

# Introducción al desarrollo ágil

## SCRUM

# Alguna experiencia ágil?

Has podido implantar algún aspecto  
ágil en vuestro trabajo?

# Organización "improvisación"

Las personas por encima de los procedimientos y las herramientas

# Organización "disciplinada"

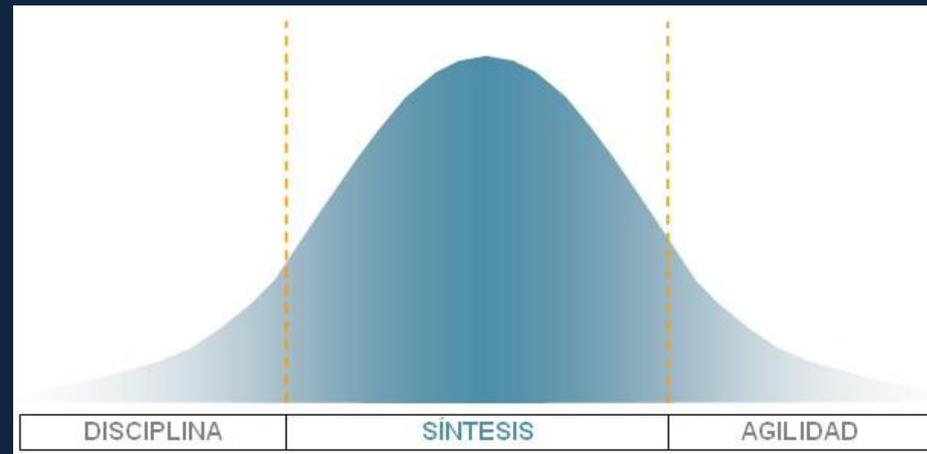
Las personas se coordinan mediante procedimientos y herramientas

# Síntesis

Los procedimientos evolucionan y se especializan. Las personas no solo "ejecutan" sino que aportan y cuidan el **conocimiento de la organización**

# Organización ágil

Se aplican métodos ágiles en organizaciones con voluntad de evolucionar sus procedimientos de trabajo





# Agile Manifesto

## Manifiesto para el desarrollo ágil de software

Estamos poniendo al descubierto mejores formas de desarrollar software  
Haciendolo y ayudando a otros a hacerlo. Mediante este trabajo hemos acabado  
valorando:

**Individuos e interacciones** per encima de procesos y herramientas

**Software que funciona** por encima de documentación exhaustiva

**Colaboración con el cliente** por encima de negociación de contratos

**Respuesta al cambio** por encima de ceñirse a una planificación

Es decir, aunque elementos de la derecha tienen valor,  
nosotros valoramos mas 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

# Agile Manifesto

Martin Fowler

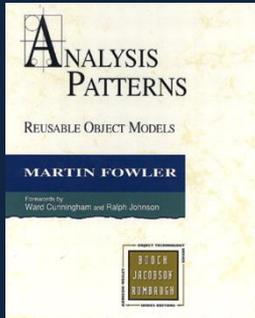
UK, 1963

Especialidad en análisis y diseño en POO

UML

Patrones de diseño

Metodologías ágiles: XP



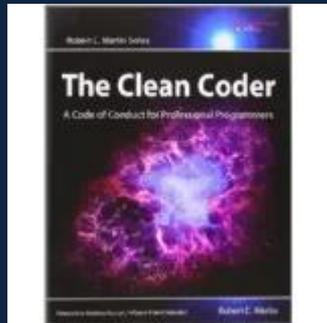
# Agile Manifesto

Robert Cecil Martin

EEUU, 1952

Ingeniero de software

Muy especializado en técnicas ágiles de programación, UML y patrones de diseño



# Agile Manifesto

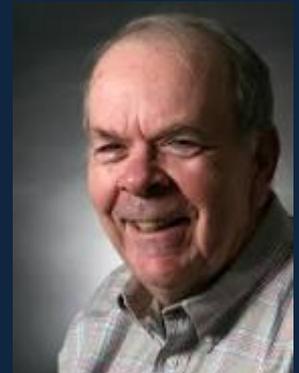
Jim Highsmith

EEUU, 1945

Especialidad en metodologías de desarrollo de software

Creador de ASD (1999): *Adaptive Software Development*, (Jim Highsmith I Sam Bayer)

Creador de un modelo en contraposición al proceso tradicional en cascada, basado en la colaboración



# Agile Manifesto

Kent Beck

EEUU, 1961

Ingeniero de software

Tarjetas CRC

Pruebas Unitarias jUnit

Creador de eXtreme Programming, (XP) y  
de Test-Driven Development (TDD)



# XP Programming

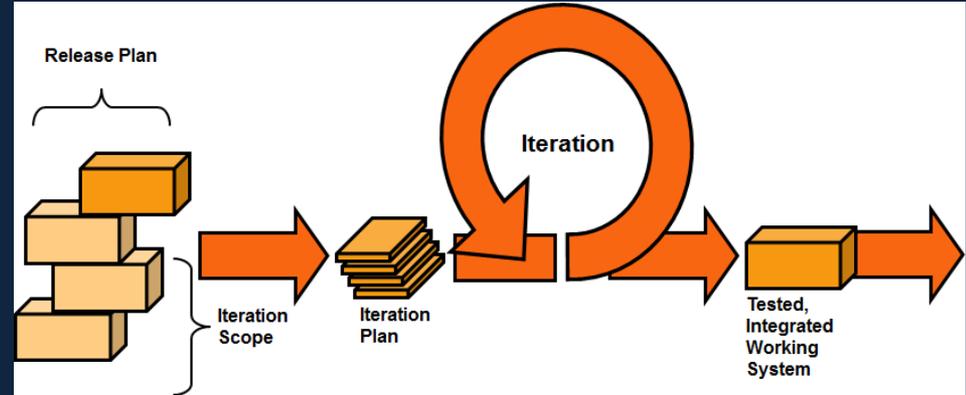
XP es una de las técnicas de programación ágil más conocidas y más maltratadas de todos los tiempos.

En esencia transmite los mismos principios que SCRUM:

- **Simplicidad**
- **Comunicación**
- **Realimentación**
- **Coraje**
- **Respeto**

SCRUM = Gestión

XP = Buenas prácticas en el desarrollo



Normas del XP Programming:

- Desarrollo iterativo e incremental
- Pruebas unitarias continuas
- Programación en parejas
- Integración del equipo con el cliente
- Corrección de todos los errores SIEMPRE
- Refactorización de código SIEMPRE
- Propiedad del código compartida
- Simplicidad en el código

# Agile Manifesto

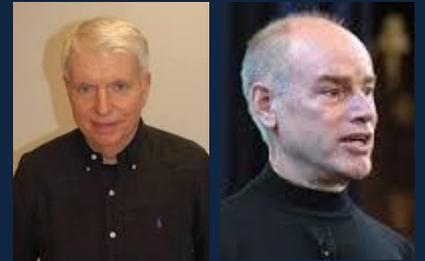
Ken Schwaber i Jeff Sutherland

Ken Schwaber: EEUU, 1945

Jeff Sutherland: EEUU, ????

Desarrolladores de software

Adaptación de SCRUM y principios ágiles  
(1995) de la versión original (1986)



