



**Unkasoft**<sup>®</sup>  
PLATFORM



**Navegapolis.net**



**Qualitatis**<sup>α</sup>  
Open Knowledge

# Metodologías ágiles para el desarrollo de software **SCRUM + XP**

Autor: <http://www.lawebdejm.com>

# Agenda



n **Introducción**

n **La corriente ágil**

n **Gestión y planificación con SCRUM**

n **Desarrollo de software con XP**

n **Ejercicios**

# Agenda



n **Introducción**

n La corriente ágil

n Gestión y planificación con SCRUM

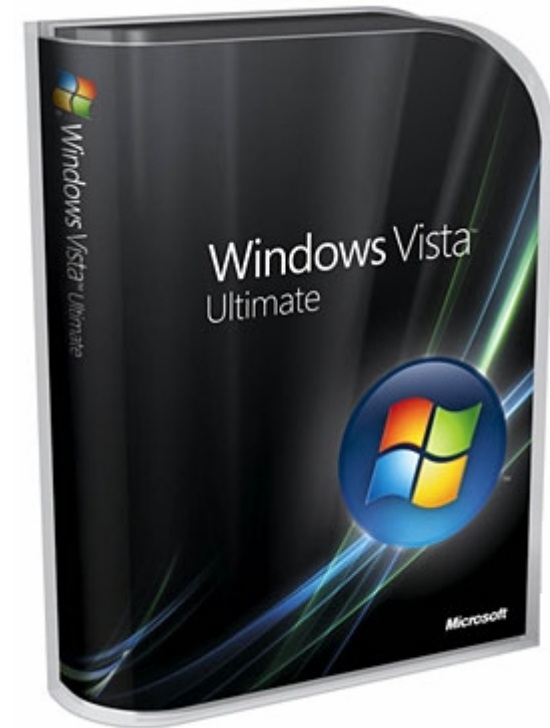
n Desarrollo de software con XP

n Ejercicios

# Introducción:

## El software no van bien

1. No cumple con lo esperado
2. Es más caro de lo que pensábamos
3. Tardamos más en desarrollarlo de lo que creíamos
4. Cada vez es más complejo

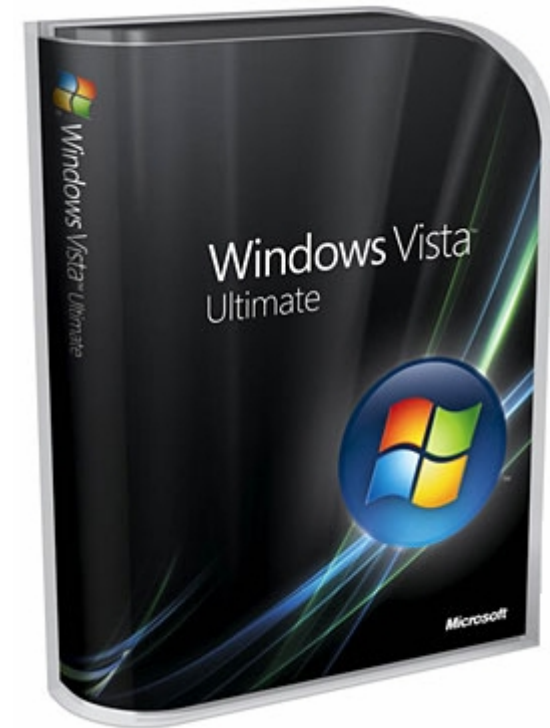


# Y para muestra... un “botón”

## El botón “Inicio” de Windows Vista



- 43 personas involucradas
- 1 año completo de desarrollo
- Apagado del sistema: 15 opciones
- Retraso total: 2 años



# Cambio de mentalidad



- En el proceso de desarrollo
- En las personas involucradas
- En la gestión del proyecto



