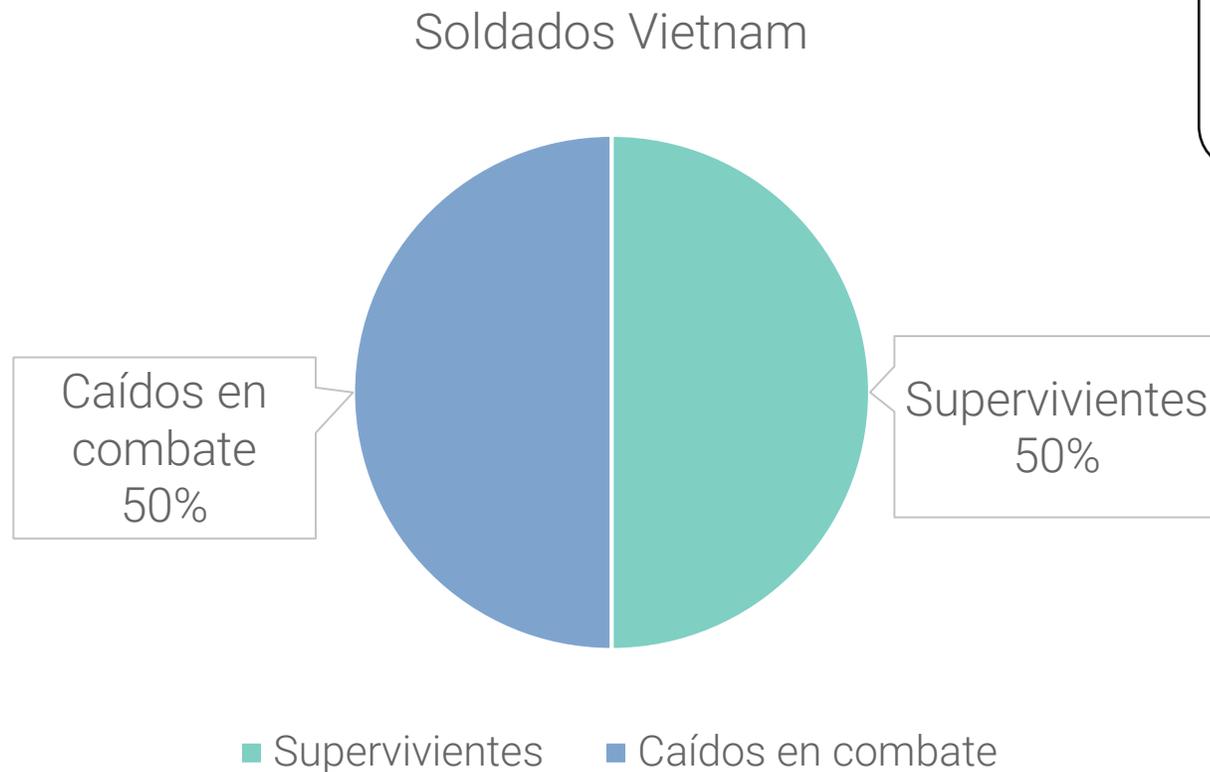
[6]



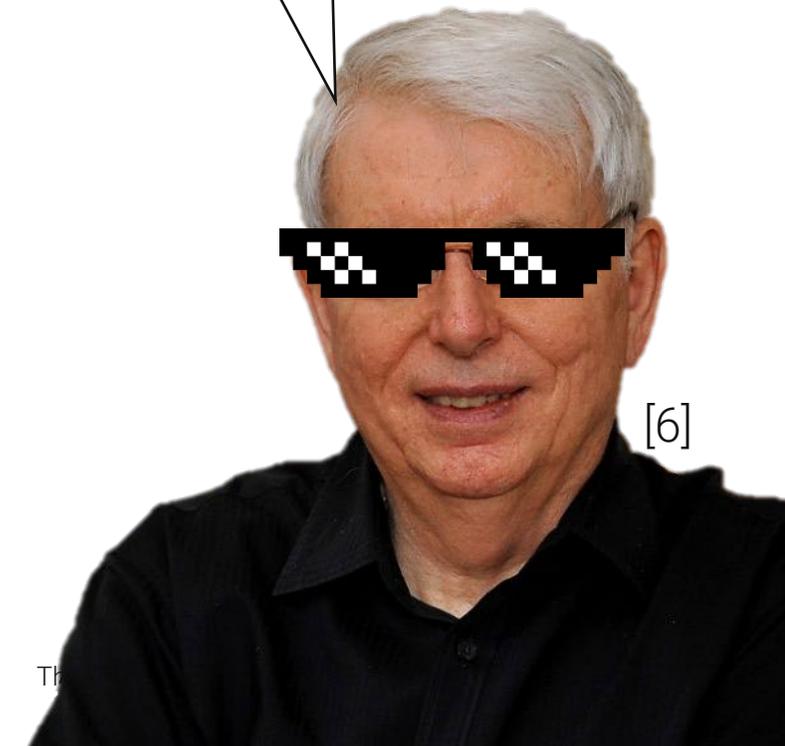
Expiloto veterano de Vietnam

“Sobrevivimos solo el 50% de los que fuimos a Vietnam”

01. Tipos de gestión | Métodos adaptativos



¿Cómo sobreviví?
No siguiendo el plan.



[6]

02.

Valores Agile

- Lean, Agile, Scrum
- 3Ms
- El manifiesto Agile
- Los 12 principios
- Shu Ha Ri

Lean, Agile, Scrum

Maximizar **valor** para el cliente

Lean

Mentalidad, valores y **principios**

Agile

Framework para la gestión adaptativa de un proyecto

Scrum

02. Valores Ágile | Lean, Agile, Scrum

3Ms



Muda

Actividad que no
agrega valor



Mura

Variabilidad, Mura
crea Muda



Muri

Sobrecarga,
normalmente
causado por Mura

El manifiesto Agile

<https://agilemanifesto.org/iso/es/manifesto.html>

Manifiesto por el Desarrollo Ágil de Software

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:

Individuos e interacciones sobre procesos y herramientas
Software funcionando sobre documentación extensiva
Colaboración con el cliente sobre negociación contractual
Respuesta ante el cambio sobre seguir un plan

Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Los 12 principios

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software producto 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 producto 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 producto 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.

<https://agilemanifesto.org/iso/es/principles.html>

Shu Ha Ri



Shu

Aprende la regla



Ha

Rompe la regla



Ri

Define la regla

[7-9]

03.

Scrum

- El proceso
- Los roles
- Artefactos
- Reuniones

03.

Scrum

- El proceso
- Los roles
- Artefactos
- Reuniones

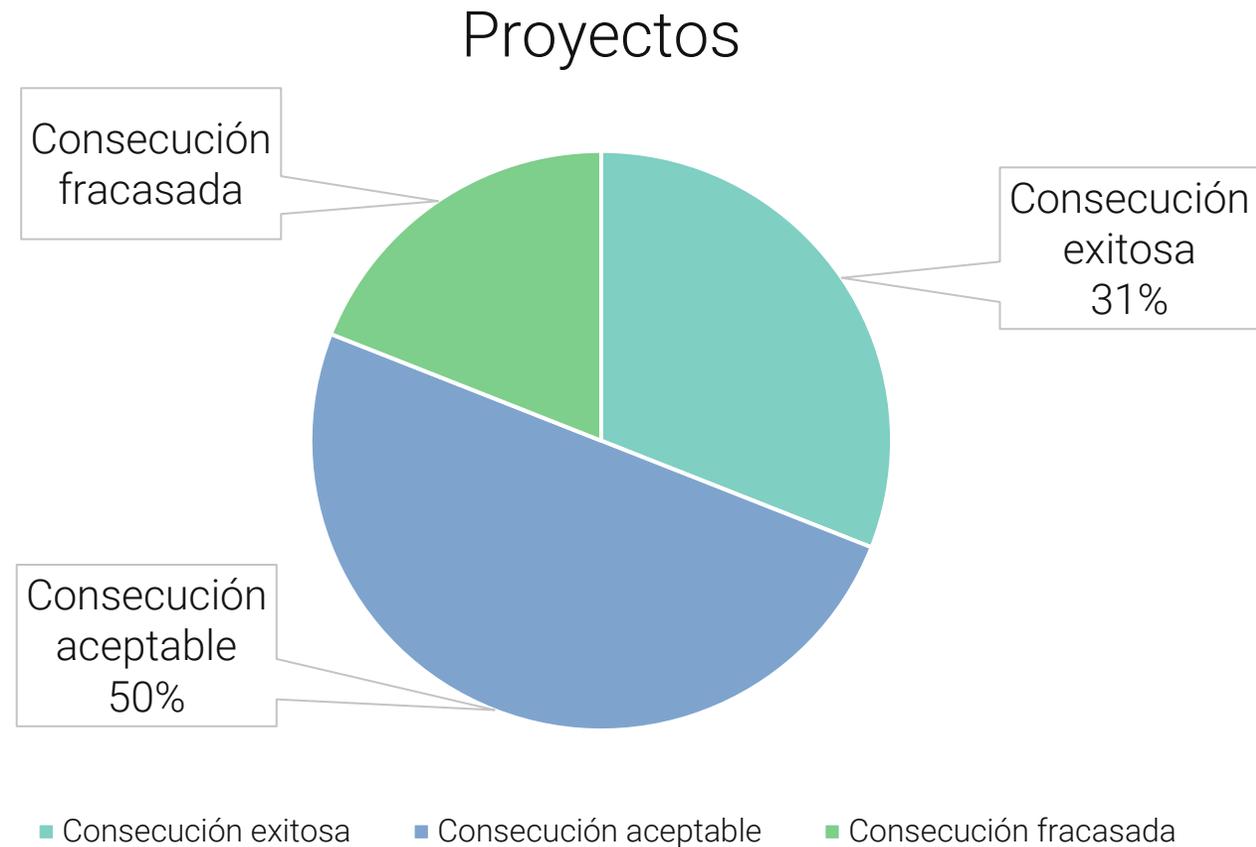
El proceso

Cambio de perspectiva

Valoración basada en:

