

# Metodología Scrum

Autor: Manuel Trigás Gallego

Director de Proyecto: Ana Cristina Domingo Troncho

*Desarrollo detallado de la fase de aprobación de un proyecto informático mediante el uso de metodologías ágiles.*

# ¿Qué es un proyecto?

Según la definición que nos proporciona PMI en su guía **PMBOOK**, un proyecto se podría definir como *"un servicio temporal que se lleva a cabo para crear un producto, servicio o resultado único"*.

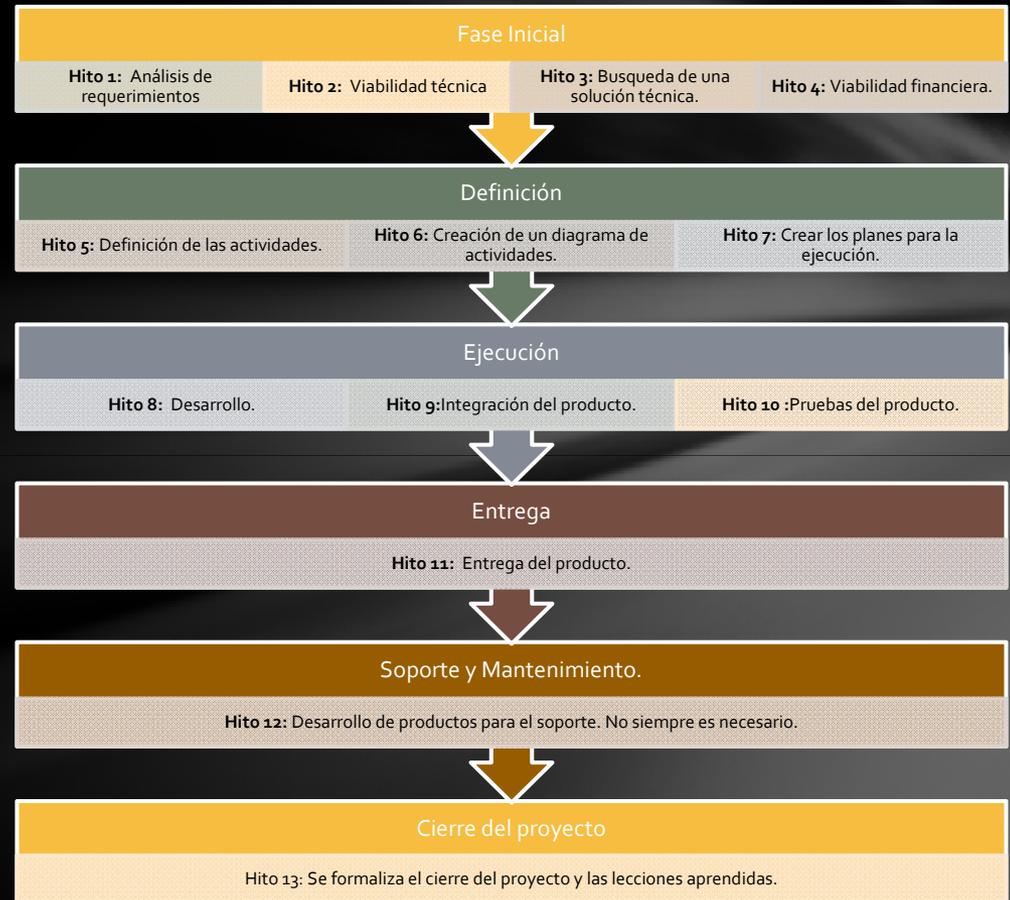


# Fases y ciclos de vida de un proyecto

**Ciclo de vida de un proyecto:** Son fases que servirán para tener un control más específico de cada una de las tareas.

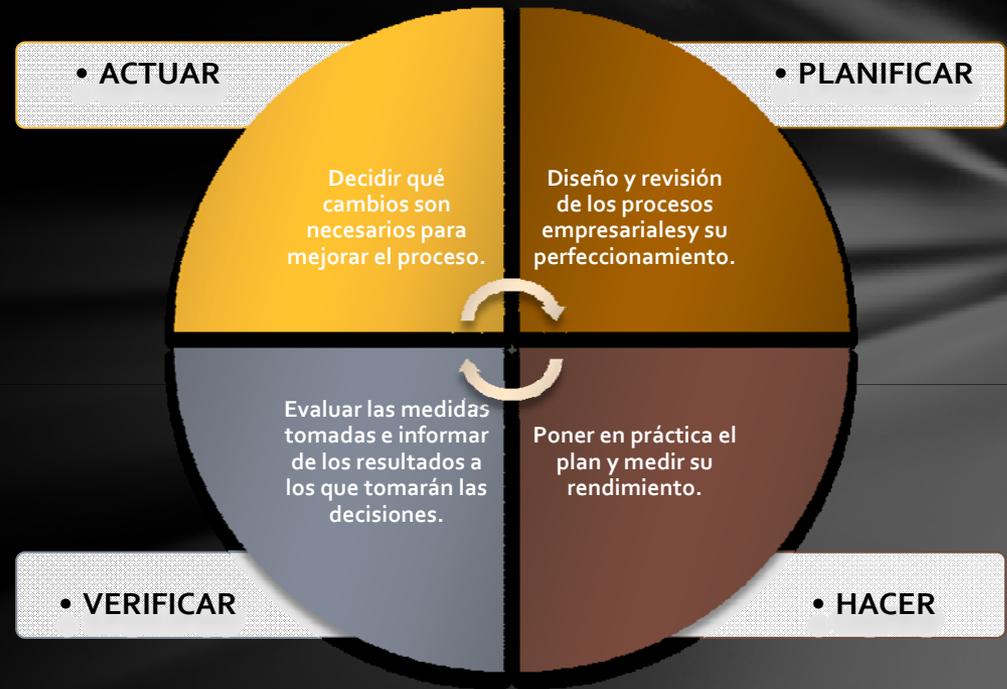
Los ciclos de vida del proyecto responden a:

- ¿Qué trabajo se va a realizar en cada fase?
- ¿Quién participará en cada fase?
- Las fases se definirán de forma secuencial, es decir, que una fase no comienza hasta que termina la otra. Suelen contener una serie hitos o tareas que marcan los momentos más importantes en el desarrollo del proyecto.



# El modelo tradicional de planificación

*El modelo PDCA fue desarrollado por Walter Shewhart en los laboratorios Bell en 1930, pero fue popularizado por W. Edwards Deming y se suele uno referir a este modelo como "circulo de Deming" o también "espiral de mejora continua".*