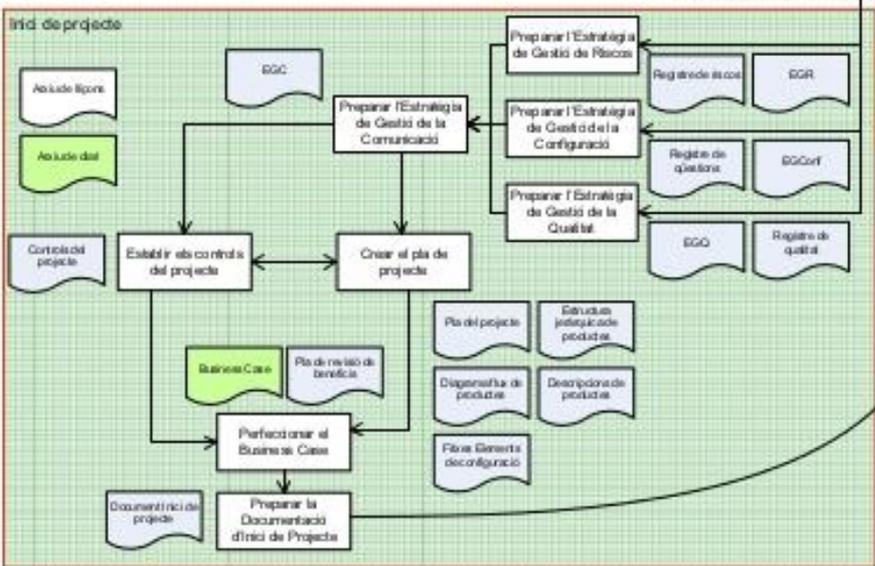
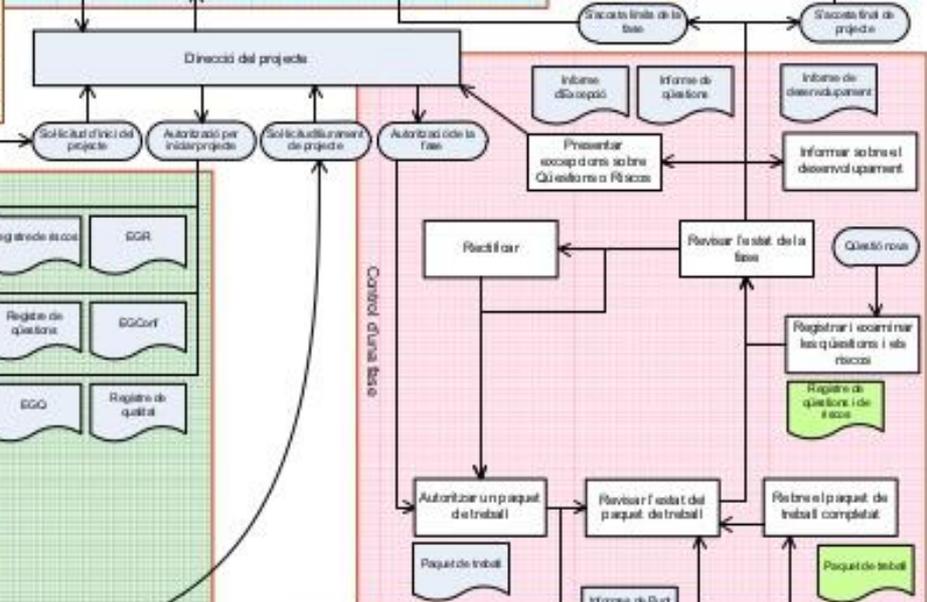
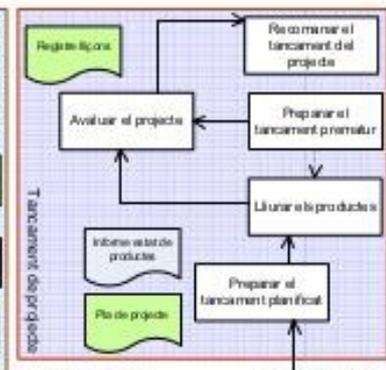
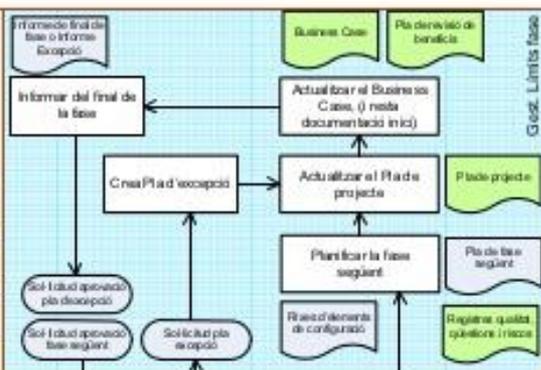
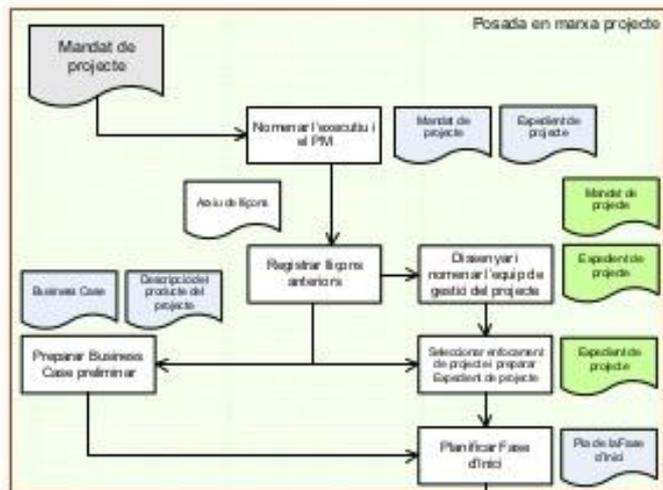
# Los SCRUM **no** ...

1. Aplicar SCRUM **no** es prescindir de la **documentación** (doc profesional, enfocada a esquema, modelo e iterativa)
2. Aplicar SCRUM **no** significa **prescindir de definir el alcance** antes de iniciar el proyecto
3. Aplicar SCRUM **no** significa prescindir de las **comunicaciones formales**, (siguen siendo útiles actos y documentación de acuerdos)
4. Adaptarse al cambio **no** significa resistirse al cambio con procedimientos o con herramientas
5. Colaborar con el cliente **no** es “si a todo”

# SCRUM **no** es una metodología

Es un **marco de trabajo** (framework)

*Para poder decir que **haces SCRUM**, como mínimo has de cumplir:  
(Transparencia, Inspección, adaptación y Mejora continua) + (Daily Meeting, Time Box, Sprint)*



Proyecto

# Proyecto

**complejo, incierto,  
cambiante**

# Proyecto



- **Acotado en el tiempo**
- **Controlado en recursos**
- **Definido en el alcance**

Datos, objetivos y decisiones

No sólo los económicos

Objetivos claros

SCRUM no da una definición  
de Proyecto

# SCRUM no da una definición de Proyecto

Por encima del “plan” está el producto

# SCRUM

Definido por Hirotaka Takeuchi y Ikujiro Nonaka en 1986 como aproximación al desarrollo de productos de forma general, haciendo énfasis en la rapidez y la flexibilidad

# SCRUM está basado en la teoría del **control de procesos empíricos**

*Wikipèdia: El empirismo es una teoría filosófica que enfatiza el papel de la experiencia, ligada a la percepción sensorial, en la formación del conocimiento*

# En que se caracteriza el Control de procesos empíricos?

3 pilares que definen el empirismo:

- Transparencia
- Inspección
- Adaptación

+ concepto de **Mejora continua**

# 3 pilares

## Transparencia

La información “ha de fluir”.  
Se ha de hablar “el mismo idioma”

Los aspectos significativos del proceso han de ser conocidos por todos los participantes. Esto requiere que estos aspectos han de estar definidos mediante un estándar común, de forma que todos tengan la misma percepción de las características de cada aspecto. (por ejemplo: definición de “acabado”)

# 3 pilares

## Inspección



(también del proceso mismo) **Proyecto = Cazar desviaciones**

Todo proceso persigue un objetivo. Para la consecución de este objetivo es necesario que los participantes en el proceso evalúen de forma continua sus resultados **y el proceso mismo**, para detectar posibles desviaciones del objetivo lo mas pronto posible

# 3 pilares

## Adaptación

Que hacemos cuando detectamos una desviación? Nos **adaptamos**

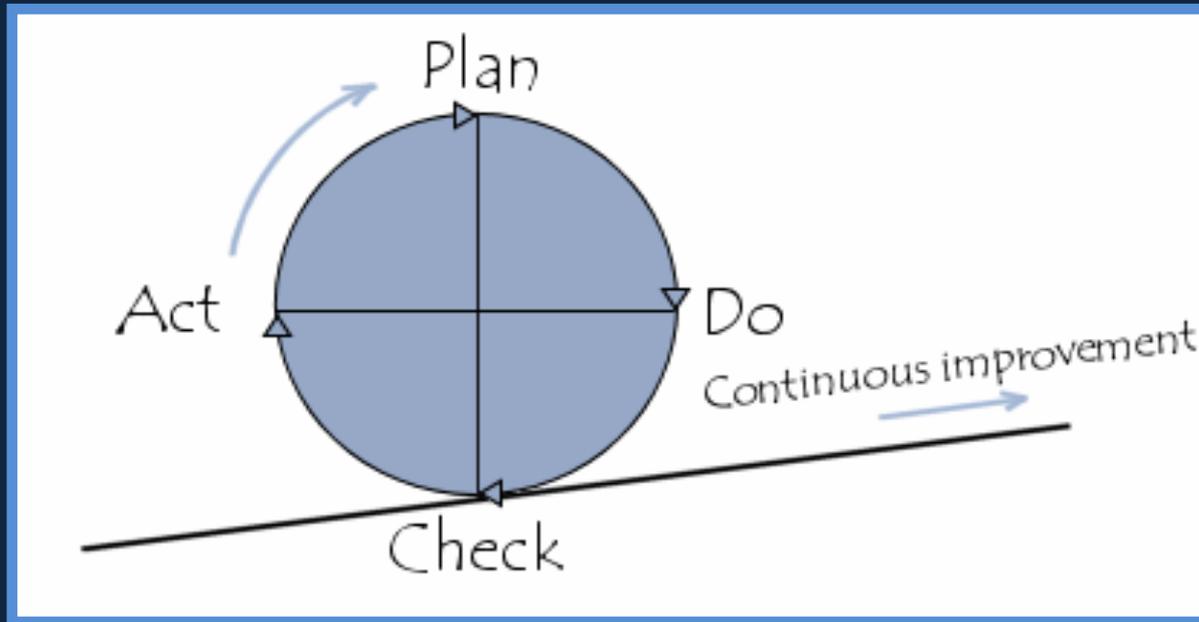
Adaptarse es:

1. Crear un plan para corregir la desviación
2. Cambiar los objetivos afectados

Cuando se detecta una desviación, la respuesta a esta desviación ha de ser la adaptación, es decir, la adopción de acciones o planes que, o bien ayuden a corregir la desviación, o bien reconfiguren el objetivo

SCRUM es

# Mejora continua (actitud)



# Principios de SCRUM

# Principios de SCRUM

- Satisfacción del cliente
- Receptividad ante el cambio de requerimientos
- Trabajo enfocado al producto o servicio
- Desarrollo sostenible
- Cooperación diaria y abierta entre negocio y desarrolladores
- Comunicación directa persona a persona
- Individuos motivados frente individuos dirigidos
- Orientación a la excelencia
- Simplicidad
- Equipos auto-organizados
- Adaptabilidad

# Valores de SCRUM

# Valores de SCRUM

**Compromiso, (commitment):** Para trabajar en equipo es necesario un alto grado de compromiso

**Enfoque, (focus):** Dividir el problema en partes mas pequeñas que nos permitan concentrarnos en la resolución de un único problema asumible para el equipo.

**Organización abierta, (Openness):** De forma continua expresemos con el equipo como nos encontramos y que estamos haciendo para trabajar en equipo. Aprendemos de los demás. Pedimos ayuda. Ofrecemos ayuda.