- Tiempo
- Coste
- Ejecución

03. Scrum | El proceso



Standish Group – Chaos Report 2020

El proceso

Cambio de perspectiva

2/3 de las funcionalidades desarrolladas se utilizan casi nunca o nunca

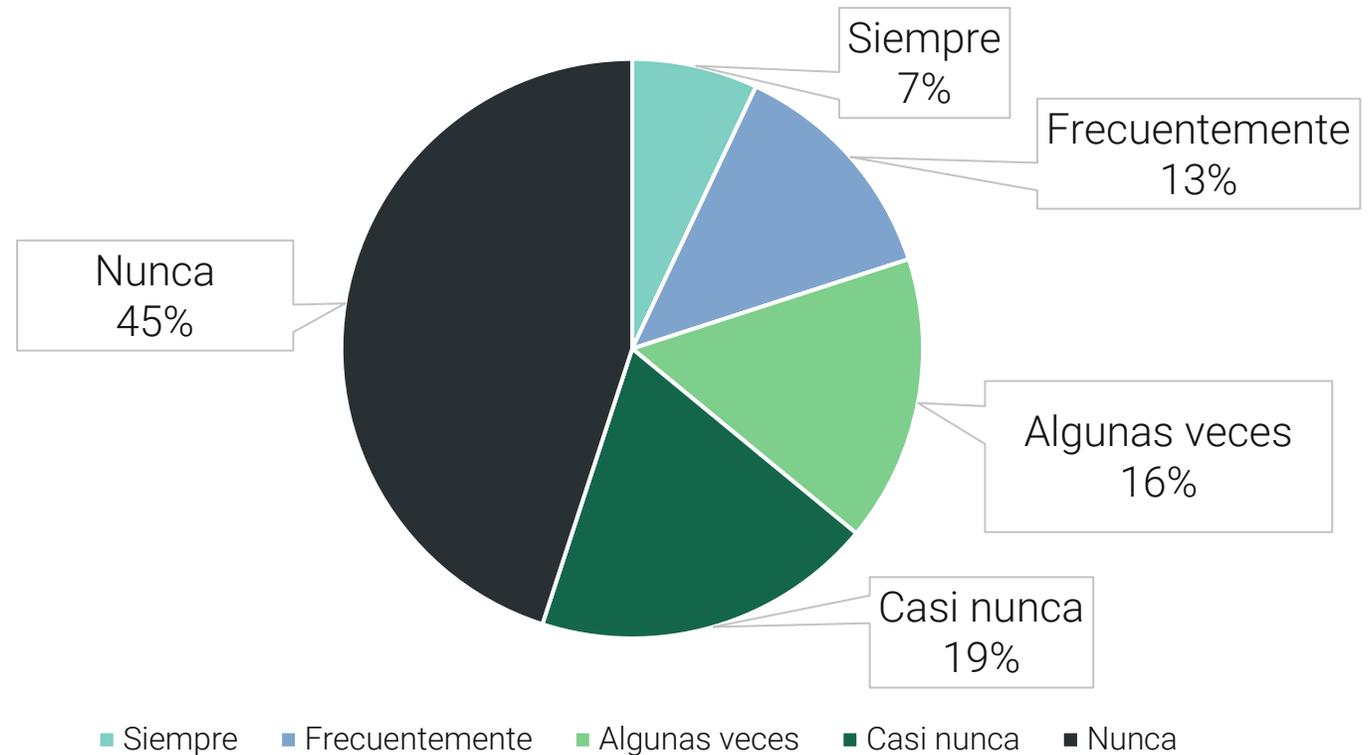
Standish Group – Chaos Report 2002

Los requerimientos cambian en un año

36%

Standish Group – Chaos Report 2012

Frecuencia de uso de las funcionalidades



El proceso

Cambio de perspectiva

Metodologías tradicionales

Fijo

Alcance

Plan en
marcha

Estimado

Coste

Tiempo

Metodologías ágiles

Coste

Tiempo

Valor en
marcha

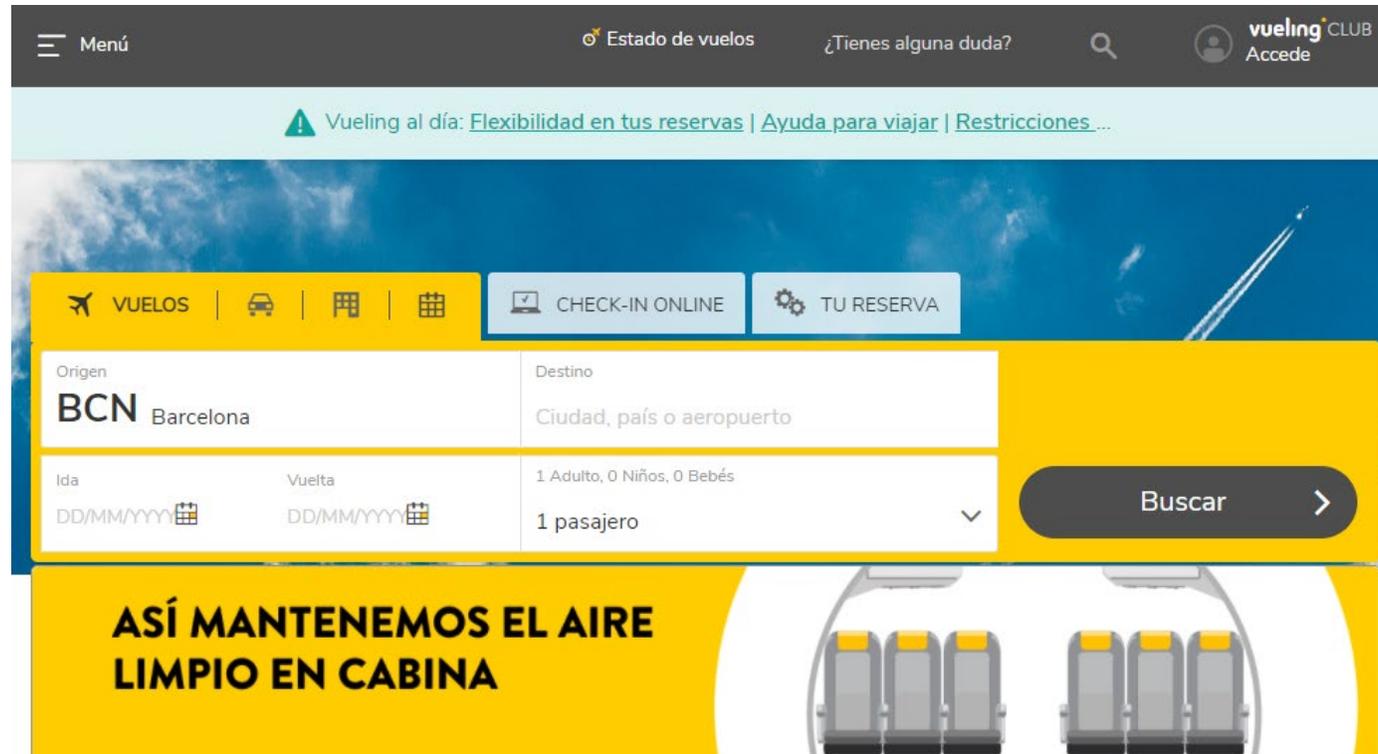
Alcance

El proceso

Cambio de perspectiva

Entrega continua de valor agregado

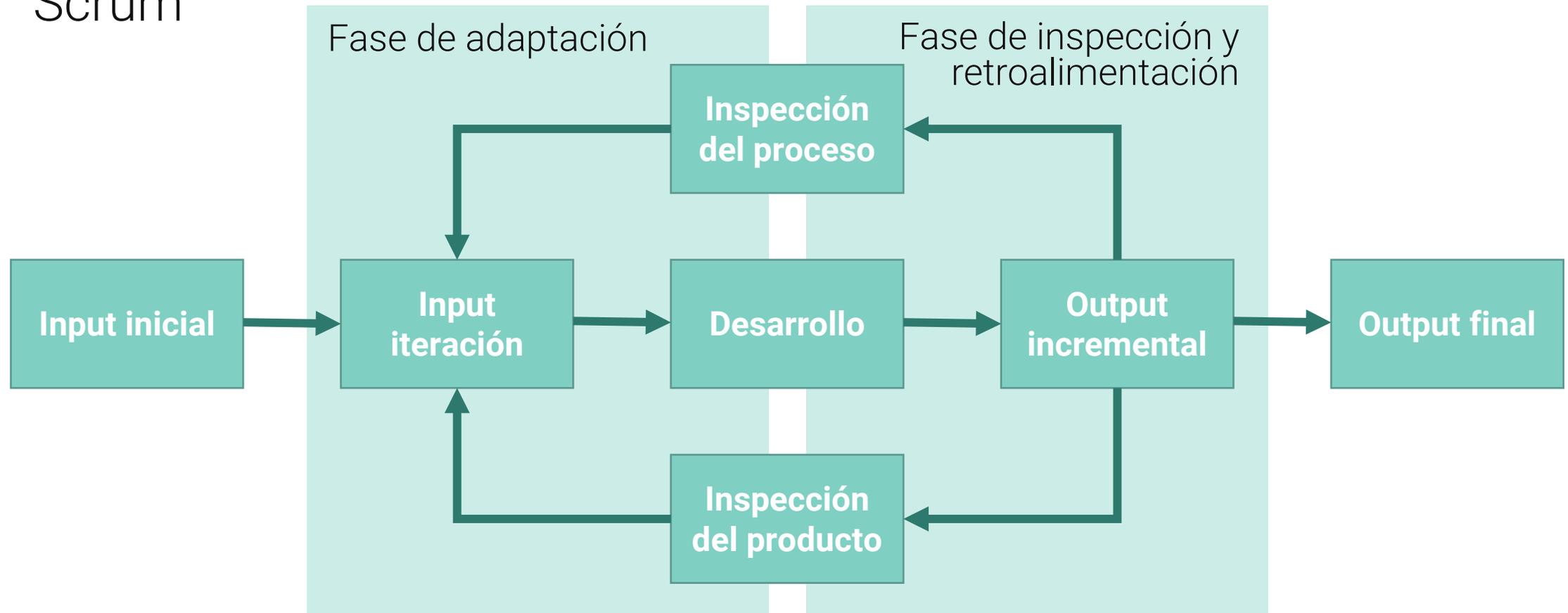
03. Scrum | El proceso



El proceso

Cómo se entrega ese valor agregado de manera continua

Scrum



El sprint

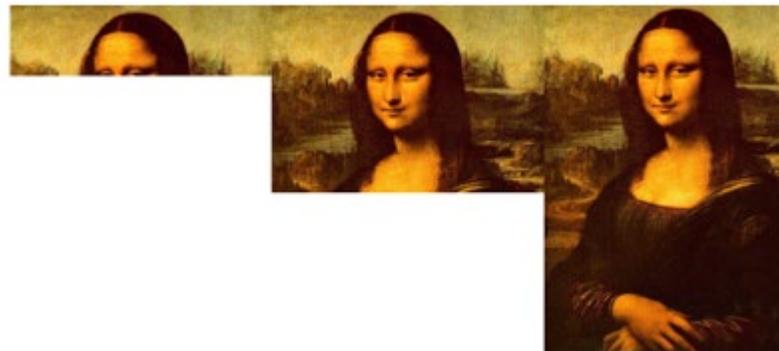
El sprint es **iterativo** e **incremental**

Iterativo



Refina el producto, pero no le añade valor

Incremental



Añadir valor al ¿producto?
¿Es funcional?