# Metodología y tipos

*Definiremos la metodología como aquella disciplina que indicará que métodos y técnicas hay que usar en cada fase del ciclo de vida de desarrollo del proyecto.*

## Tipos de metodologías.

### 1. Por su naturaleza:

- M.Orientadas al flujo de información → Diseño estructurado de Yourdon.
- M.Orientadas a objetos → Rational Unified Process
- M.Híbrida → Métrica 3, SSADM

### 2. Por su formalismo:

- M.Pesadas → Casada, RUP.
- M.Ágiles → Extrem Programming, Scrum.



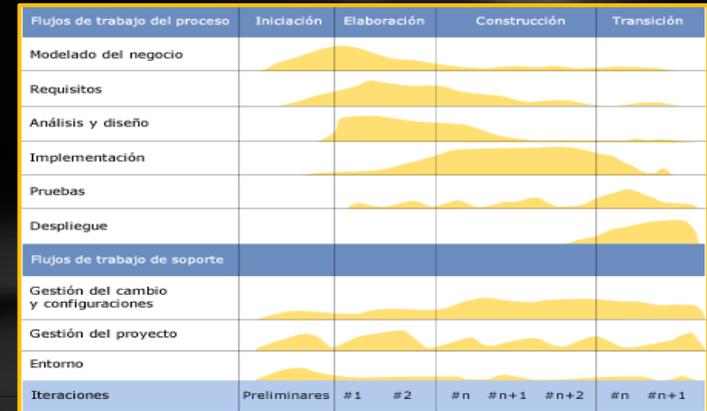
# Metodologías Tradicionales

*Las características más relevantes de esta metodología son:*

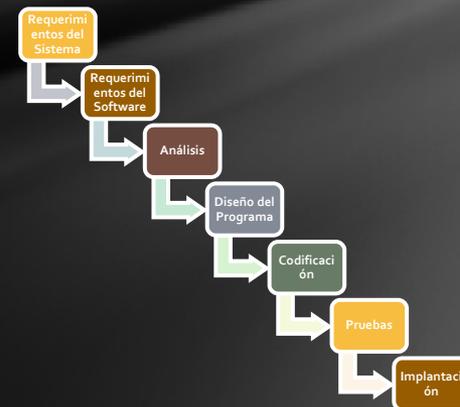
- *Los requisitos son definidos durante todo el proyecto.*
- *Se basa en los procesos.*
- *Se supone que el proyecto no va a surgir ningún tipo de cambio por lo tanto no está sujeto a variables.*
- *Los proyectos suelen estar bien documentados.*
- *Gestión predictiva.*
- *El desarrollo se define en fases cuyo conjunto se denomina "ciclo de vida".*
- *Documentación exhaustiva de todo el proyecto.*
- *Se enfocan en obtener el producto en tiempo estimado y con el coste establecido.*