**Respeto:** Con el compromiso y el trabajo en equipo llegamos a respetar nuestro trabajo y el trabajo de los demás

**Coraje:** El trabajo en equipo y el respeto nos da el que necesitamos para afrontar los retos de proyectos complejos e inciertos

# Organizaciones de SCRUM



Improving the Profession of Software Development



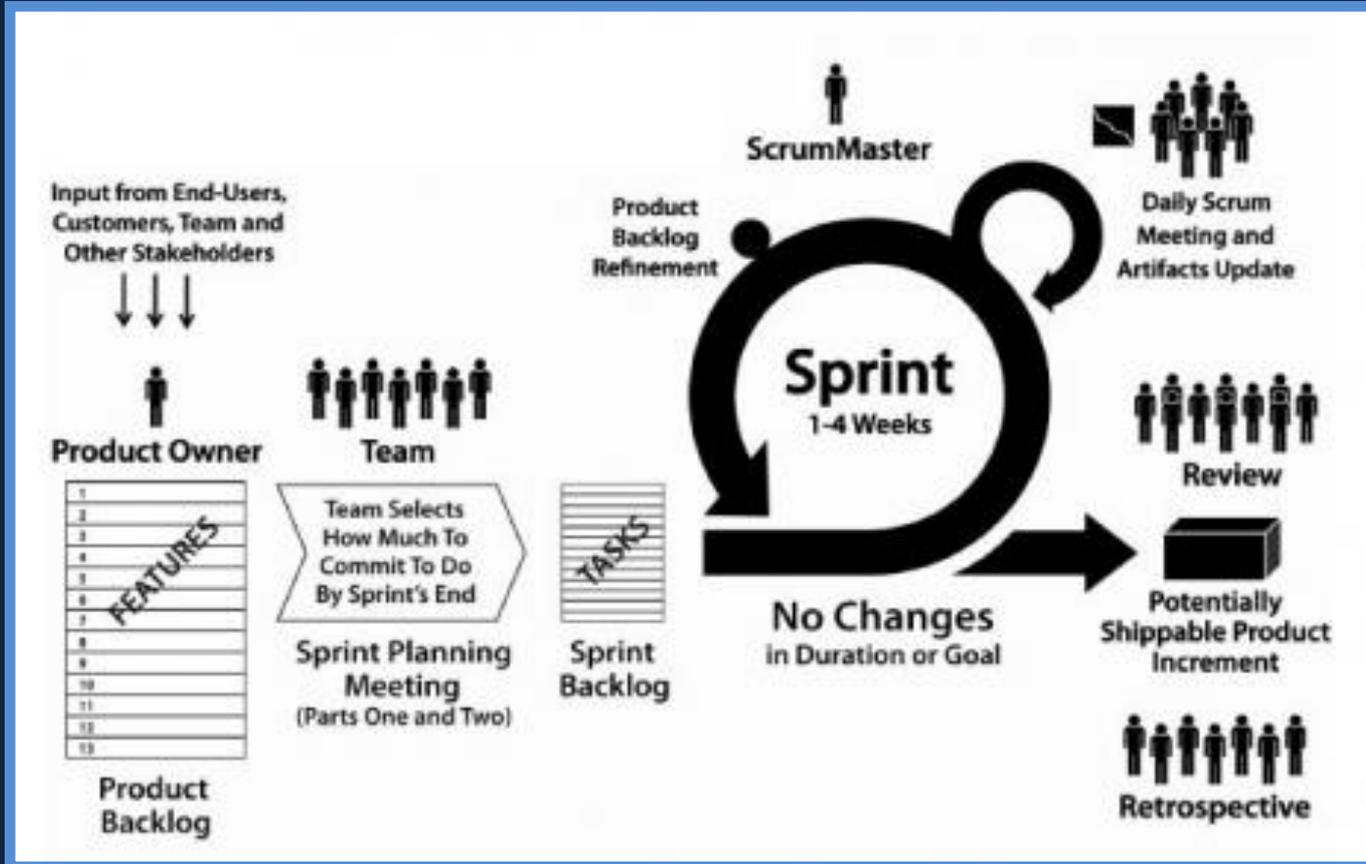
Scrum.org, (<https://www.scrum.org/>)

Scrum Alliance, (<http://www.scrumalliance.org/>)

European Scrum, (<http://www.europeanscrum.org/>)

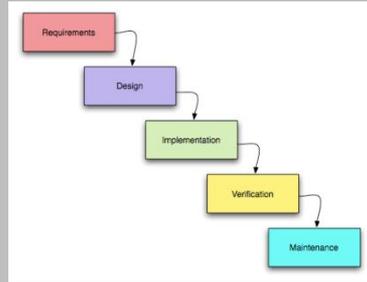
# Organización de SCRUM

# Organización de SCRUM



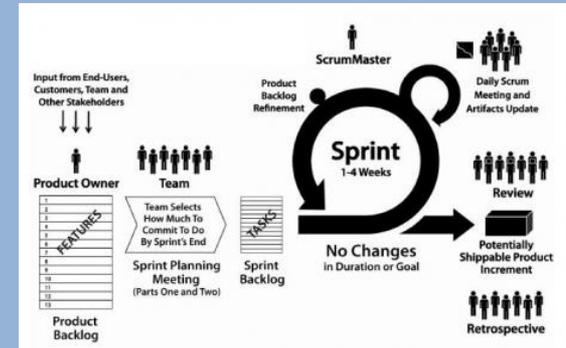
# Organización de SCRUM

## Modelo tradicional, (cascada)



- Predictivo
- Relay race, (carrera de relevos)
- Organización jerárquica
- Departamental
- Objetivos completos
- Controlado en Tiempo, presupuesto, alcance y calidad

## SCRUM



- Adaptativo
- Holístico . Deporte de equipo
- Aproximación Matricial
- Autogestionado
- Entregas incrementales. Aportación continua de valor
- Controlado en tiempo, presupuesto, alcance, calidad y expectativas → El cliente colabora

# Roles, artefactos, actividades

- Personas
- Herramientas
- Flujo

# Roles, artefactos, actividades



Product Owner



Development Team



Scrum Master



Stakeholders

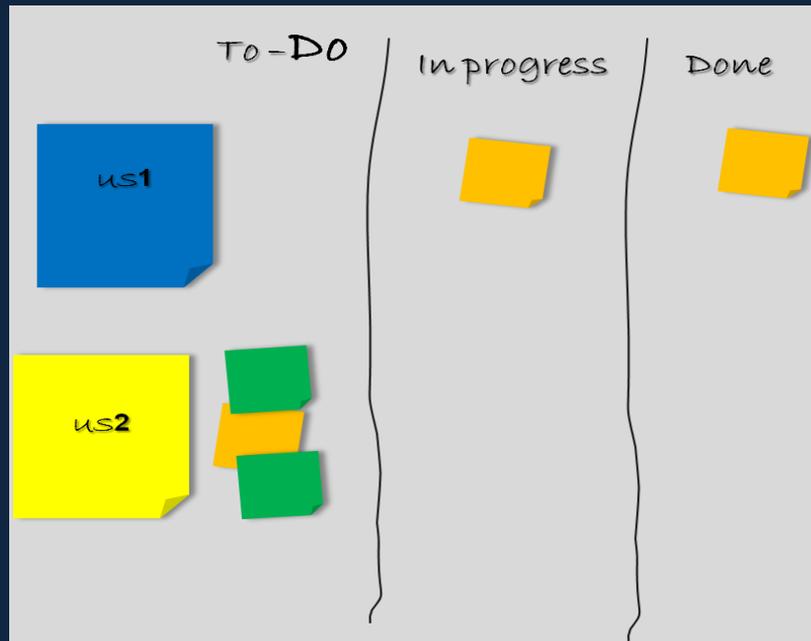
# Roles, artefactos, actividades

## Lists

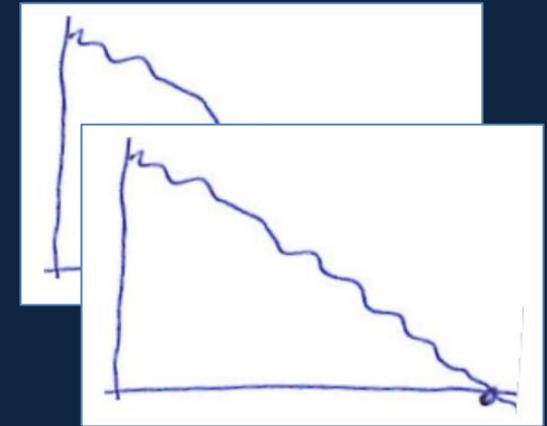


- Product Backlog
- Sprint Backlog
- Incidence Backlog
- Impediments Backlog

## Scrum Board



## Graphs



Release Burn-down  
Sprint Burn-down

# Roles, artefactos, actividades

Sprint 0 o First Sprint



Sprint



Sprint Planning, (planificación del Sprint)



Daily Scrum Meeting, (reunión diaria)



Sprint Review, (Revisión del Sprint)



Sprint Retrospective, (Retrospectiva del Sprint)



Refinement / Grooming, (Refinamiento)

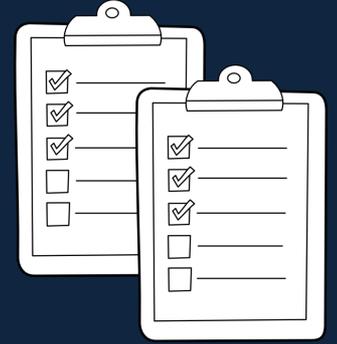


# Artefactos de SCRUM

## Presentación



# Artefactos SCRUM



- Product Backlog

Lista de User Stories

Sólo uno

Responsable: PO

# Artefactos SCRUM



- Sprint Backlog

Lista de User Stories del Sprint

Se puede tocar?

Divisible en tareas?

Las tareas estimadas en que?