[10]

El sprint

El sprint es **iterativo** e **incremental**

Iterativo
+
Incremental



Añade valor a un producto funcional y lo refina el producto

MVP - Minimum Viable Product
(producto viable mínimo)

[10]

El sprint

El sprint es **incremental** e **iterativo**

Proyecto
Programar
una aplicación
para runners

Incremental

1. Añadir gestión de usuario y registro de distancia (MVP)
2. Medir frecuencia cardiaca
3. Reproducir música del dispositivo

Iterativo

1. Login con correo, registrar con GPS del móvil
2. Login con Facebook, registrar con GPS de un Smartwatch
3. Poder agregar amigos, poder reproducir música de tu Spotify

El sprint

El sprint está **protegido**

- En la reunión de planificación del sprint, el **equipo planea** y **se compromete** para la consecución de un objetivo durante ese sprint.
- Finalizada la reunión de planificación del sprint, el **sprint se bloquea**:
 - No se puede modificar el objetivo del sprint.
 - No se pueden modificar los criterios de calidad.
 - No se pueden modificar los criterios de finalización.
 - No se pueden modificar los criterios de aceptación.
- Solo el **product owner** puede **detener/cancelar un sprint**, **no modificarlo**. No es un fracaso, pero debe de evitarse y no abusar de ello.

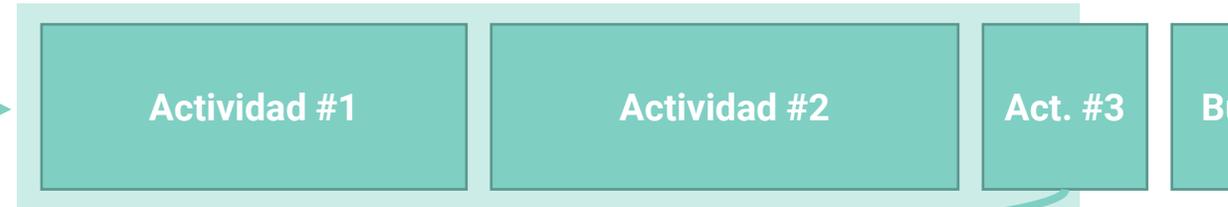
El sprint

El sprint está **acotado temporalmente**

Planificación del Sprint 1



Ejecución del Sprint 1



Planificación del Sprint 2



Ejecución del Sprint 2



Planificación del Sprint 3



Ejecución del Sprint 3



El sprint

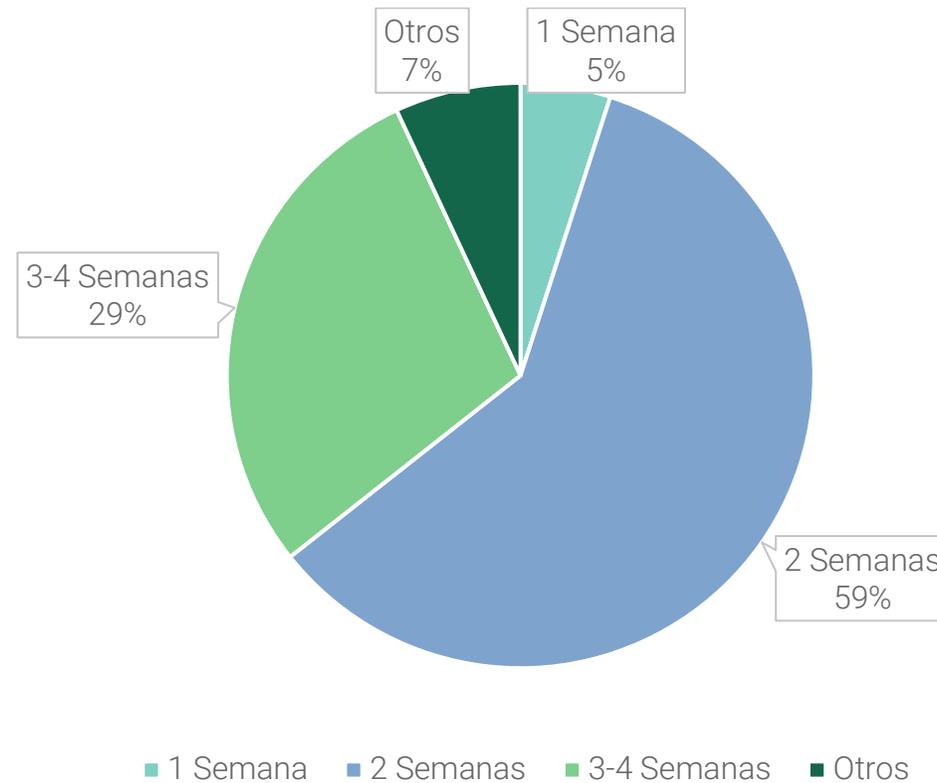
Duración del sprint

03. Scrum | El proceso

Duración (semanas)
promedio de un sprint

2.4

State de Scrum de 2018



El sprint

Criterios de **finalización** y **aceptación**

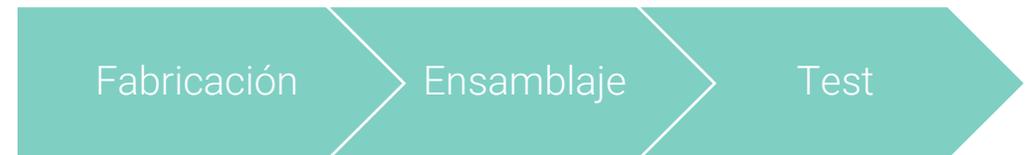


[11]

¿Está finalizado/terminado?

Solo se considera **finalizado** aquello que el **cliente puede utilizar**

Proceso para considerar un coche terminado (simplificado 😊)



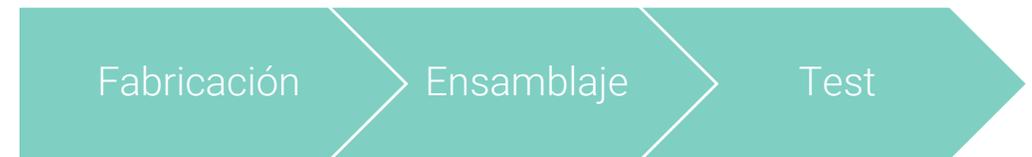
El sprint

Criterios de **finalización** y **aceptación**



[12]

¿Está finalizado/terminado?



Sí, ha seguido el proceso y cliente lo puede utilizar

El cliente quiere un coche de 4 ruedas, ¿lo aceptará?

No

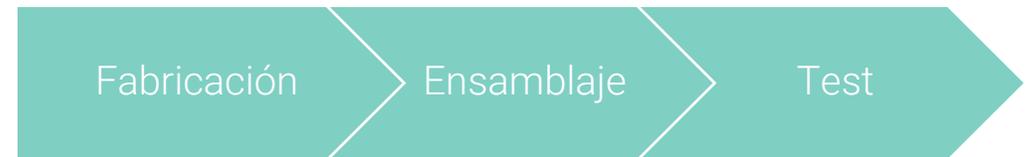
El sprint

Criterios de **finalización** y **aceptación**



[13]

¿Está finalizado/terminado?



Sí, ha seguido el proceso y cliente lo puede utilizar

El cliente quiere un coche de 4 ruedas, ¿lo aceptará?

Sí

¿Es lo que el cliente desea como producto final?

El sprint

Criterios de **finalización** y **aceptación**

[14]



03.

Scrum

- El proceso
- Los roles
- Artefactos
- Reuniones

Los roles

5 valores de Scrum



Foco

Equipo centrado en el sprint



Apertura

Proyecto accesible y transparente



Respeto

(¿Hace falta decir nada más?)