Tipos de metodologías:

**RUP:**



**CASCADA:**



# Metodologías Ágiles

*Las metodologías ágiles surgen como una alternativa a las metodologías tradicionales las cuales, tal y como acabamos de ver en los apartados anteriores son demasiado burocráticas y por tanto rígidas para las actuales características del mercado.*

*La característica realmente nueva que aportan estas metodologías es reconocer a las personas como el principal valor para que un proyecto consiga terminarse de forma correcta. "El factor más importante en el desarrollo de software no son las técnicas y las herramientas que emplean los programadores, sino la calidad de los programadores." (Robert. L. Class).*



# Comparativa Metodología Tradicional y Metodología Ágil

## *METODOLOGIA TRADICIONAL*

---

- *Se identifican las tareas al inicio del proyecto.*
- *Control predictivo en que predice las variables de tiempo, alcance y presupuesto. El desarrollo se realiza en cascada, se hace una entrega final del proyecto.*
- *Debido a que las entregas se hacen al final, puede que el producto no cumpla con los requisitos establecidos y sea necesario hacer cambios que comprometan a dos elementos o vértices del triángulo: coste y tiempo. Proceso controlado con muchas más normas.*
- *Uso de reuniones entre equipo y cliente.*
- *Resistencia a los cambios.*
- *No se suelen hacer análisis del trabajo realizado, denominado retrospectión, en el que se intenta ver las cosas que se han realizado, de forma correcta, qué hay que mejorar y qué problemas podrían aparecer.*
- *Mayor número de roles.*

## *METODOLOGIA ÁGIL*

---

- *Se basa en el control empírico, en que se asume que va a haber cambios en el contexto del proyecto, por lo tanto, el control del proyecto se basará en controlar los resultados obtenidos y en función de éstos, hacer las adaptaciones adecuadas (ciclo PDCA).*
- *Las fases se plantean en función de los objetivos del producto, que suelen ser en cortos períodos de tiempo y en los que se hacen demostraciones del producto a los clientes; de esta forma es más fácil realizar los cambios. El proceso no necesita de tanto control.*
- *El cliente es parte del proyecto.*
- *Todo el equipo participa en todas las fases del proyecto.*
- *Hay menos roles.*
- *Se realiza retrospectiva durante todo el proyecto.*

# El Manifiesto Ágil

*"Estamos poniendo al descubierto mejores métodos para desarrollar software, haciéndolo y ayudando a otros a que lo hagan con este trabajo, trabajo, hemos llegado a valorar:*

A los individuos y su interacción

Por encima de

los procesos y las herramientas.

El software que funciona

Por encima de

documentación exhaustiva.

La colaboración con el cliente

Por encima de

de la negociación contractual.

Las respuestas al cambio

Por encima de

seguimiento de un plan.

*"Aunque hay valor en los elementos de la derecha, valoramos más los de la izquierda"*

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.

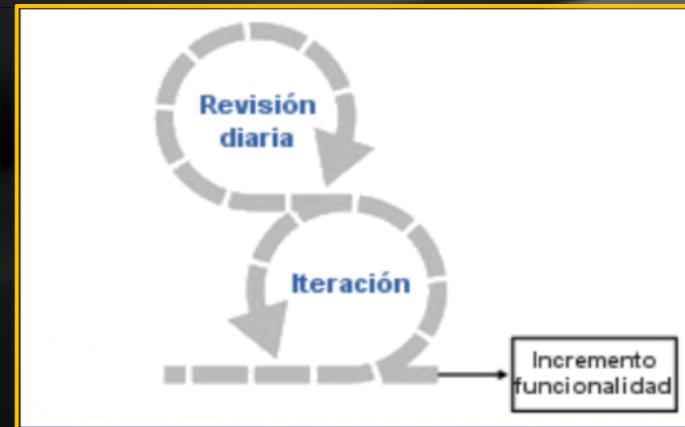
# Scrum

En el año 1986 **Takeuchi y Nonaka** publicaron el artículo "The New Product Development Game" el cual dará a conocer una nueva forma de gestionar proyectos en la que la agilidad, flexibilidad, y la incertidumbre son los elementos principales.

Se comparó esta forma de trabajo en equipo, con la colaboración que hacen los jugadores de Rugby y la utilización de una formación denominada **SCRUM**.

Scrum al ser una metodología de desarrollo ágil tiene como base la idea de creación de ciclos breves para el desarrollo, que comúnmente se llaman **interacciones** y que en Scrum se llamarán "Sprints".