Responsable: DT y el SM

# Artefactos SCRUM



- Impedimentos, incidencias y bloqueos

Lista de problemas, que se han de registrar y que afectan a la ejecución de una tarea y, por tanto, del sprint

# Artefactos SCRUM



- **Impedimentos, incidencias y bloqueos**
  - **Impediment Backlog:** Lista de problemas, que sirven como registro para que el Scrum Master pueda buscar soluciones
  - **Incidence Backlog:** La Incidence Backlog es una lista de problemas detectados a nivel de tarea para un Sprint. Cualquier cambio no previsto sobre una tarea se anota en esta lista, para ser tratada en la reunión de Sprint Retrospective.
  - **Parking Backlog:** El Parking Backlog es la lista de tareas que se encuentran “bloqueadas” en un Sprint. Una tarea puede estar bloqueada porque se ha detectado algún problema que impide acabarla, o bien porque se está a la espera de un resultado intermedio, etc.

# Artefactos SCRUM



- Impediment backlog

Problemas que se caracterizan por la “sorpresa” y requieren adaptación. Acostumbra a bloquear la tarea durante tiempo limitado

Quien informa de los problemas?

Ejemplos?

# Artefactos SCRUM



- Incidence Backlog

Se caracterizan por representar un “retraso” y que puede resolverse por el equipo

Quien informa?

Ejemplos?

# Artefactos SCRUM



- Parking Backlog

Se caracterizan por un “bloqueo” sobre una tarea y que ha de resolverse en el tiempo del sprint

Quien informa?

Ejemplos?

# Artefactos SCRUM Scrum Board

Nom del projecte							
Info Equip 😊 😐 😞 😊	User Stories	To Do		In Progress		Completed	
Prod. Backlog	User Story 1	Tk1.1		Tk1.2		Criteris acceptació Sprint	
		Tk1.3					
	User Story 2	Tk2.1	Tk2.2		Tk2.3		Notes
	User Story 3		Tk3.2	Tk3.1			
		Tk3.3			Tk3.4		
		Graphs		Incidence Backlog		Parking Backlog	
		Burn-Down Sprint	Burn-Down Release				

# Roles de SCRUM

## Presentación de los roles



SCRUM team

# Roles - Product Owner



# Roles - Product Owner



Enlace entre el cliente y el equipo de desarrollo.  
Enfocado a negocio o a TIC.

- Da **soporte** para resolver cualquier cuestión **funcional** o **impedimento**
- Estrategia. Conoce el “**negocio**”
- Define los **objetivos**
- Mantiene el **Product Backlog**
- Negocia el **alcance** con el cliente
- Define consensuadamente los **criterios de aceptación** del proyecto y de cada sprint
- Mantiene el presupuesto

Donde participa:

De que es responsable:

Lo veremos mas tarde

Lo veremos mas tarde

# Roles - Scrum Master



# Roles - Scrum Master



El Scrum Master **NO** es el Project Manager.

Es el enlace entre el DT y el PO

- Es un **coach/mentor** para los componentes del Development Team, (DT)
- Proporciona **soporte al DT** y resuelve los problemas
- Reporta, archiva y **lleva registro**
- Propone, promueve y potencia **mejoras sobre el proceso** y sobre el Scrum Team

Donde participa:

De que es responsable:

Lo veremos mas tarde

Lo veremos mas tarde

Roles

Development Team



# Roles

## Development Team



Entre 3 y 9 personas, sin contar el PO ni el SM  
Todos los componentes del equipo deberían estar en contacto directo entre ellos y con el SM

- Es flexible
- Está auto-organizado
- Es multidisciplinar

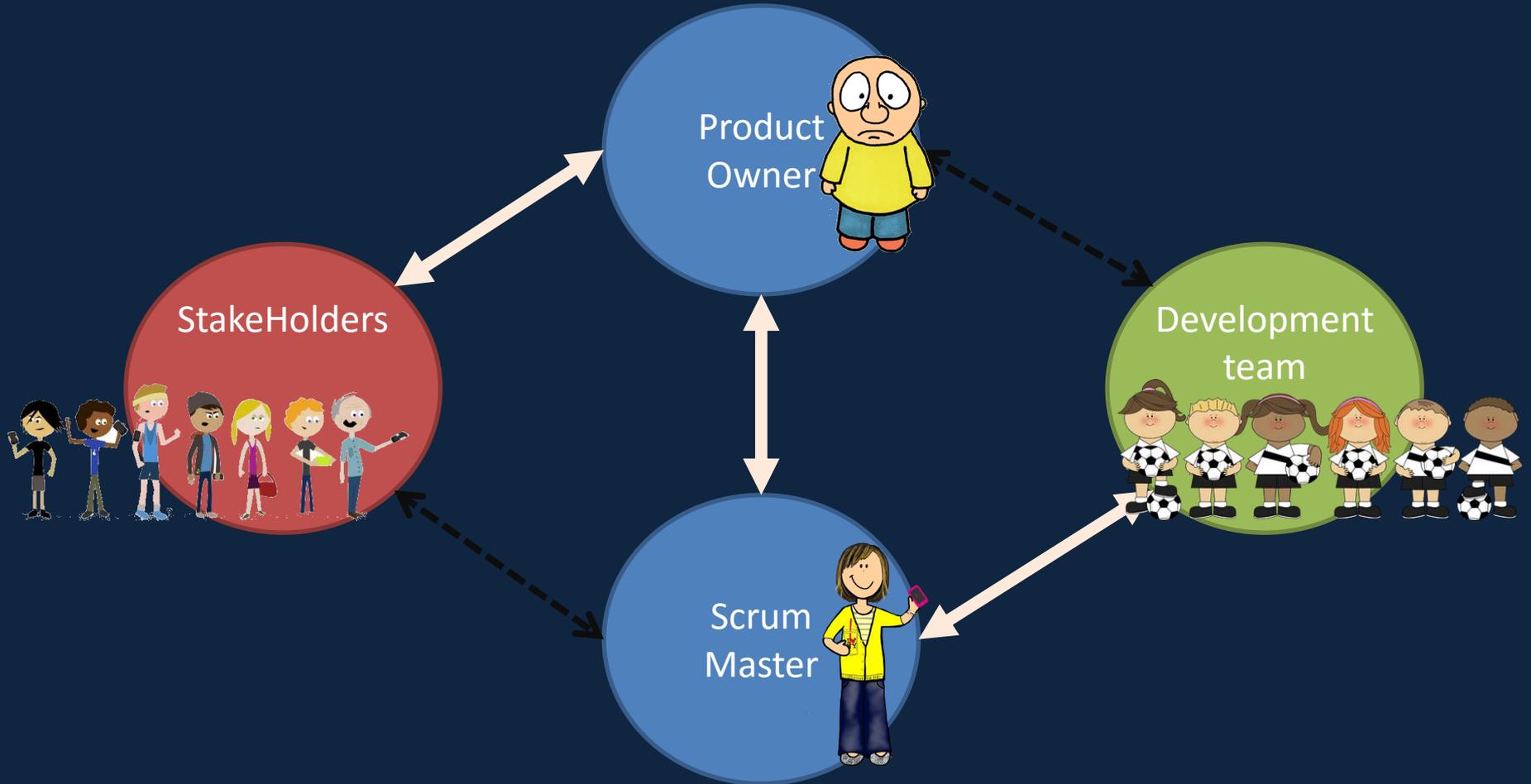
Donde participa:

De que es responsable:

Lo veremos mas tarde

Lo veremos mas tarde

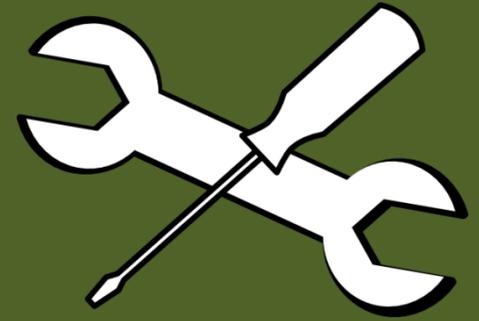
# Roles, artefactos, actividades



# User Stories y Planning Poker



# User Stories



Las User Stories son fichas que explican el detalle funcional de cada ítem del Product Backlog

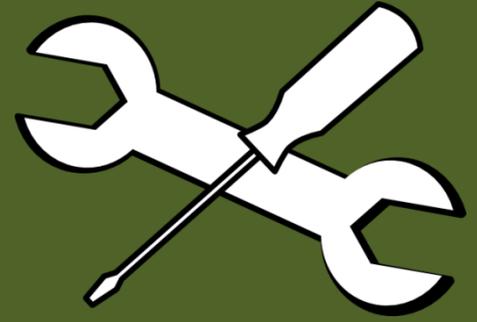
Incluyen información descriptiva

Prioridad

Criterios de aceptación

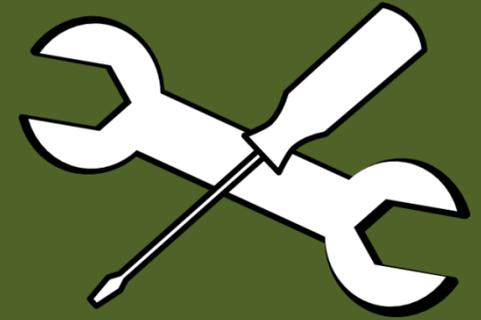
“Peso” en forma de Story Points

# Story Points

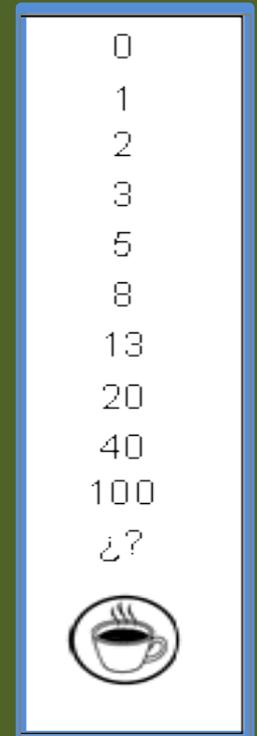


Un Story Point es la forma de consensuar el esfuerzo para construir una funcionalidad dada

# Planning Poker



- Para una funcionalidad dada, se expomen sus características y toda la información necesaria para poder dar una valoración, (incluyendo los criterios de aceptación). Una vez hecha la exposición, cada miembro del equipo la puntúa. El objetivo de éste método de valoración es doble
  1. Consenso
  2. Imparcialidad
- Si pero... Que representa en esfuerzo 1 Story Point?