# Agenda



n Introducción

n **La corriente ágil**

n Gestión y planificación con SCRUM

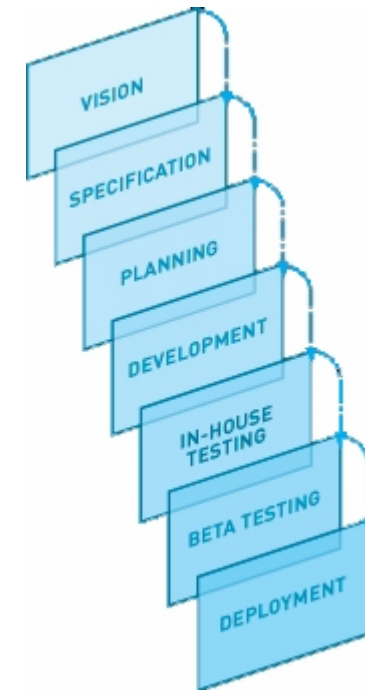
n Desarrollo de software con XP

n Ejercicios

# Ingeniería de software clásica



- Heredado del entorno industrial
- Entorno predictivo e inmutable
- Organización en cascada
- Ejecución controlada
- Cambios costosos o imposibles



# Gestión de proyectos ágil



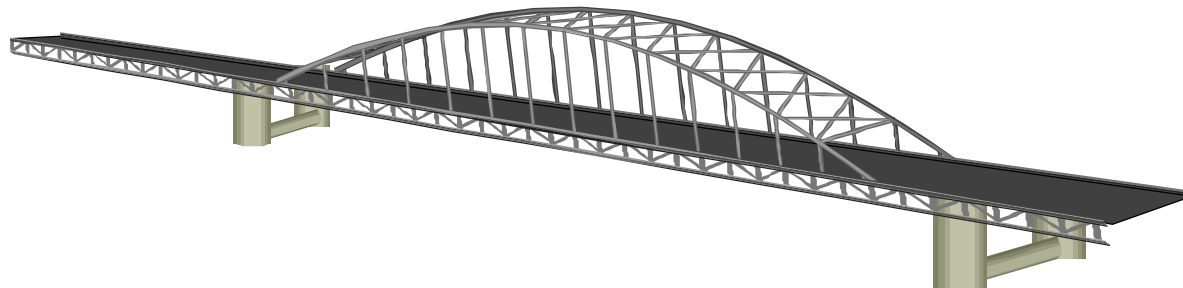
- El software no sigue las mismas leyes
- Entorno impredecible, cambiante y con alta innovación
- Cambios sencillos
- Organización inversa
- Operario controlado = Operario desmotivado



# Proyectos clásicos



- Requisitos iniciales estables y conocidos
- Tiempo suficiente para todas las fases
- Proyectos conocidos y/o repetitivos
- Cambios imposibles