Valor

Defender el método de gestión

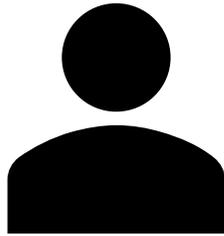


Compromiso

El equipo se compromete a alcanzar los objetivos

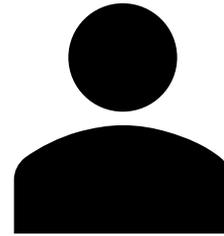
Los roles

El equipo se Scrum



Product owner

- Definir y priorizar los PBI del product backlog
- Toma de decisiones sobre los lanzamiento de producto (el qué y el cuándo)
- Responsable del ROI



Scrum master

- Facilita el proceso de Scrum
- Elimina la amenaza
- Responsable de la efectividad del equipo de Scrum

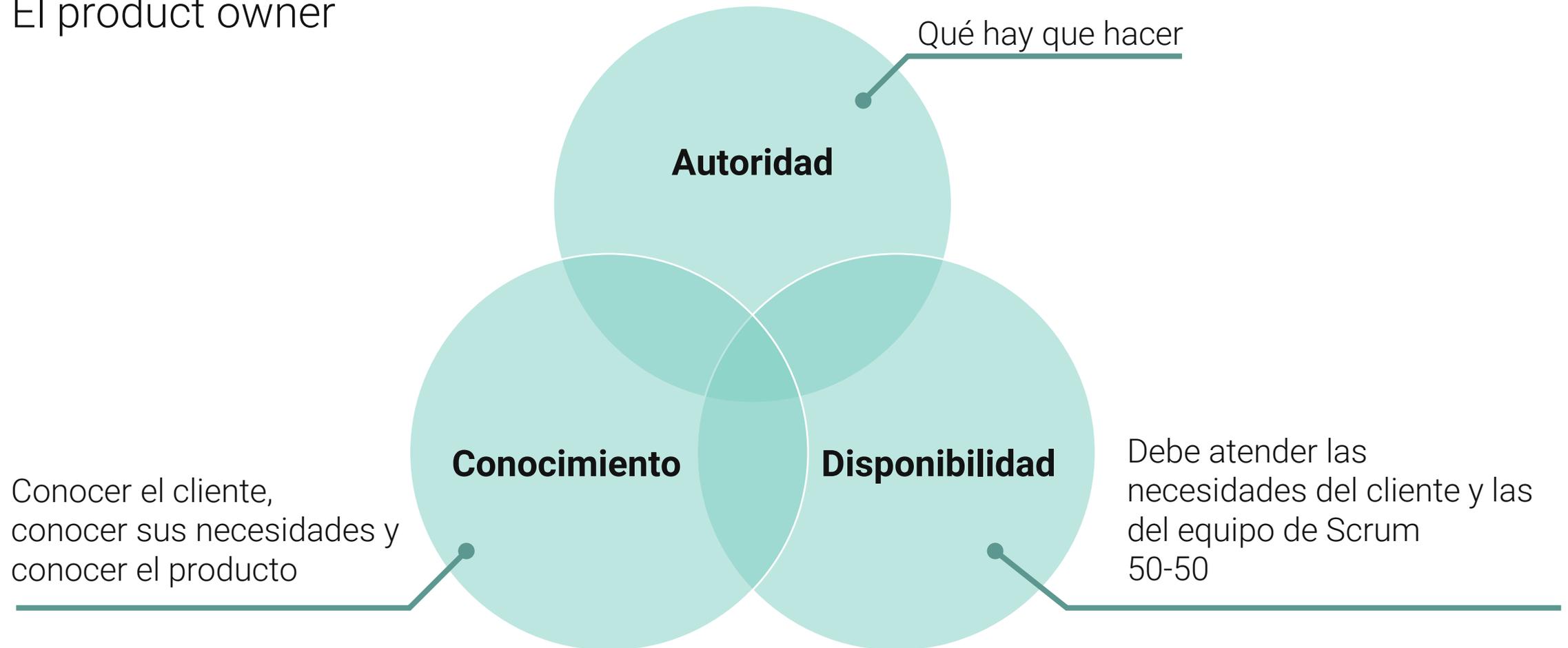


Equipo desarrollo

- Estructurado horizontalmente
- Auto-organizado
- Entre 3 y 9 miembros

Los roles

El product owner



03. Scrum | Los roles

Los roles

El product owner - Responsabilidades

1. Encamina el éxito del producto
2. Crear la visión del producto
3. Crea y mantiene el product backlog
4. Colabora con el equipo
5. Colabora con los stakeholders
6. Participa en las reuniones del sprint

Los roles

El product owner - Responsabilidades

3. Crea y mantiene el product backlog



Claro

Entendible, completo y no ambiguo o inequívoco



Ordenado

Establecer prioridad entre PBIs



Optimizar el valor

Optimizar ratio valor añadido / trabajo



Visible y transparente

Debe de estar disponible al resto del equipo en su totalidad



Entendimiento

Debe de asegurarse que el equipo entiende el product backlog

Los roles

El Scrum master- Responsabilidades

1. Actuar como agente de cambio
2. Está al servicio del resto del equipo de Scrum
3. Elimina los impedimentos y amenazas
4. Entrena al product owner y al equipo
5. Protege al equipo
6. Guía al equipo

Los roles

El Scrum master

El Líder sin autoridad



03. Scrum | Los roles

Los roles

El equipo de desarrollo - Responsabilidades

1. Auto-organizarse y responsabilizarse del desarrollo
2. Entregar un incremento del producto
3. Administrar el Sprint backlog
4. Registrar el progreso del Sprint
5. Participar en las reuniones de Sprint (planificación, revisión, retrospectiva, seguimiento diario)

Los roles

El equipo de desarrollo – La Autoridad

- El equipo **puede tomar cualquier decisión** requerida para **alcanzar el éxito**
- El equipo tiene el poder para **solicitar cualquier recurso** que necesite

03.

Scrum

- El proceso
- Los roles
- Artefactos
- Reuniones

Los artefactos

El product backlog

1. Única fuente de requerimientos

2. Contiene todo lo necesario para cumplir con la visión de producto

3. **Lista** ordenada de características, funciones, requerimientos, mejoras y arreglos

4. Nunca está completo

5. Constante cambio para identificar las necesidades de producto

6. Re-priorizado frecuentemente

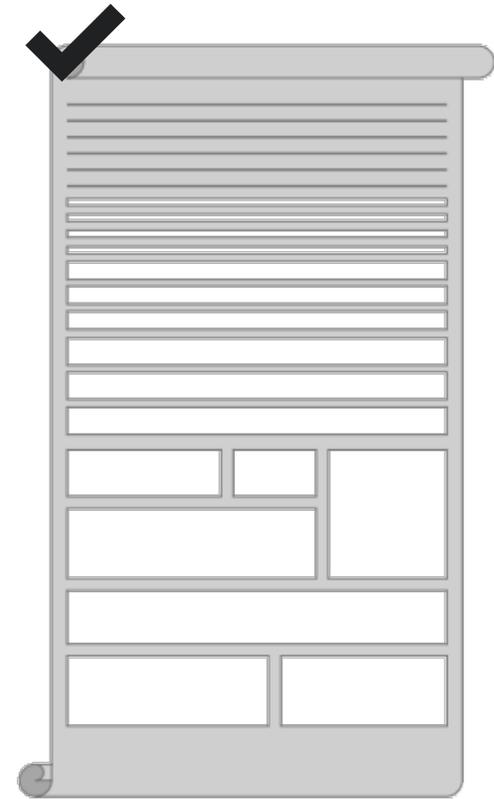
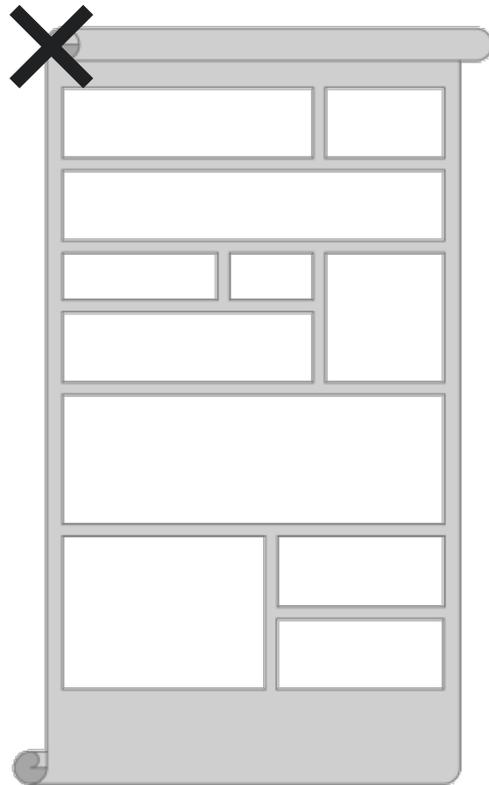
7. Los ítems del backlog se llaman PBIs (Product Backlog Items)
La mayoría de equipos Scrum utilizan Historias de Usuario como PBIs

8. Los PBIs se ordenan por valor de negocio

9. **El product owner es la autoridad final en el orden del backlog**

Los artefactos

El product backlog



Los artefactos

Historias de usuario

1. Una historia de usuario es un requerimiento del producto
2. Una historia del usuario tiene un valor agregado visible para el cliente
3. Cuando se implementa una historia de usuario, se desarrolla una nueva característica que el usuario puede usar
4. NO es una descripción detallada

Esto NO es una historia de usuario

“La aplicación debe de ser programada en Java”

Esto es una historia de usuario

“La aplicación debe permitirme ver los datos”

Los artefactos

Historias de usuario

Como [Rol] quiero [Acción] para que [Valor de negocio]

Como encargado del picking de productos quiero visualizar los pedidos por cliente para poder dar salida de almacén los productos ordenados según cliente y facilitar la operación de logística.