# Scrum Board

En el Scrum Board se muestra:

- **Las User Stories del Sprint:** Las User Stories se colocan en orden de prioridad, (a la parte superior las mas prioritarias), para que el equipo conozca con un simple vistazo la importancia de cada tarea.
- **Las tareas** de cada User Story y su situación
- **Las listas** de incidencias, impedimentos y Parking Backlog

Las tareas son Post-its que se mueven en el Scrum Board

Cada tarea, (post-it), tiene una estimación inicial y el nombre de la persona que se responsabiliza en cada momento. Si la estimación varia, se ha de anotar al post-it, y si la desviación es grave se ha de registrar una incidencia

# Scrum Board



Columnas del tablero:

- To Do, (Pendiente)
- In Process
- Acabado

**Quien es responsable:** El Scrum Master controla el Scrum Board con la colaboración de todo el DT. Además, el Scrum Master puede cambiar el Scrum Board en tiempo real, (fuera de las Daily Meeting), para adaptarse a cambios, reasignar tareas, atender peticiones del DT si acaba tareas antes de hora, etc.

Nom del projecte						
Info Equip 😊😐😞😊	User Stories	To Do		In Progress		Completed
Prod. Backlog	User Story 1	Tk1.1		Tk1.2		
		Tk1.3				
	User Story 2	Tk2.1	Tk2.2			Tk2.3
	User Story 3		Tk3.2	Tk3.1		
		Tk3.3			Tk3.4	
	Graphs			Incidence Backlog		Parking Backlog
	Burn-Down Sprint	Burn-Down Release				

Criteria  
acceptació  
Sprint

Notes

# Scrum Board

**EXTINTOR**

**USER STORIES**

**PENDENT**

**DEV**

**TEST**

**ACABAT**

**Checklist**

Stiky Points: 0, 1, 2, 3, 5, 8, 13, 20, 40, 100

**Calendar**

Novembre 2014	
Juliol	Agost
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31

**Reunions**

- Reunió Inici Sprint: Dilluns 9h - 12h
- Reunió diària: Cada dia 9:30h - 9:45h
- Reunió Final Sprint: Dimecres 10:30h - 11:30h
- Reunió retrospectiva: Dimecres 12h - 13h

**Line Graph**

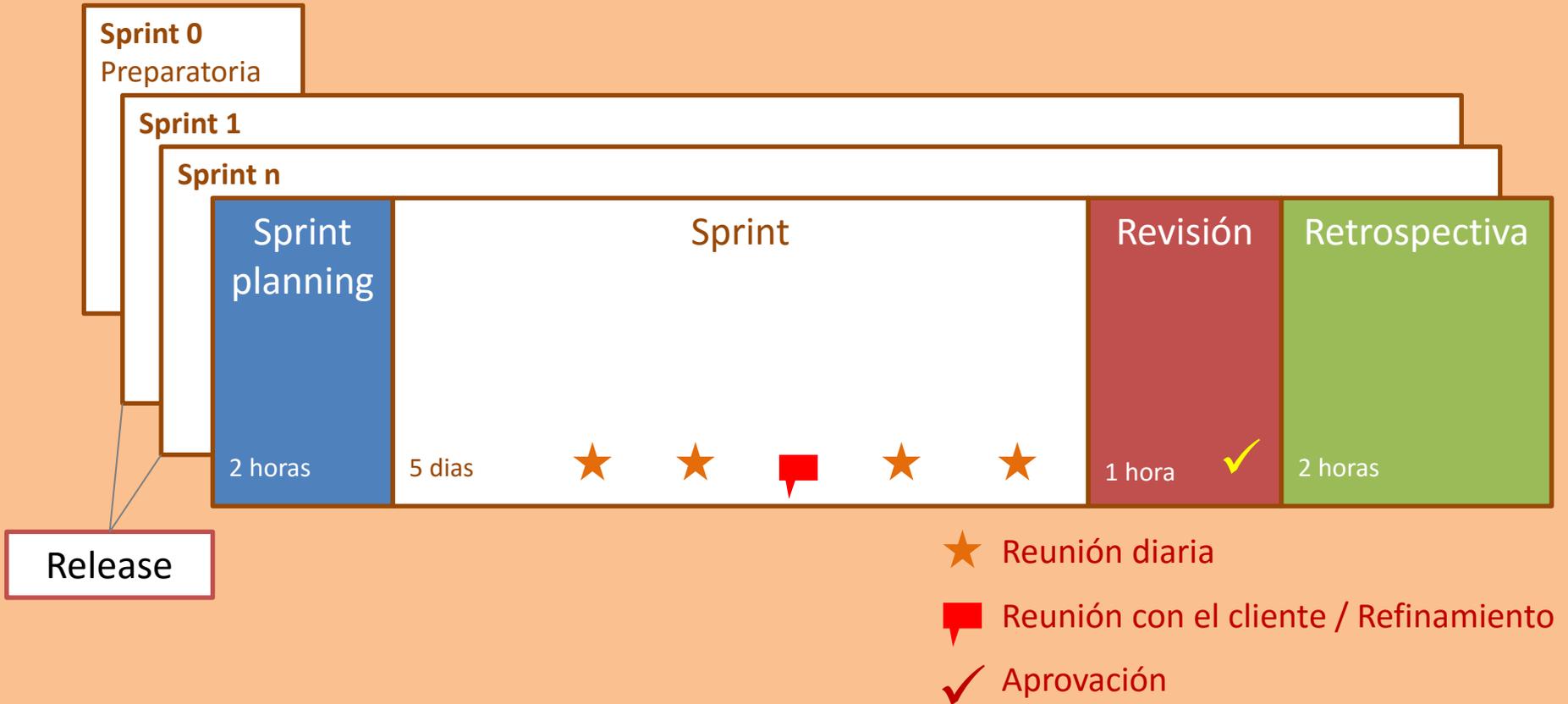
Time	Value
0	10
1	8
2	6
3	4
4	2
5	1

# Actividades SCRUM

En detalle



# Actividades de SCRUM



# Time Box



## Actividad

## Time Box

### **Sprint 0**

No hay un límite establecido para esta fase. Dependerá del tiempo disponible para lanzar el proyecto, o las fechas para la entrega de un prototipo, etc.

### **Sprint Planning**

Un máximo de 8h para Sprints de un mes. Siendo proporcional para Sprints inferiores

### **Daily meeting**

En ningún caso mas de 15 minutos

### **Sprint Review**

Un máximo de 4h para Sprints de un mes. Siendo proporcional para Sprints inferiores

### **Sprint Retrospective**

Un máximo de 3h para Sprints de un mes. Siendo proporcional para Sprints inferiores

### **Grooming**

Se recomienda un tiempo global de entre el 5% y 10% del Sprint.

Actividades de

# SCRUM - Sprint Planning

Sprint  
planning

Sprint

Revisión

Retrospectiva

# Actividades de

# SCRUM - Sprint Planning

Sprint  
planning

Sprint

Revisión

Retrospectiva

## Para que sirve?

1. Para **planificar** en detalle el Sprint
2. Para **recoger la funcionalidad** a desarrollar
3. Para **aclarar dudas**
4. Para crear las User Stories
5. Para determinar los **criterios de aceptación** del Sprint y de cada User Story
6. Para separar el User Story en tareas y determinar el esfuerzo de cada tarea

## Que se necesita?

- User Stories valoradas
- Tareas valoradas
- Product Backlog detallado suficientemente

## Que pasa despues?

- Daily Meeting

Actividades de

# SCRUM - Daily Meeting

Sprint  
planning

Sprint

Revisión

Retrospectiva

# Actividades de

# SCRUM - Daily Meeting

Sprint  
planning

Sprint

Revisión

Retrospectiva

## Para que sirve?

1. Para **explicarse**
2. Para hacer **seguimiento** del estado a nivel de tarea
3. Para determinar que tareas hace cada persona en ese momento
4. Para **resolver dudas**

## Que se necesita?

- Todos los participantes hablan
- Duración máxima: **15 minutos**
- Siempre en el **mismo sitio**
- Siempre a la **misma hora**
- **Obligatorio** para el DT
- Voluntario para el SM
- El PO sólo si se le invita

Actividades de

# SCRUM - Sprint Review

Sprint  
planning

Sprint

Revisión

Retrospectiva

# Actividades de

# SCRUM - Sprint Review

Sprint  
planning

Sprint

Revisión

Retrospectiva

## Para que sirve?

(Parte 1)

1. Para mostrar al PO el resultado/situación final del Sprint

(Parte 2)

1. Para mostrar al usuario/cliente el incremento de producto
2. Obtener aceptación

## Que se necesita?

- La aceptación lanza el siguiente Sprint
- Se ha de explicar al usuario los objetivos del Sprint
- Incluir siempre algún comentario útil
- Se ha de realizar Demo