# Proyectos ágiles

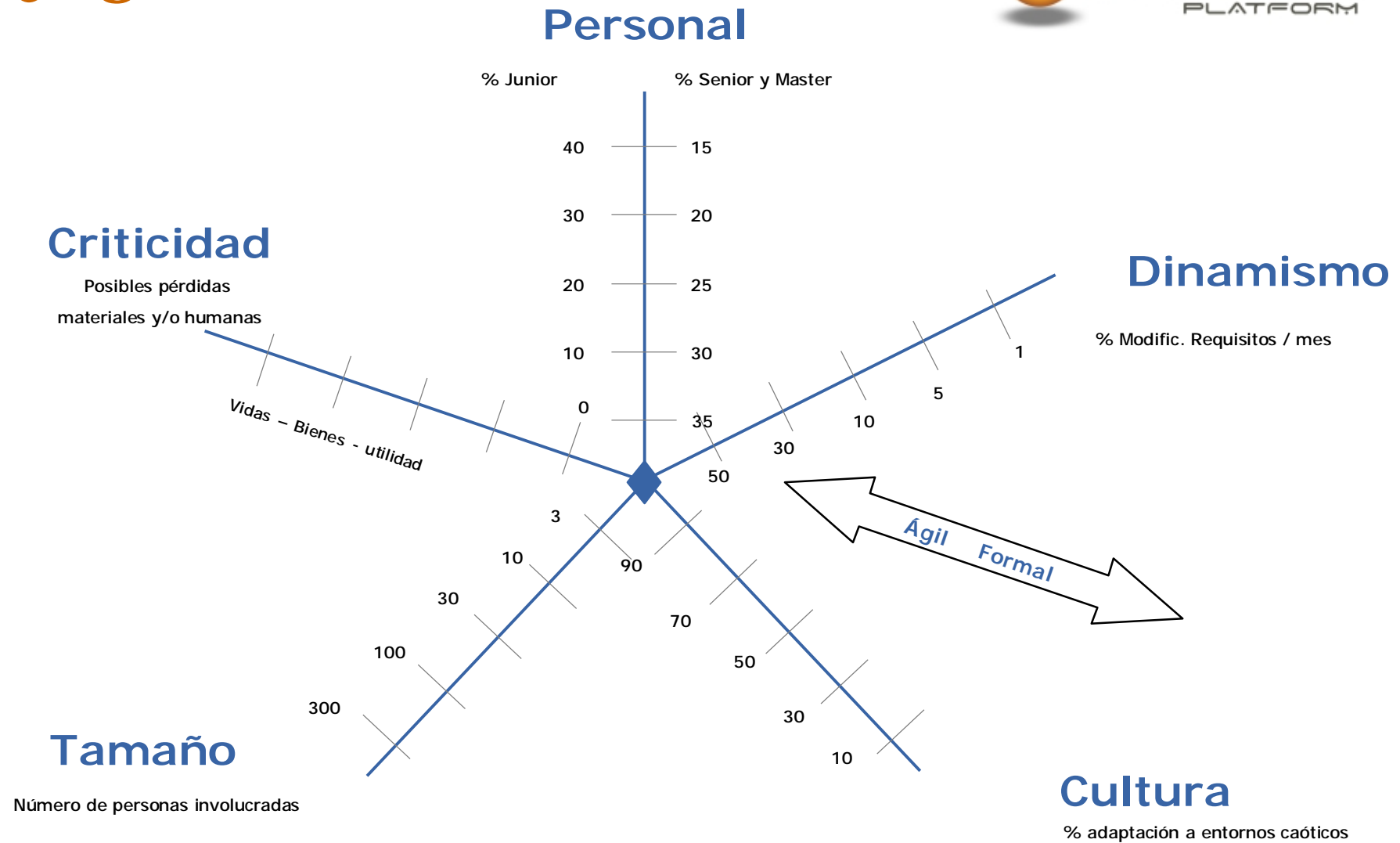


- Lanzamiento constante de novedades
- Requisitos iniciales desconocidos/inciertos
- Fecha de entrega ajustada
- Alta innovación
- Prototipado

iPod de Apple:  
6 versiones en 6 años



# ¿Ágil o clásico?



---

# ¿Ágil o clásico?



## Que no nos cieguen las modas

# Comparativa



## Clásico

**Requisitos detallados**

**Planificación estricta**

**Requisitos no-cambiantes**

**Seguimiento y control**

**División y especialización**

## Ágil

**Visión general del producto**

**Adaptación a la situación**

**Evolución constante**

**Autogestión**

**Equipo multi-disciplinar**

# Manifiesto ágil - 2001



**Preferimos**

**Desconfiamos**

**A las personas y su comunicación**

**Los procesos y las herramientas**

**El software que funciona**

**La documentación exhaustiva**

**La colaboración con el cliente**

**La negociación contractual**

**La respuesta al cambio**

**Seguimiento de un plan**

<http://agilemanifesto.org/>

# Agenda



n Introducción

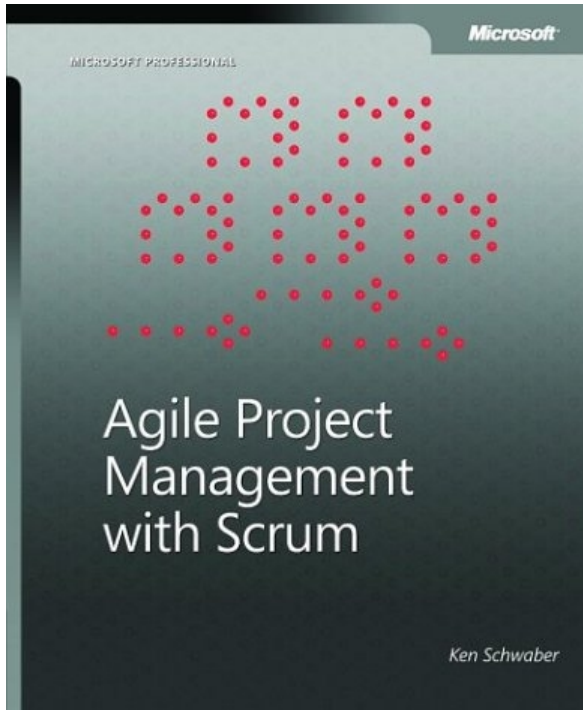
n La corriente ágil

n **Gestión y planificación con SCRUM**

n Desarrollo de software con XP

n Ejercicios

# SCRUM



Metodología ágil para la  
*planificación y seguimiento*  
de proyectos de todo tipo