Se basa en el ciclo de desarrollo ágil que consta de 5 fases: **Concepto** → **Especulación** → **Exploración** → **Revisión** → **Cierre**.



# Componentes de Scrum

*Scrum se puede dividir de forma general en 3 fases, que podemos entender como reuniones. Las reuniones forman parte de los artefactos de esta metodología junto con los roles y los elementos que lo forman.*

## 1. LAS REUNIONES:

- **Planificación del Backlog:** Se definirá un documento en el que se reflejarán los requisitos del sistema por prioridades.
- **Seguimiento del Sprint:** En esta fase se hacen reuniones diarias en la que las s para evaluar el avance de las tareas serán:
- **Revisión del Sprint:** Cuando se finaliza el Sprint se realizará una revisión del incremento que se ha generado.

## 2. LOS ROLES

- a) **Los cerdos:** Son las personas que están comprometidas con el proyecto y el proceso de Scrum.
- **Product Owner:** Es la persona que toma las decisiones y es la que realmente conoce el negocio del cliente y su visión del producto
  - **Scrum Master:** Es el encargado de comprobar que el modelo y la metodología funciona.
  - **Equipo De Desarrollo:** Suele ser un equipo pequeño de unas 5-9 personas y tienen autoridad para organizar y tomar decisiones para conseguir su objetivo.
- b) **Las gallinas:**
- **Usuarios:** Es el destinatario final del producto.
  - **Stakeholders:** Las personas a las que el proyecto les producirá un beneficio. Participan durante las revisiones del Sprint.
  - **Managers:** Toma las decisiones finales participando en la selección de los objetivos y de los requisitos.



# Elementos de Scrum

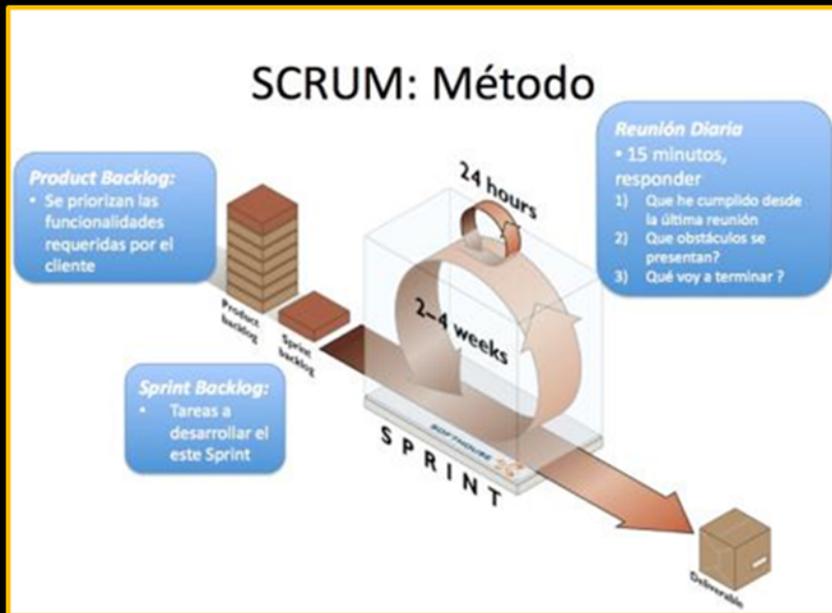
**Product Backlog:** lista de necesidades del cliente.

**Sprint Backlog:** lista de tareas que se realizan en un Sprint.

**Incremento:** parte añadida o desarrollada en un Sprint, es un parte terminada y totalmente operativa.

Id	Prioridad	Descripción	Est.	Por
1	Muy alta	Plataforma tecnológica	30	AR
2	Muy alta	Interfaz usuario	40	LR
3	Muy alta	Un usuario se registra en el sistema	40	LR
4	Alta	El operador define el flujo y textos de un expediente	60	AR
5	Alta	Etc...	999	XX