Los artefactos

Historias de usuario

1. Toda historia de usuario debe de ser discutida por el equipo de Scrum cuando el product owner propone su realización durante un determinado sprint
2. El product owner debe resolver las preguntas del equipo de desarrollo
3. Todas las preguntas deben ser escritas en notas
4. Las historias de usuario deben de ser cumplimentadas con los criterios de finalización y aceptación

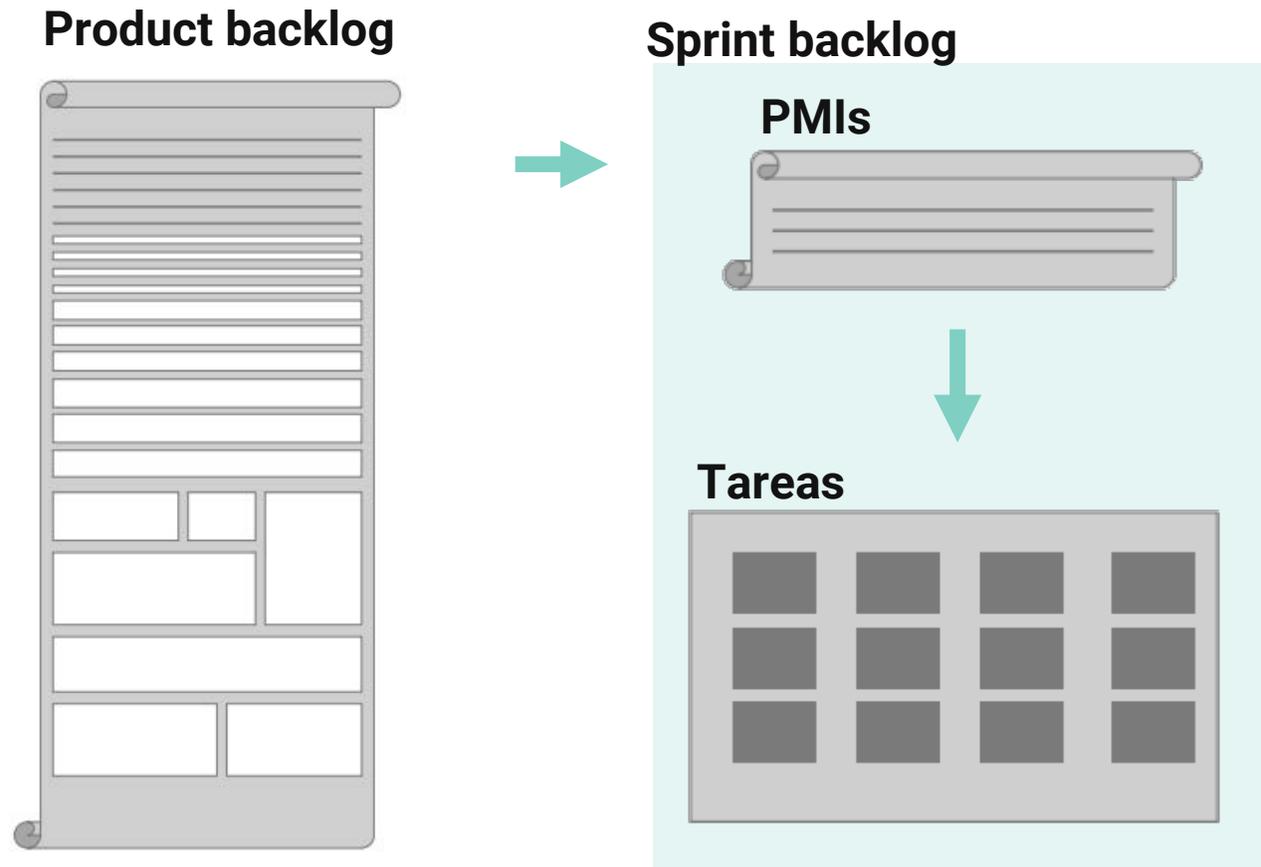
Los artefactos

Historias de usuario – Puntos de historia

- Deben de reflejar la estimación relativa a: esfuerzo, riesgo, complejidad e incertidumbre
- No hacer una estimación en tiempo (las horas no escalan)
- Deben de ser puntos relativos
- Utilizar serie Fibonacci para evidenciar las diferencias entre historias de usuario
- Los puntos no tienen “unidad”
- Los puntos los fija el equipo de desarrollo

Los artefactos

Sprint backlog



Los artefactos

Incremento de producto (entregable)

- El equipo de desarrollo presenta al final de cada sprint un incremento de producto potencialmente entregable
- El incremento es la suma de todos los elementos del product backlog completados durante este sprint y los anteriores
- Tiene que cumplir los criterios de finalización
- Debe de ser aceptado por el product owner

03.

Scrum

- El proceso
- Los roles
- Artefactos
- Reuniones

Reuniones

Reuniones de Scrum

- Planeación del Sprint
- Seguimiento diario
- Revisión del sprint (producto)
- Retrospectiva del sprint (proceso)
- Refinamiento del product backlog (ad hoc)
- Plan de lanzamiento (reléase)

Reuniones

Planeación del Sprint



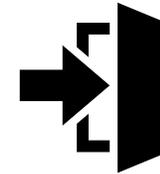
Quién

- El equipo de desarrollo
- El product owner
- El Scrum master



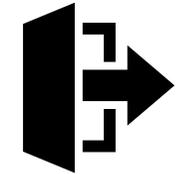
Cuándo/Cuánto

- Al inicio de cada sprint
- Duración máx: 2h / sprint (2w)



Entrada

- Product backlog
- El último incremento del producto
- Capacidad proyectada del equipo de desarrollo durante el sprint
- Desempeño anterior del equipo de desarrollo



Salida

- ¿Qué?
Meta del sprint
- ¿Cómo?
Sprint backlog

Reuniones

Seguimiento diario



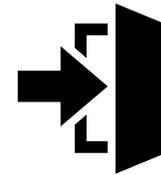
Quién

- El equipo de desarrollo
- El Scrum master



Cuándo/Cuánto

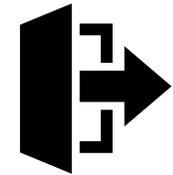
- A la misma hora todos los días, definida por el equipo
- Duración: 15 min



Entrada

Cada miembro contesta a:

- ¿Qué hice ayer?
- ¿Qué haré hoy?
- ¿Veó algún impedimento?



Salida

- Un entendimiento común del trabajo
- Un backlog actualizado con los impedimentos

Reuniones

Seguimiento diario – Tablero Scrum (Kanban)

To do

In progress

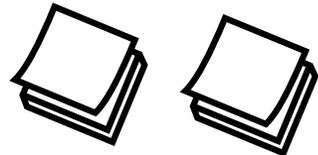
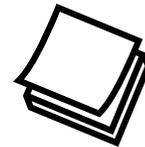
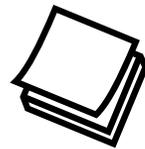
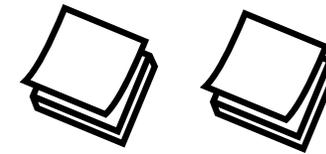
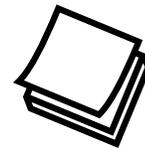
Done

Buffer

Historia 1

Historia 2

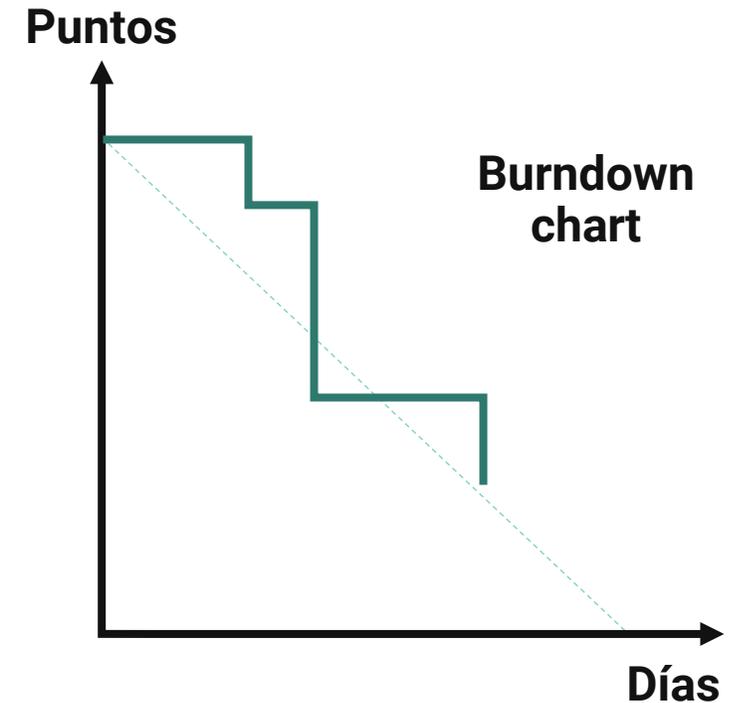
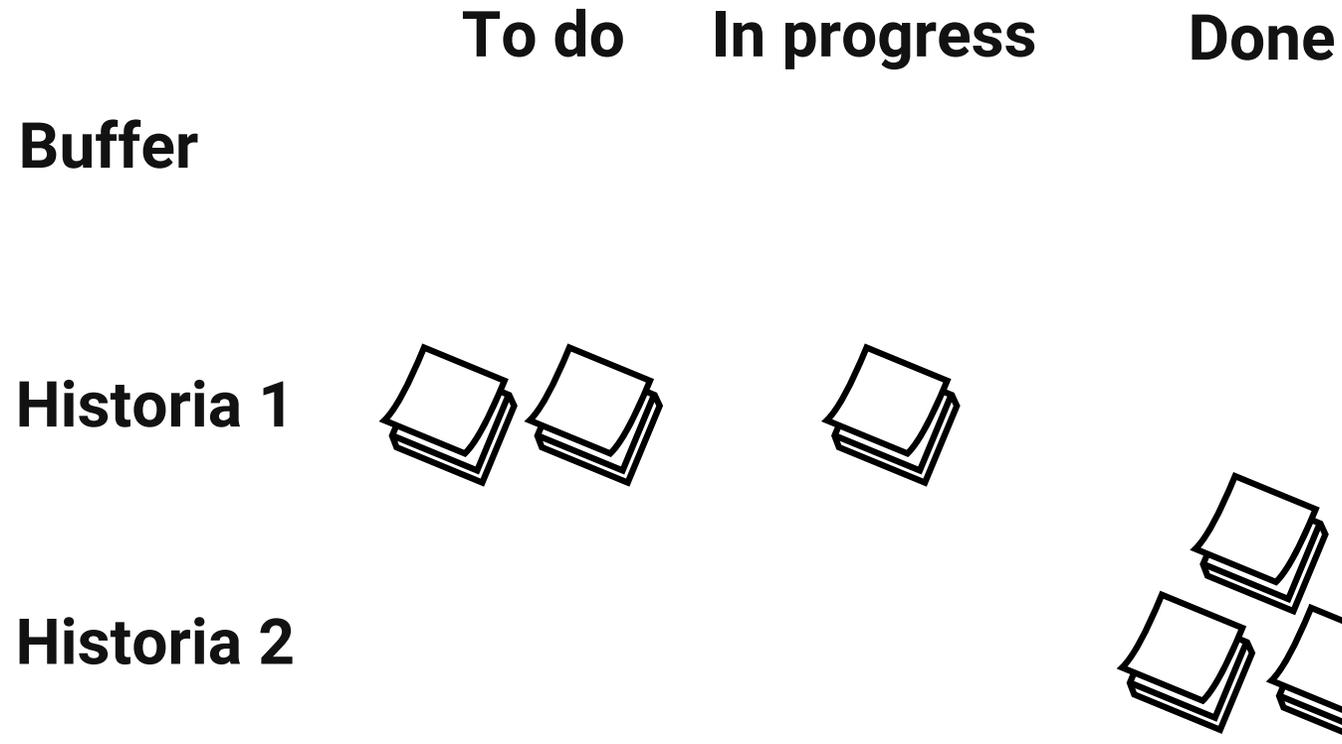
Historia 3