## Que pasa después?

- Sprint Retrospective

Actividades de

Sprint  
planning

Sprint

Revisión

Retrospectiva

# SCRUM - Sprint Retrospective

# Actividades de

# SCRUM - Sprint Retrospective

Sprint  
planning

Sprint

Revisión

Retrospectiva

## Para que sirve?

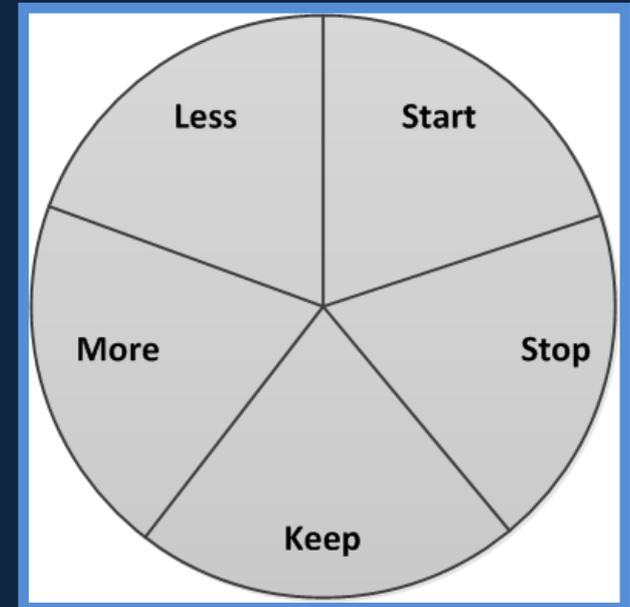
1. Para **debatir** entre SM y DT sobre el curso del Sprint
2. **Revisar** incidentes y bloqueos
3. Para **buscar soluciones**
4. Para aplicar la **mejora continua**

## Que se necesita?

- Es la aplicación de la mejora continua

## Que pasa después?

- Se intentan aplicar las mejoras acordadas en el siguiente Sprint



# Actividades de

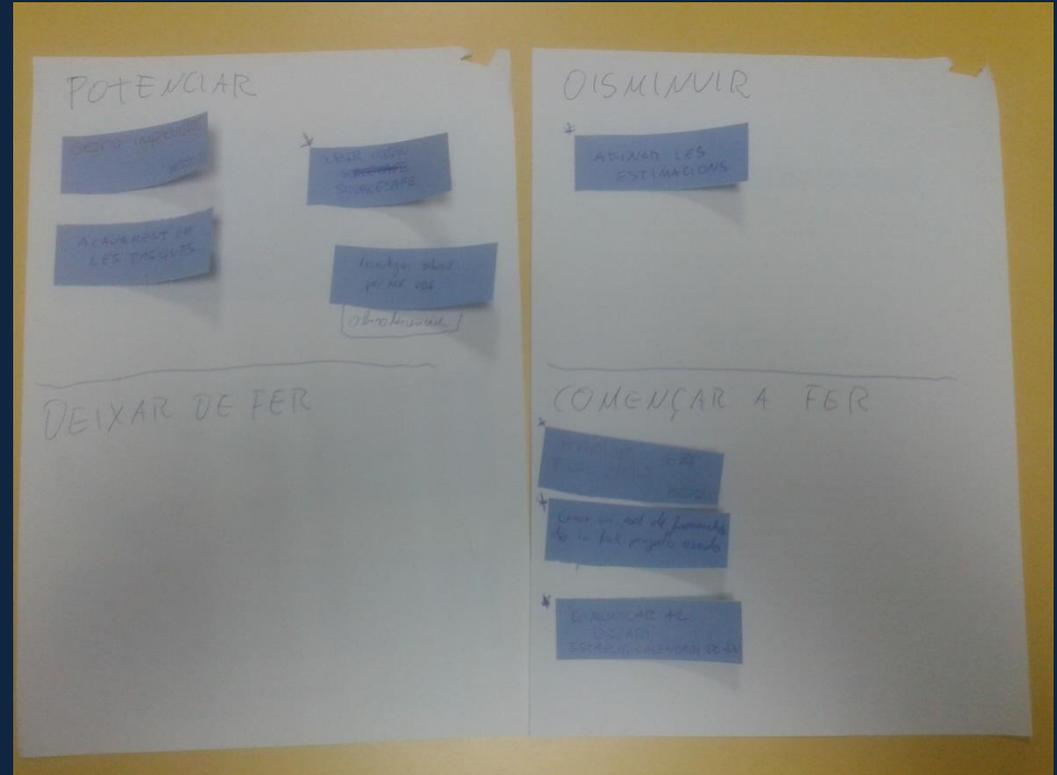
Sprint  
planning

Sprint

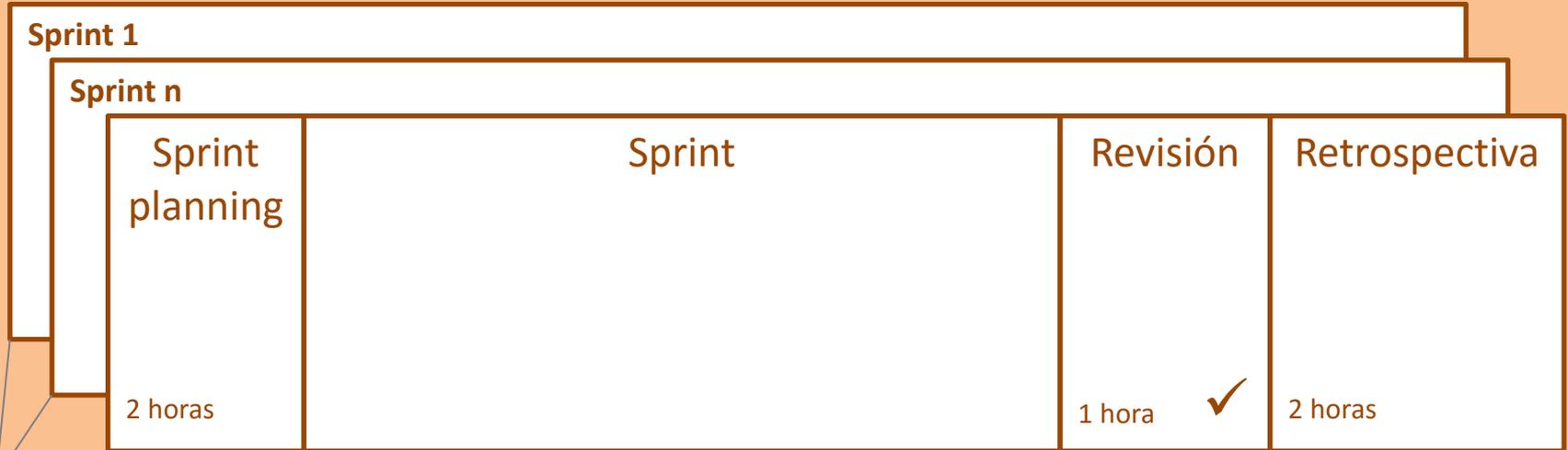
Revisión

Retrospectiva

# SCRUM - Sprint Retrospective



# La release



Release

La Release es un convenio con el Product Owner, por el que es posible entregar producto cada cierto número de Sprints, dependiendo de la funcionalidad a construir

# Relación entre actividades y roles



# Relación entre Actividades y roles



	DT	SM	PO	Stakeholder
<b>Sprint 0</b>	Opcional	Sí	Sí	Opcional
<b>Sprint Planning</b>	Sí	Sí	En la definición de lo que se hará	
<b>Daily meeting</b>	Sí	Opcional	Sólo si es invitado	
<b>Sprint Review</b>	Recomendable	Sí	Sí	Sólo a la 2ª parte de la reunión, donde se hace demo y se pide aceptación
<b>Sprint Retrospective</b>	Sí	Sí	Sólo si es invitado	
<b>Grooming</b>	Opcional	Sí	Sí	Opcional

# Roles - Product Owner



Enlace entre el cliente y el equipo de desarrollo  
Enfocado a negocio o a TIC.

- Da **soporte** para resolver cualquier cuestión **funcional** o **impedimento**
- Estrategia. Conoce el “**negocio**”
- Define los **objetivos**
- Mantiene el **Product Backlog**
- Negocia el **alcance** con el cliente
- Define consensuadamente los **criterios de aceptación** del proyecto y de cada sprint
- Mantiene el presupuesto

## Donde participa:

- Sprint 0
- Sprint Planning (definición de los objetivos)
- Sprint Review
- Sprint Retrospective si se le invita
- Grooming que pida o donde sea invitado

## De que es responsable:

- Del **Product Backlog**
- Del gráfico **Release Burn-down**

**Recomendaciones/Restricciones:** El PO no puede ser a la vez el Scrum Master.

# Roles - Scrum Master



El Scrum Master **NO** es el **Project Manager**.  
Es el enlace entre el DT y el PO

- Es un **coach/mentor** para los componentes del Development Team, (DT)
- Proporciona **soporte al DT** y resuelve los problemas
- Reporta, archiva y **lleva registro**
- Propone, promueve y potencia **mejoras sobre el proceso** y sobre el Scrum Team

## Donde participa:

- Sprint 0
- Sprint Planning
- Opcionalmente en los Daily Meetings
- Sprint Review y Sprint Retrospective
- A las reuniones de Grooming que pida y a las que sea invitado

## De que es responsable:

- Del **Sprint Backlog** junto con el DT
- Del **Scrum Board** junto con el DT
- Del gráfico **Sprint Burn-down**
- Del **Incident Backlog** y del **Impediment Backlog**
- De coordinar la reunión de **Scrum Retrospective**

# Roles

## Development Team



Entre 3 y 9 personas, sin contar el PO ni el SM  
Todos los componentes del equipo deberían estar en contacto directo entre ellos y con el SM

- **Es flexible**
- **Està auto-organitzado**
- **Es multidisciplinar**

### Donde participa:

- Sprint Planning
- Daily Meeting
- Opcionalment al Sprint Review
- Sprint Retrospective
- A las reuniones de grooming donde sea invitado