## Ejemplo de Product Backlog



Requisito	Tarea	Quien	Estado (No iniciada / en progreso / completada)	Día:											
				1	2	3	4	5	6	7	8	9	10		
<b>Horas pendientes</b>				1120	1088	1076	1048	1040	1032	1020	1008	992	972		
Requisito A	Tarea 1	Joao	Completada	16	8										
Requisito A	Tarea 4	Laura	Completada	4											
Requisito A	Tarea 5	Laura	Completada	4											
Requisito A	Tarea 3	Gabri	Completada	8											
Requisito A	Tarea 2	Laura	Completada	16	8	4									
Requisito A	Tarea 6	Gabri	Completada	8	8	8									
Requisito A	Tarea 7	Joao	Completada	16	16	16	8								
Requisito A	Tarea 8	Laura	Completada	8	8	8									
Requisito A	Tarea 9	Laura	Completada	8	8	8	8	8							
Requisito A	Tarea 10	Laura	Completada	8	8	8	8	8	8	4					
Requisito A	Tarea 11	Joao	Completada	16	16	16	16	16	16	16	8				
Requisito B	Tarea 12	Gabri	Completada	16	16	16	16	16	16	16	16	16	8		
Requisito B	Tarea 13	Laura	Completada	16	16	16	16	16	16	16	16	16	8		
Requisito B	Tarea 14	Joao	En progreso	8	8	8	8	8	8	8	8	8	8	4	
Requisito B	Tarea 15	Gabri	En progreso	8	8	8	8	8	8	8	8	8	8	8	
Requisito B	Tarea 16	Laura	En progreso	8	8	8	8	8	8	8	8	8	8	8	
Requisito C	Tarea 17	Joao	No iniciada	4	4	4	4	4	4	4	4	4	4	4	
Requisito C	Tarea 18	Gabri	No iniciada	8	8	8	8	8	8	8	8	8	8	8	
Requisito C	Tarea 19	Laura	No iniciada	16	16	16	16	16	16	16	16	16	16	16	
Requisito C	Tarea 20	Joao	No iniciada	8	8	8	8	8	8	8	8	8	8	8	

## Ejemplo de Sprint Backlog

# Product Backlog

*Es el inventario en el que se almacenan todas las funcionalidades o requisitos en forma de lista priorizada. Estos requisitos serán los que tendrá el producto o los que irá adquiriendo en sucesivas iteraciones.*

*La lista será gestionada y creada por el cliente con la ayuda del Scrum Master, quien indicará el coste estimado para completar un requisito, y además contendrá todo lo que aporte un valor final al producto.*

**Las tres características principales de esta lista de objetivos serán:**

- **Contendrá los objetivos del producto, se suele usar para expresar las historias de usuario .**
- **En cada objetivo, se indicará el valor que le da el cliente y el coste estimado; de esta manera, se realiza la lista, priorizando por valor y coste, se basará en el ROI.**
- **En la lista se tendrán que indicar las posibles iteraciones y los releases que se han indicado al cliente.**
- **La lista ha de incluir los posibles riesgos e incluir las tareas necesarias para solventarlos.**

# Desarrollo de las fases de un proyecto con Scrum

# Preparación del Proyecto

Conocida como **Sprint Cero**, es la fase inicial en la que se intenta comprender el caso de negocio con la finalidad de tomar decisiones que agreguen valor al producto.

## Las tareas a realizar en el Sprint 0 son:

**Definir el proyecto:** Se debería de indicar de forma clara el propósito del proyecto, no es necesario entrar en detalle pero sí que todo el equipo sea capaz de entender cuáles son las necesidades del producto y del cliente.

**Definir "terminado":** Marcará el punto en el que se va a considerar que la tarea está terminada.

**Definición del Backlog inicial:** Se comienza la creación del Backlog del producto para que el Sprint siguiente contenga elementos de la lista suficientes para comenzar a trabajar.

**Definición de los entregables:** Una vez que se tiene el Backlog con las funcionalidades, es necesario establecer criterios para hacer pequeñas entregas "entregables" del producto y así obtener su valor y un feedback temprano.

## Constitución del equipo:

Se hace una reunión inicial con todos los roles del equipo para tratar:

- **Dimensión del proyecto.**
- **Revisiones del Backlog.**
- **Organización del equipo y horario para establecer reuniones de control.**