Reuniones

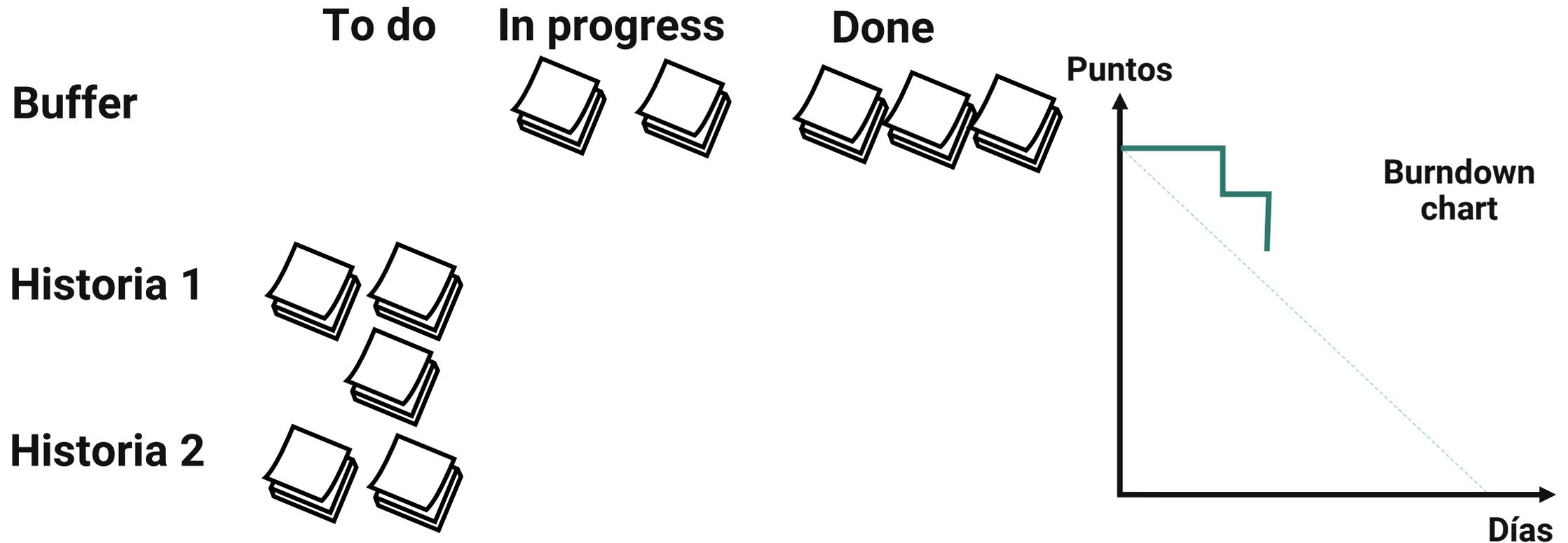
Seguimiento diario – Tablero Scrum (Kanban) – Alarmas (1)

03. Scrum | Reuniones



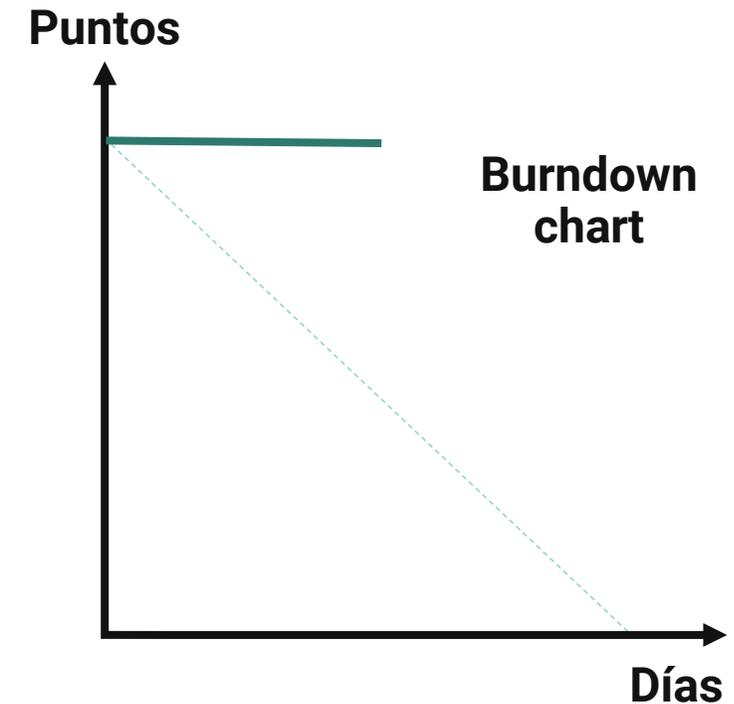
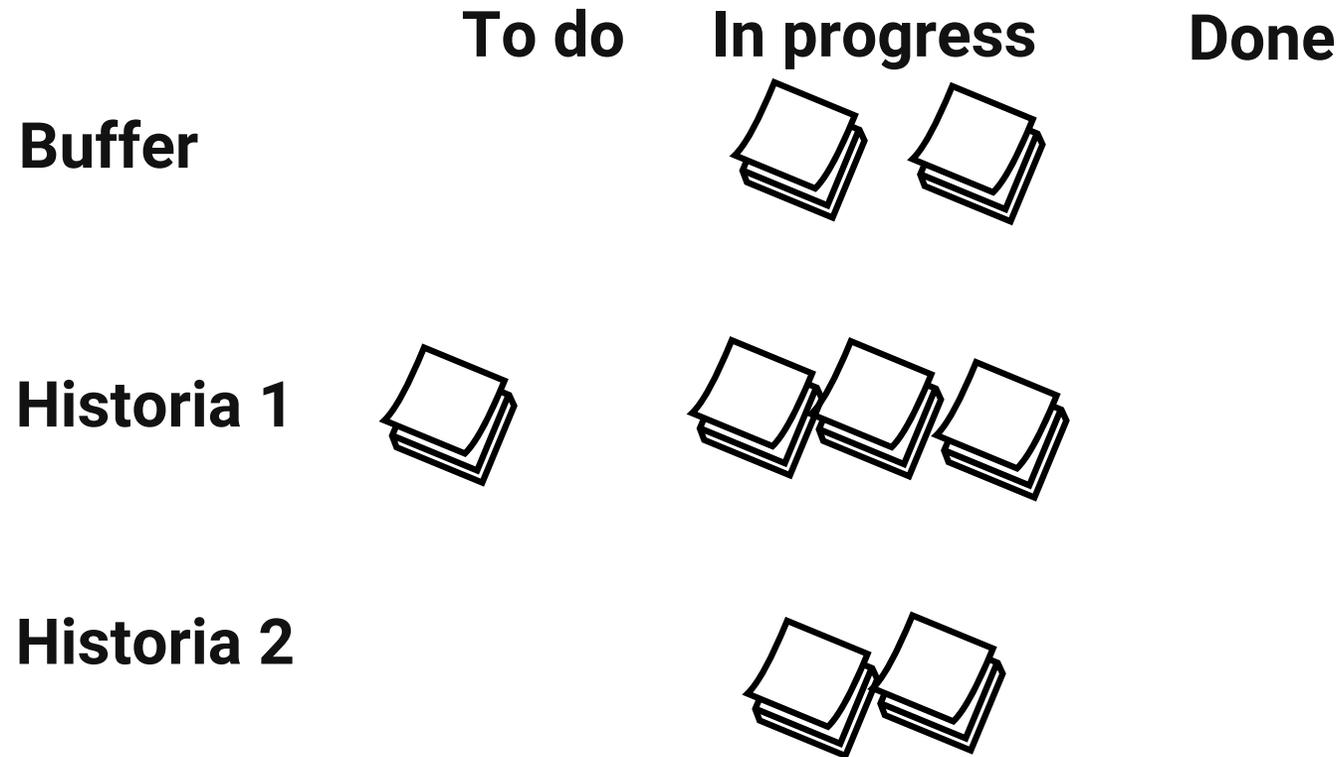
Reuniones

Seguimiento diario – Tablero Scrum (Kanban) – Alarmas (2)



Reuniones

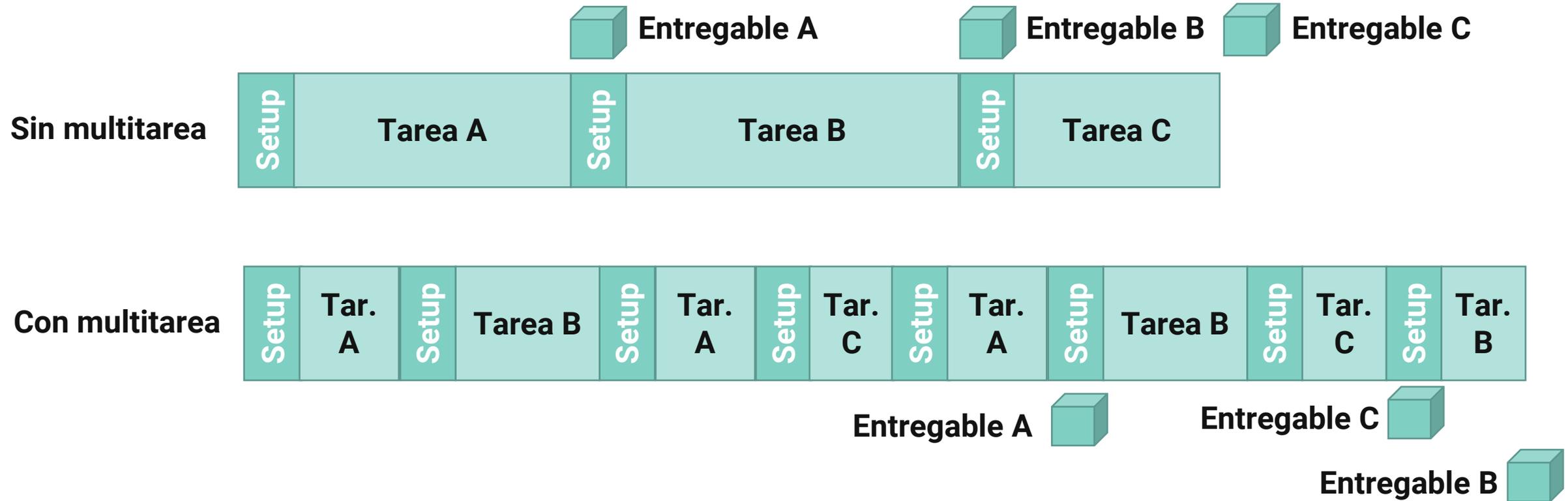
Seguimiento diario – Tablero Scrum (Kanban) – Alarmas (3)