### De que es responsable:

- Determinar el **detalle de cada funcionalidad** descrita al Product Backlog, y hacer la subdivisión en **tareas**
- **Estimación del esfuerzo**, en Story Points al Product Backlog, y en horas a cada tarea
- Gestionar el Sprint Backlog
- **Proporcionar producto "acabado"**. Convenientemente testeado, siguiendo los criterios de aceptación marcados.
- Ejecutar diariamente el **Daily meeting** y cumplir sus normas

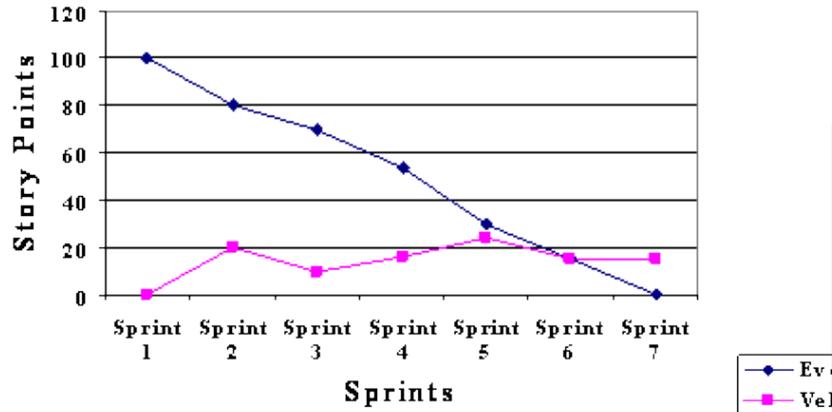
# Los gráficos de SCRUM

## En detalle

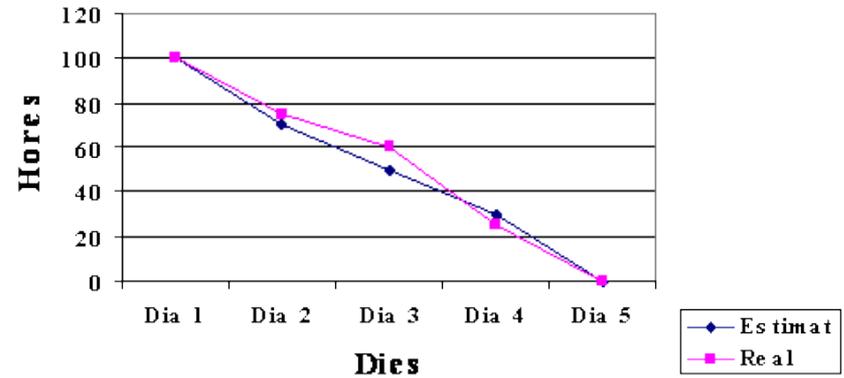


# Artefactos SCRUM Graphs

## Release Burn-down



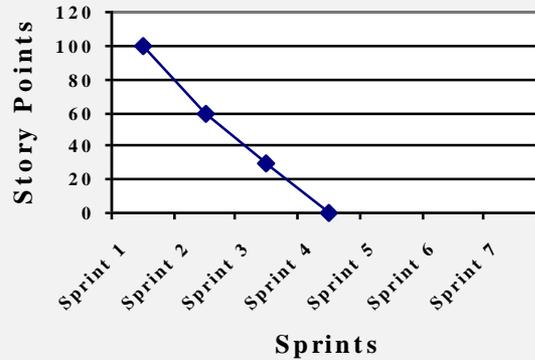
## Sprint Burn-down



# Artefactos SCRUM Release Burn Down

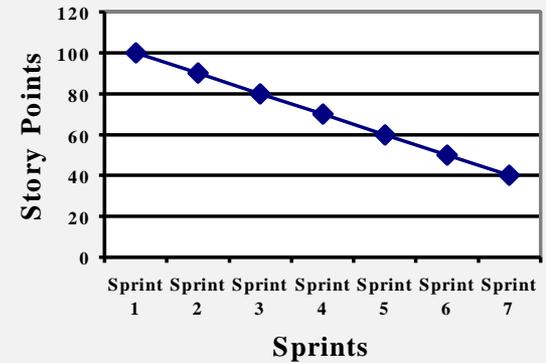
Exemples de desviacions en el Release Burn-down a tenir en compte:

Release Burn-down



L'equip va molt ràpid. Sobren Sprints

Release Burn-down

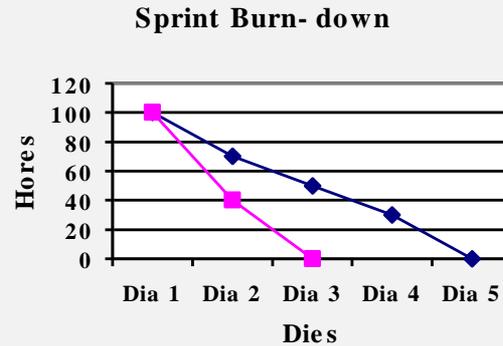


L'equip va molt lent. Falten Sprints o a l'equip li passa alguna cosa

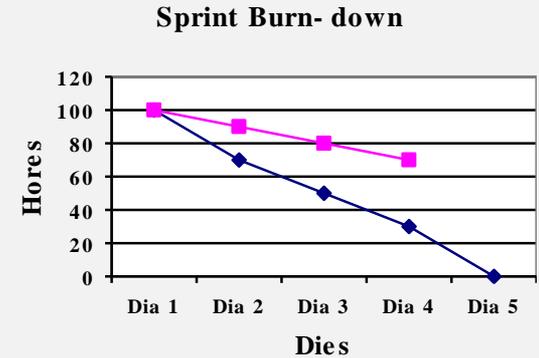
# Artefactos SCRUM Sprint Burn Down

El gráfico de Sprint Burn-down muestra en todo momento el “trabajo pendiente”, y permite ver la velocidad a la que se resuelven las funcionalidades comprometidas en el sprint. Para cada día de iteración el SM incorpora el total de horas de tareas en las diferentes columnas de trabajo pendiente o en curso

Exemples de desviacions en el Sprint Burn-down a tenir en compte:



Les tasques assignades al Sprint s'estan resolent molt ràpidament. L'equip va fer una estimació “pessimista”. Probablement no s'ha triat el nombre suficient d'ítems del Product Backlog. Caldria afegir-ne més



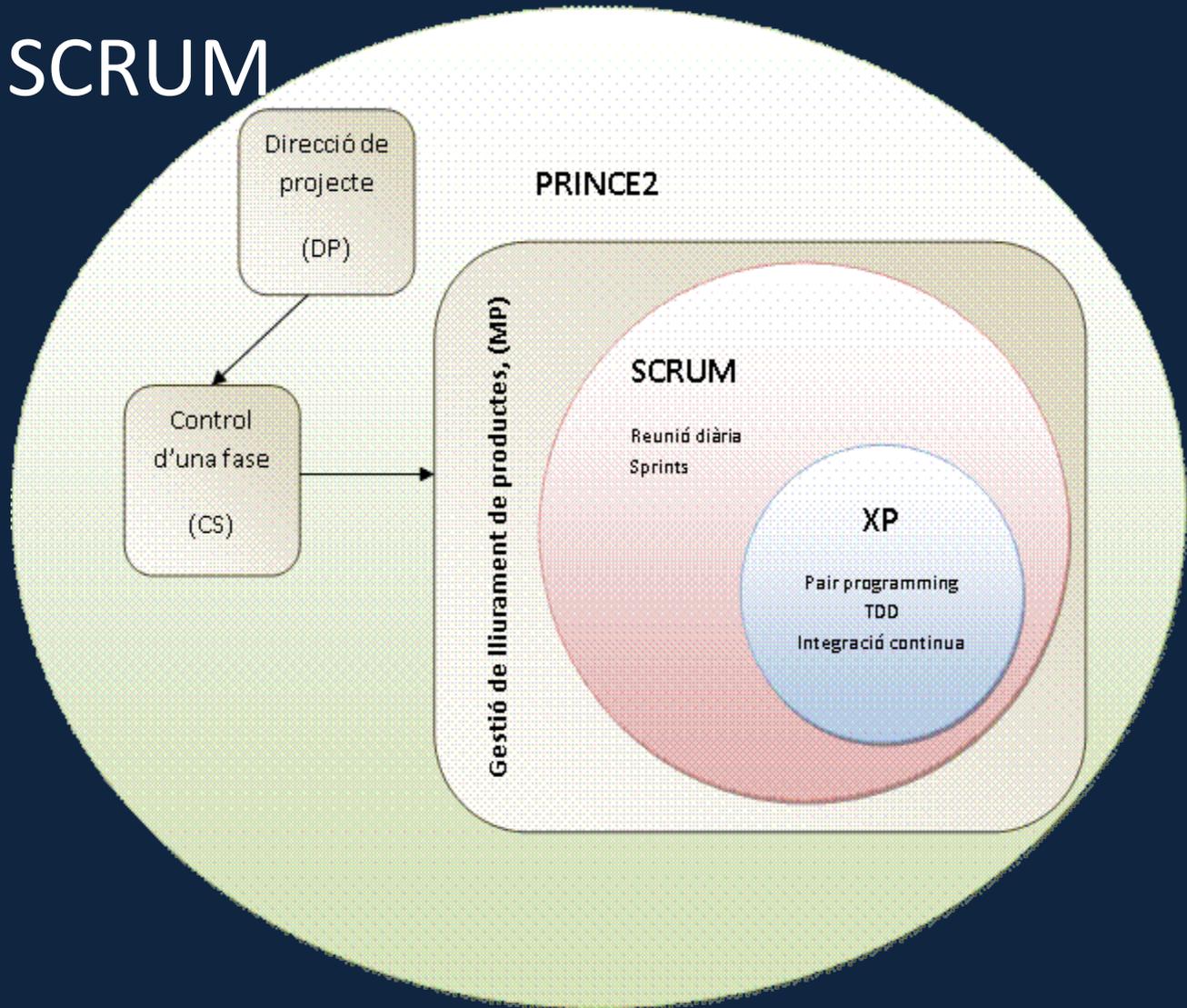
Les tasques s'estan resolent molt lentament. L'equip va fer una estimació “optimista”. Cal renegociar el Sprint amb el PO. Cal treure tasques