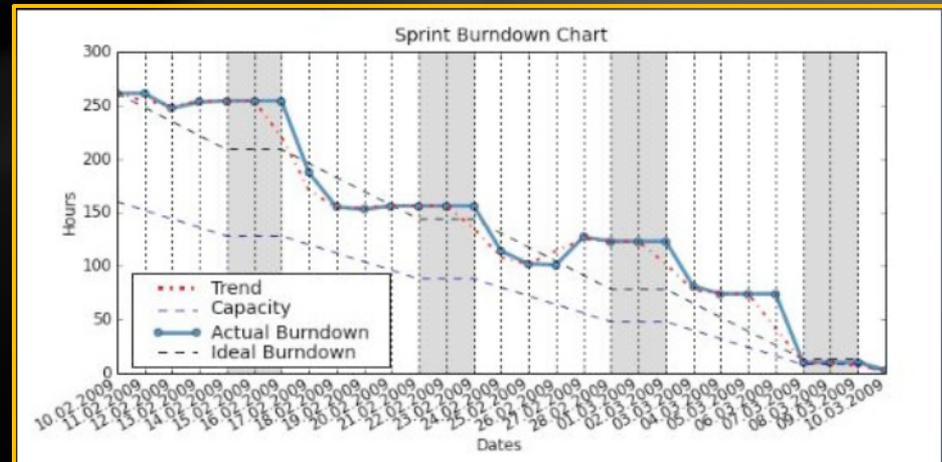
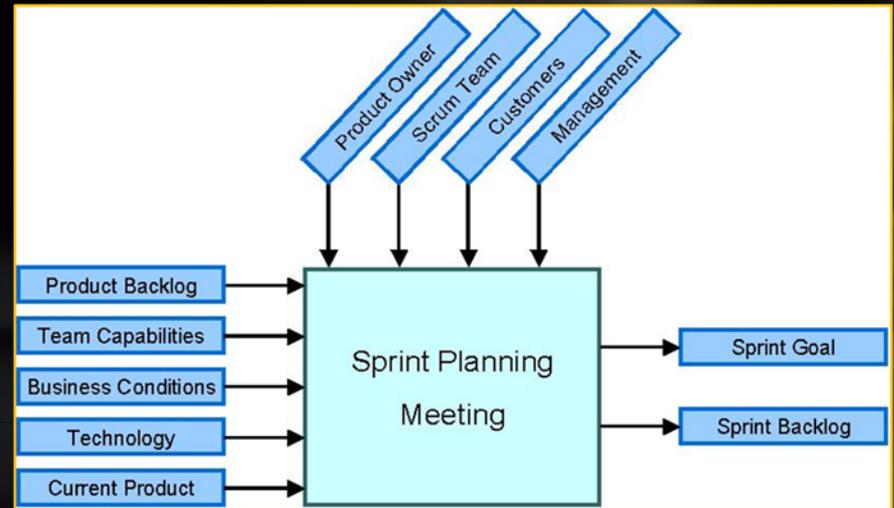
# Planificación del Sprint

Denominado también “**Sprint Planning Meeting**”, tiene como finalidad realizar una reunión, en la que participarán el Product owner, el Scrum Master y el equipo, con la intención de seleccionar de la lista Backlog del producto las funcionalidades sobre las que se va a trabajar, y que darán valor al producto.

Antes de comenzar la reunión el **Product Owner** tendrá que preparar el Backlog.

La reunión se realiza en con time-box de ocho horas que se divide en 2 partes de 4 horas.

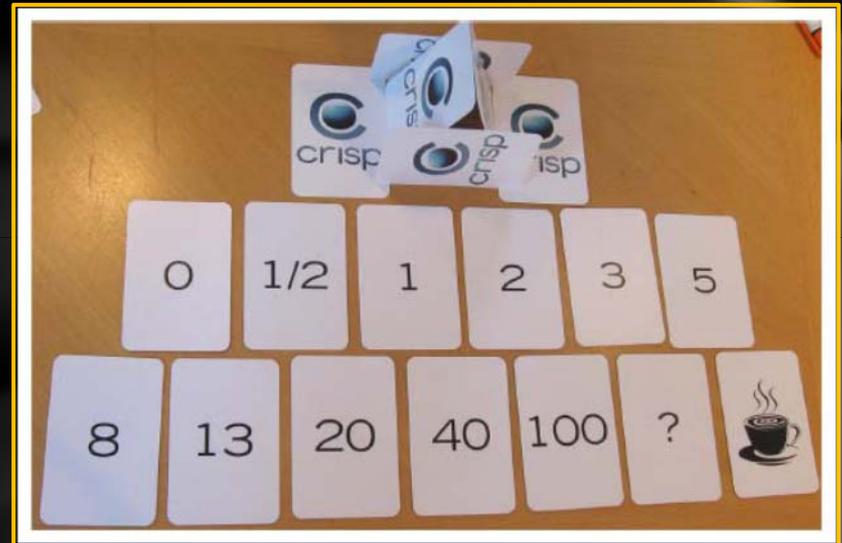
El progreso de las tareas en las que se mide la velocidad y el progreso del Sprint con respecto a las horas, se realizará mediante gráficos Burndown Chart.