Reuniones

Seguimiento diario – Tablero Scrum (Kanban) – Alarmas (3) - Multitarea

03. Scrum | Reuniones



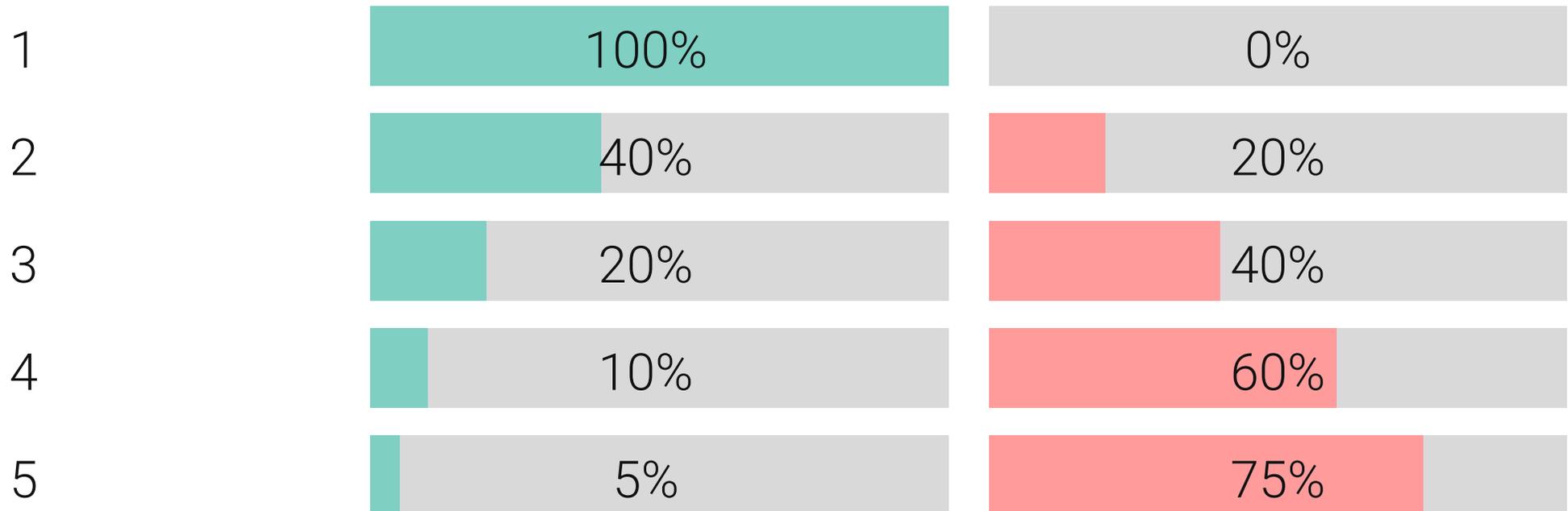
Reuniones

Seguimiento diario – Tablero Scrum (Kanban) – Alarmas (3) - Multitarea

Proyectos simultáneos

Disponibilidad por proyecto

Deficiencias causadas por cambio de contexto



Gerald Weinberg: Quality Software Management

Reuniones

Revisión del sprint



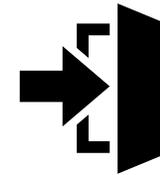
Quién

- El equipo de desarrollo
- El product owner
- El Scrum master
- Stakeholders



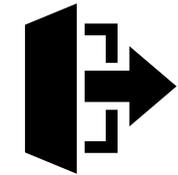
Cuándo/Cuánto

- Al final de cada sprint
- Duración: 1 h / sprint (2w)



Entrada

- Incremento
- Product backlog



Salida

- Incremento potencialmente entregable
- Velocidad
- Retroalimentación al product backlog

Reuniones

Retrospectiva del sprint



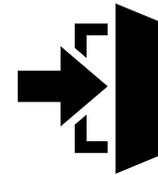
Quién

- El equipo de desarrollo
- El product owner
- El Scrum master



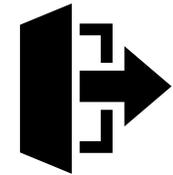
Cuándo/Cuánto

- Después de la revisión de sprint
- Duración: 0.75 h / sprint (2w)