Usualmente el gráfico se actualiza después de llevar a cabo la reunión diaria

# Extendiendo SCRUM

# Extiendiendo SCRUM

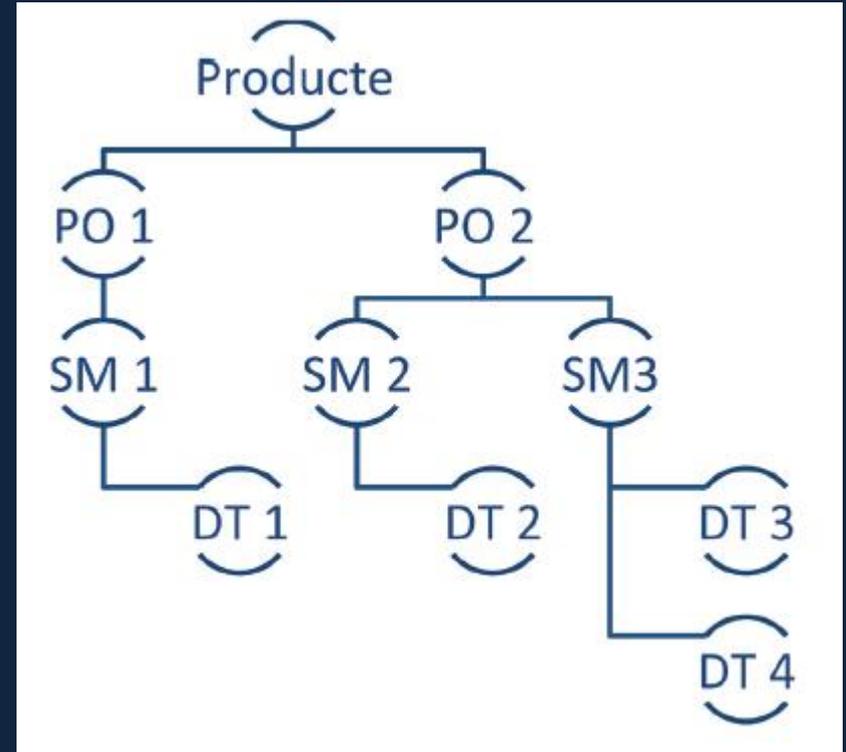
Convivir con  
otros  
métodos



# Extendiendo SCRUM

A nivel de producto:

1. Un producto tiene UN ÚNICO Product Backlog
2. Un producto puede tener diversos PO. Cada PO ve una “vista” del product Backlog
3. Un PO puede gestionar diversos SM
4. Un SM responde sólo a un PO para un producto
5. Un SM puede tener a su cargo diversos DT, y tiene UN ÚNICO Sprint Backlog
6. Un DT tiene UN ÚNICO SM



# Extendiendo SCRUM

La CoE (Center Of Excellence)

1. Bicefalia: Chief PO + Chief SM
2. Organización por líneas de producto

Funciones de la CoE

1. Gestión del programa

Gestiona la evolución de la infraestructura y las soluciones TIC

2. Soporte :

Metodología, herramientas, plantillas

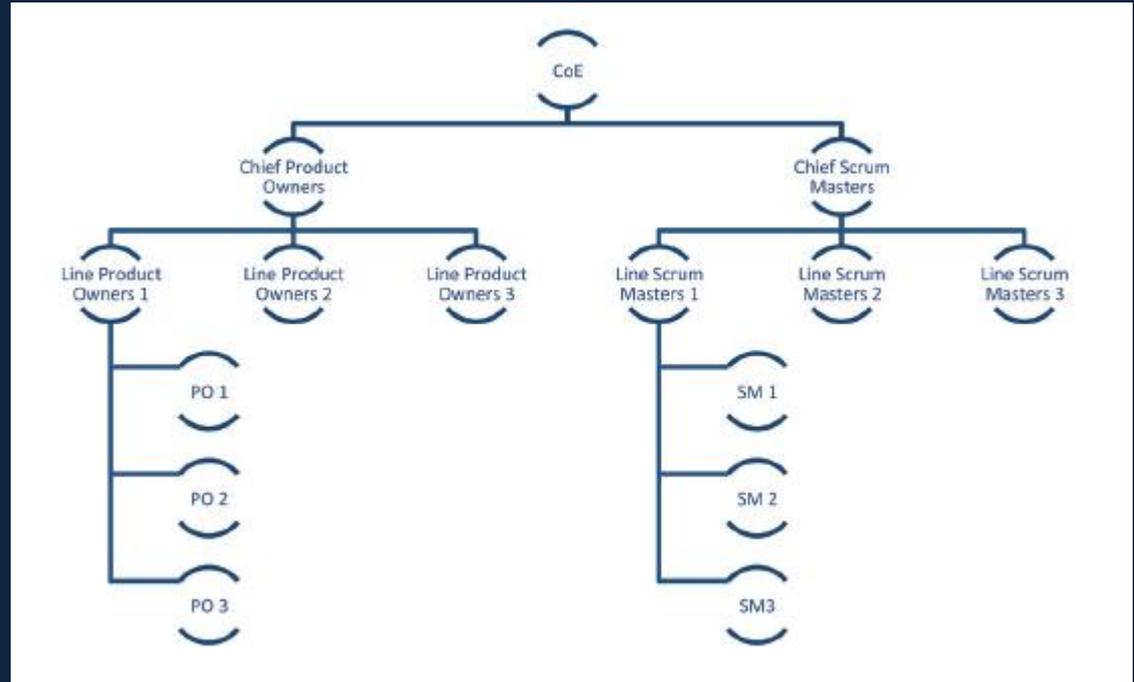
3. Formación :

Capacitación y mentoría

En herramientas, en método, en habilidades

4. Asesoramiento :

Intercede y centraliza las comunicaciones



Gestión de producto

+ CoE

=

**SCRUM PMO**

# Certificación SCRUM



1. Test en línea
2. Inglés
3. 60 minutos
4. Formato: Múltiples respuestas test
5. 80 preguntas
6. El test se aprueba a partir del 85% de respuestas acertadas, (68)
7. 150\$

Test de prueba

<https://www.scrum.org/Assessments>

# Como aplicar SCRUM?

1. [Tienes 3 a 9] + 2?
2. Separar los proyectos. Entender los requerimientos. Conocer el alcance
3. Establecer ciclos, (sprints)
4. Establecer reuniones diarias
5. Hacer partícipe al equipo y fomentar la comunicación
6. Fomentar la transparencia, la inspección y la adaptación
7. Determinar roles y vías de comunicación con el usuario

## Como se si aplico SCRUM?

[www.sierra-charlie.com/download/SB023-TheNokiaTest.pdf](http://www.sierra-charlie.com/download/SB023-TheNokiaTest.pdf)

[www.verheulconsultants.nl/ScrumButtTest.pdf](http://www.verheulconsultants.nl/ScrumButtTest.pdf)

*Se puede decir que haces SCRUM cuando, como mínimo :  
(Transparencia, Inspección, adaptación y Mejora continua)  
+ (Daily Meeting, Time Box, Sprint)*

# Literatura

## 1. SCRUM y XP desde las trincheras

<http://www.proyectalis.com/wp-content/uploads/2008/02/scrum-y-xp-desde-las-trincheras.pdf>

## 2. SCRUM guide de Scrum.org

<http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-CAT.pdf#zoom=100>

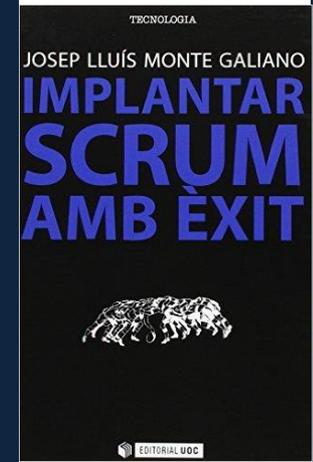
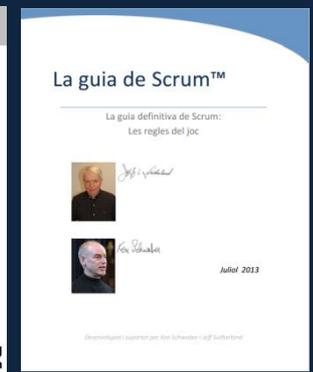
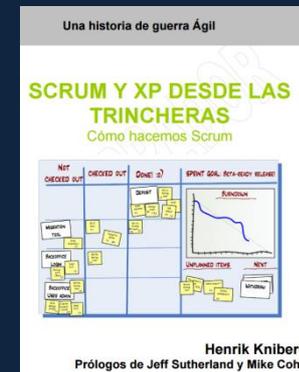
## 3. Kanban vs Scrum

<https://www.crisp.se/file-uploads/Kanban-vs-Scrum.pdf>

## 3. Implantar SCRUM amb èxit

Editorial UOC

<http://www.editorialuoc.cat/implantar-scrum-amb-exit>



# Gracias

[www.slideshare.net/jlmoga/introscrumes](http://www.slideshare.net/jlmoga/introscrumes)  
[www.slideshare.net/jlmoga/introscrumcat](http://www.slideshare.net/jlmoga/introscrumcat)

[www.moga.cat/agils](http://www.moga.cat/agils)

*2016, Josep Lluís Monte Galiano  
moga@moga.cat  
www.moga.cat*