# Estimaciones

## Planificación de Pocker

- Cada miembro del equipo tendrá una baraja de 13 cartas.
- Se propone una historia, el miembro del equipo selecciona una carta y la coloca boca abajo. Esta carta representará su estimación para la historia propuesta.
- Cuando todos los miembros han seleccionado su carta, se le da la vuelta al mismo tiempo.
- Se comprueban las estimaciones, y si hay muchas discrepancias se discute sobre esas diferencias y se ponen en común las ideas sobre la naturaleza del trabajo. Este proceso se repetirá hasta que las estimaciones sean parejas o aproximadas.
- Se comprueban las estimaciones y si hay muchas discrepancias se discute sobre esas diferencias y se ponen en común las ideas sobre la naturaleza del trabajo. Este proceso se repetirá hasta que las estimaciones sean parejas o aproximadas.
- El tiempo estimado es para el desarrollo de toda la historia, por ese motivo la secuencia de números no es lineal y, por ejemplo, hay un salto entre 40 y 100. De esta manera se evita sensaciones falsas para estimaciones grandes.



# Estimaciones del Sprint

## VELOCIDAD ESTIMADA

$$\begin{aligned} & (\text{Días} - \text{hombre disponibles}) \\ & * (\text{factor de dedicación}) \\ & = \text{VELOCIDAD ESTIMADA} \end{aligned}$$

El factor de dedicación se basa en estimar el estado del equipo, si es bajo entonces se espera encontrar dificultades.

La forma más adecuada para determinar factores de dedicación razonable es el estado de los últimos Sprints.

## FACTOR DE DEDICACIÓN

$$\begin{aligned} & (\text{Factor de dedicación}) \\ & = \frac{\text{Velocidad Real}}{\text{Días} - \text{hombre disponible}} \end{aligned}$$

## EJEMPLO:

Si tenemos un equipo que completó 20 historias de usuario, con 3 personas sumando 35 días-hombre, para calcular el próximo Sprint en el que habrá un componente más, sumando 45 días-hombre y teniendo en cuenta que en el último Sprint se completaron 20 puntos, el resultado sería:

$$57\% = \frac{20 \text{ ptos}}{35 \text{ días-hombre}}$$

$$45 \text{ días} - \text{hombre} * 57\% = 25 \text{ ptos.}$$

# El desarrollo del Sprint

## Reunión de Planificación (Sprint Planning Meeting)

Definirán qué tareas se tienen que realizar y cuáles son los objetivos.

Una vez definidos, el equipo comienza su desarrollo, pero teniendo en cuenta una serie de normas:

- **El equipo puede realizar consultas de agenda fuera del Sprint.**
- **No se permite a nadie gobernar al equipo durante el Sprint. El equipo se autogestionará.**
- **Si durante el desarrollo del Sprint no se puede realizar, porque no es viable, se puede realizar una nueva planificación para realizar un nuevo Sprint.**
- **Si el equipo no puede comprometerse a cumplir todo el Backlog, realizará una consulta con el Product Owner para decidir qué ítems eliminar.**
- **Si de la misma manera, el equipo se ve capaz de realizar más ítems del Backlog durante el Sprint, que el indicado inicialmente, consultará también con el Product Owner qué ítems se podrán añadir.**

# El desarrollo del Sprint

## Reunión Diaria (Sprint Daily Meeting)

En esta reunión, los componentes del equipo comparten información relativa al desarrollo y colaborarán para hacer las adaptaciones necesarias, aumentando así su productividad.

En esta reunión se tendrá como referencia el Backlog del Sprint y el equipo gráfico burn-down con la información de la reunión anterior y, además, qué tareas hizo cada persona del equipo. La reunión no podrá consumir más de 15 minutos y contestará a tres preguntas básicas:

- **¿Qué se ha hecho de nuevo con respecto a la última reunión diaria?**
- **¿Qué será lo siguiente a realizar?**
- **¿Qué problemas hay para realizarlos?**

Se usará como herramienta de apoyo, con la lista de tareas del Sprint actualizada y con el esfuerzo pendiente de cada tare. También se tendrá un gráfico con las tareas pendientes en la iteración .

# El desarrollo del Sprint

## Reunión Revisión del Sprint (Sprint Review Meeting)

En esta reunión, los desarrolladores presentan el producto entregable que han implementado y, los gestores, clientes, usuarios y Product Owner analizan esa entrega y escuchan al equipo sobre los problemas que han tenido durante el proceso.

Esta reunión servirá para tomar decisiones que ayudan a escoger el camino más adecuado para alcanzar las metas.