Entrada

- Información de los equipos acerca del último sprint



Salida

- Qué salió bien
- Mejoras potenciales
- Plan de mejoras

Reuniones

Refinamiento del product backlog



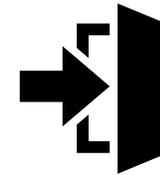
Quién

- El equipo de desarrollo
- El product owner



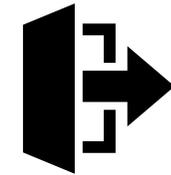
Cuándo/Cuánto

- Cuando sea necesario
- Duración: max. 10% del tiempo de sprint



Entrada

- Visión del producto
- Product backlog
- Velocidad



Salida

- Product backlog refinado

Reuniones

Plan de lanzamiento



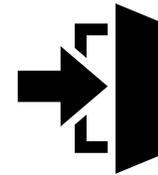
Quién

- El equipo de desarrollo
- El product owner
- El Scrum master
- Stakeholders



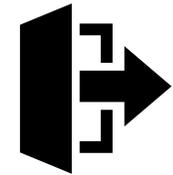
Cuándo/Cuánto

- Cuando sea necesario
- Duración: no hay regla



Entrada

- Visión del producto
- Product backlog
- Meta de lanzamiento
- Velocidad

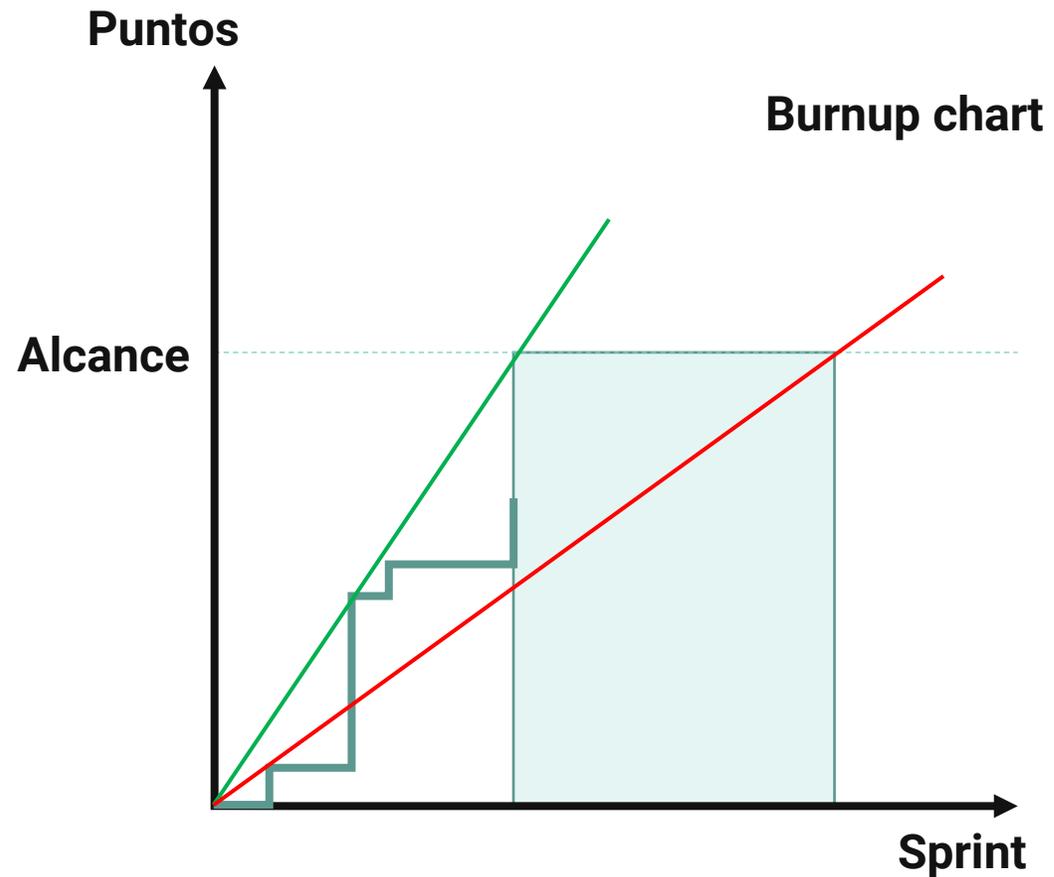


Salida

- Plan de lanzamiento

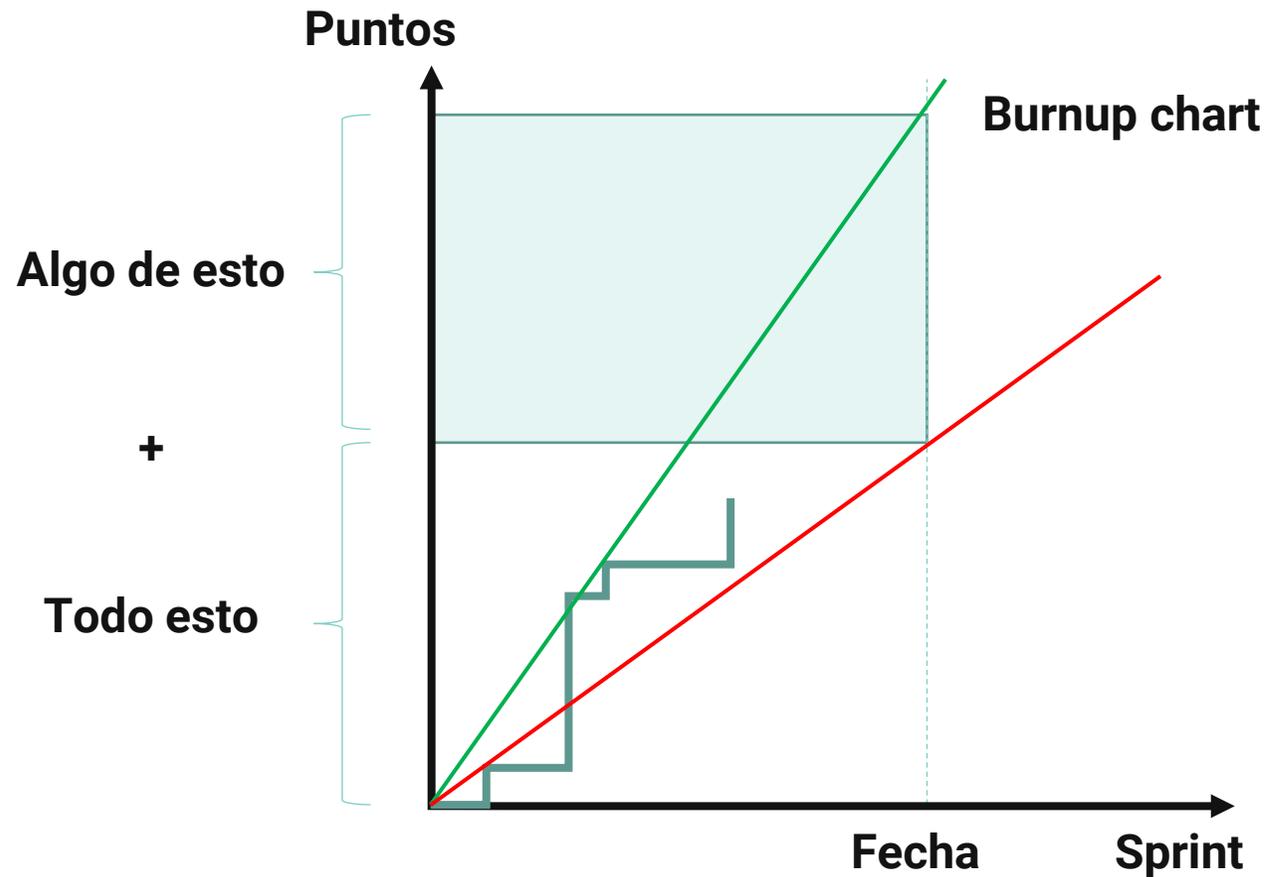
Reuniones

Plan de lanzamiento – Alcance fijo



Reuniones

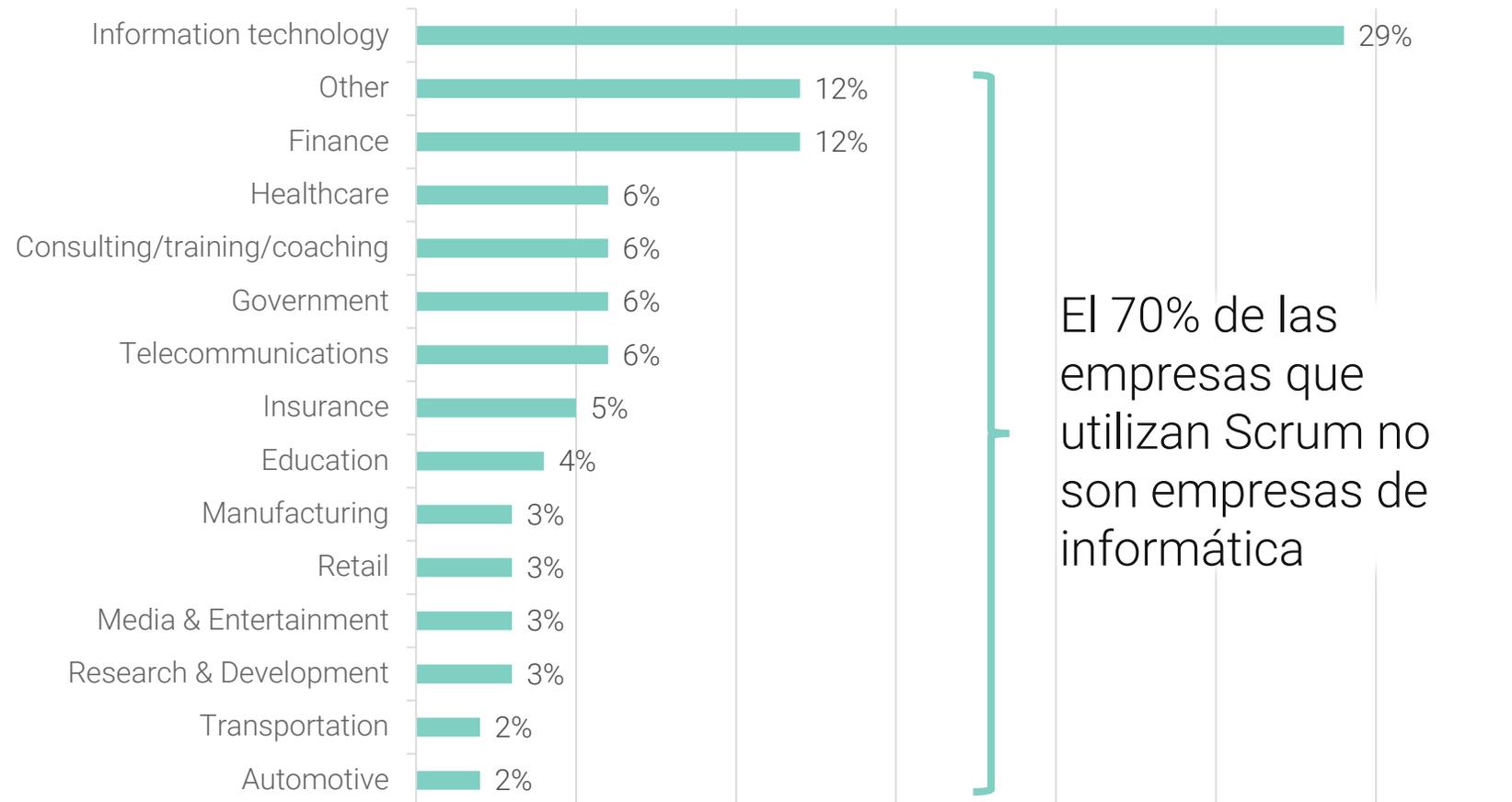
Plan de lanzamiento – Fecha fijo



03. Scrum | Reuniones



“Scrum es solo para los informáticos”



+1. Para los increídos...

State of Scrum 2015

Imágenes utilizadas en la presentación

- [1] Hitotsubashi University. (2017). Knowledge management pioneer Ikujiro Nonaka of Hitotsubashi University to receive Lifetime Achievement Award from Haas School of Business, UC Berkeley. Title of the artwork [Online image]. EurekaAlert! <https://www.eurekaalert.org/multimedia/pub/145702.php>
- [2] World Economic Forum from Coligny, Switzerland. World Economic Forum Annual Meeting Davos 2009 [Online image]. (2009). Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Hiroataka_Takeuchi_-_World_Economic_Forum_Annual_Meeting_Davos_2009.jpg
- [3] Scrum Guide Update with Ken Schwaber and Jeff Sutherland [Online image]. (2017). Scrum.org. <https://www.scrum.org/resources/2017-scrum-guide-update-ken-schwaber-and-jeff-sutherland>
- [4] USAF (1968). RF-4C Phantom 14th TRS landing at Udorn RTAFB in 1968 [Online image]. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:RF-4C_Phantom_14th_TRS_landing_at_Udorn_RTAFB_in_1968.jpg
- [5] Garrybooker, Malyszczk, Mario Fèvre. (2011). Gantt chart anatomy [Online image]. Wikimedia Commons. <https://commons.wikimedia.org/wiki/File:GanttChartAnatomyES.svg>
- [6] Jeff Sutherland and VersionOne Present Webinar on the Power of Scrum [Online image]. (2014). Yahoo! Finance. <https://finance.yahoo.com/news/jeff-sutherland-versionone-present-webinar-151800912.html>
- [7] Bruce Lee and his teacher Ip Man. (1958). Wikimedia Commons. https://commons.wikimedia.org/wiki/File:The_age_of_18_Bruce_Lee_and_Ye_Wen.jpg
- [8] Bruce Lee training. Pinterest. <https://www.pinterest.com/pin/417357090439484346/>
- [9] Lee, B. (1975). Tao of jeet kune do. Santa Clarita, CA: Ohara Publications.
- [10] Da Vinci, L. (1519). La Gioconda [Paint]. París, Museo de Louvre.
- [11] Jens Meyer. (2020). Ensamblaje de un coche Golf del fabricante alemán Volkswagen en Zwickau, Alemania [Online image]. El País. https://elpais.com/economia/2020/02/06/actualidad/1581014300_282217.html
- [12] Coche tres reudas Reliant Robin. Periodismo del motor. <https://periodismodelmotor.com/coches-tres-ruedas/152508/>
- [13] Coche burbuja. Your parking space. <https://www.yourparkingspace.co.uk/insights/the-top-10-strangest-cars-ever>
- [14] Lamborghini Centenario. (2016). Car magazine. <https://www.carmagazine.co.uk/car-news/motor-shows-events/geneva/2016/new-lamborghini-centenario-unveiled-at-the-2016-geneva-motor-show/>

INTRODUCCIÓN

Metodologías ágiles

Albert Álvarez Carulla

05.03.2021

TheAlbert.dev



@thealbertdev



"Introducción a las metodologías ágiles" por Albert Álvarez Carulla se distribuye bajo una [Licencia Creative Commons Atribución-NoComercial-SinDerivadas 4.0 Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/)