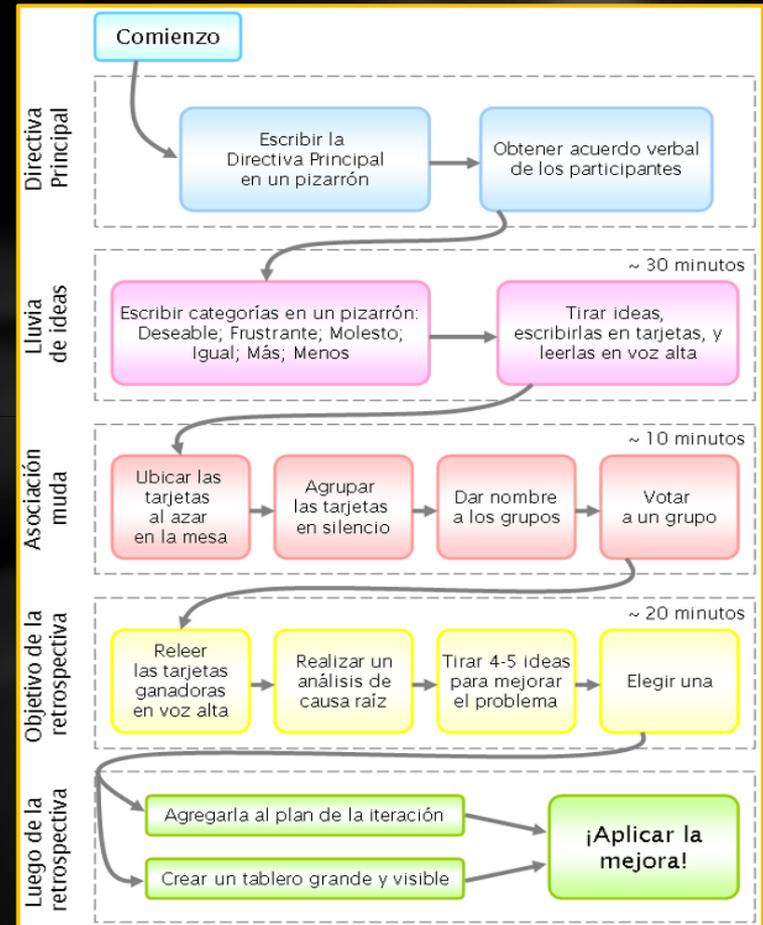


# El desarrollo del Sprint

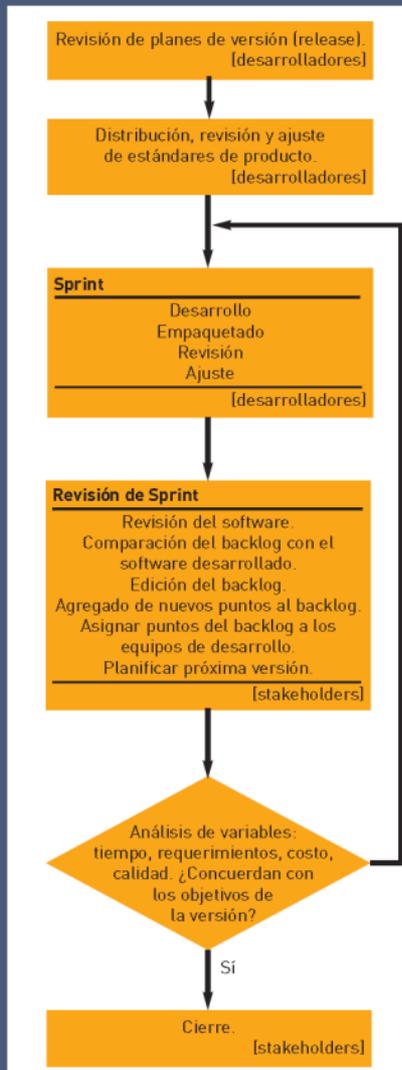
## Reunión de Retrospectiva (Sprint Retrospective Meeting)

En esta reunión, el equipo debatirá temas relacionados con el Sprint recientemente finalizado y los cambios que se podrían hacer para mejorar el próximo Sprint y que sea más productivo.

Generalmente, será el ScrumMaster quién realiza esta reunión y tendrá una duración máxima de 3 horas.



# Diagrama detallado de las fases de Scrum



## REVISIÓN DE PLANES DE VERSIÓN:

Se revisa que hay que hacer y en que punto está la distribución actual.

## SPRINT:

Es la fase de desarrollo iterativa.

Desarrollo: Análisis, implementación, testing.

Empaquetar: Generar paquetes ejecutables

Revisión: Resolución de problemas y se añaden nuevos ítems.

Ajustes: Uso de los ajustes para mejorar el producto.

## SPRINT REVIEW:

Después del Sprint se hace una reunión con el ScrumMaster donde se revisa el producto del Sprint anterior y en el que se pueden añadir puntos nuevos al backlog.

## Cierre:

En esta fase se encuentran las típicas actividades de fin de proyecto como, hacer una versión distribuable, testear, marketing etc....