<http://www.controlchaos.com>

---

# SCRUM



Surge del estudio de varios proyectos y productos exitosos y su adaptación a la industria del software:

- Industria japonesa: Toyota, Honda, Fuji-Xerox
- Borland Quattro Pro

**¡¡Basado en la teoría del caos!!**

# SCRUM



Canon

Google™

NEC



ORACLE



NOKIA  
CONNECTING PEOPLE

YAHOO!



Microsoft®

3M

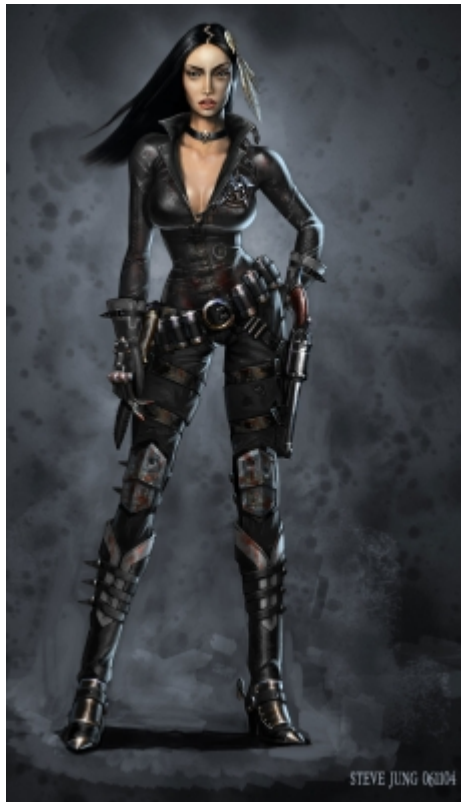


EPSON

# SCRUM en los videojuegos



La empresa HighMoon Studios es pionera en la implantación de Scrum en su proceso de desarrollo.



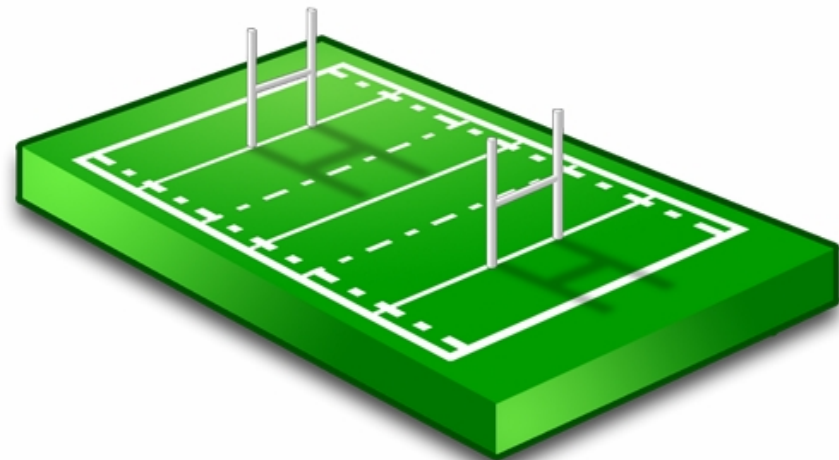
---

# SCRUM



**Reduce al máximo la burocracia y las actividades no orientadas a producir software**

**Produce resultados visibles en periodos cortos (cada 30 días)**



---

# SCRUM: Starring



## ***Product owner***

Decide en todo momento las funcionalidades que se deben añadir al producto



## ***Interesados***

Se beneficiarán del producto en desarrollo.

Usuarios finales del sistema.

# SCRUM: Starring II



## ***Scrum Master***

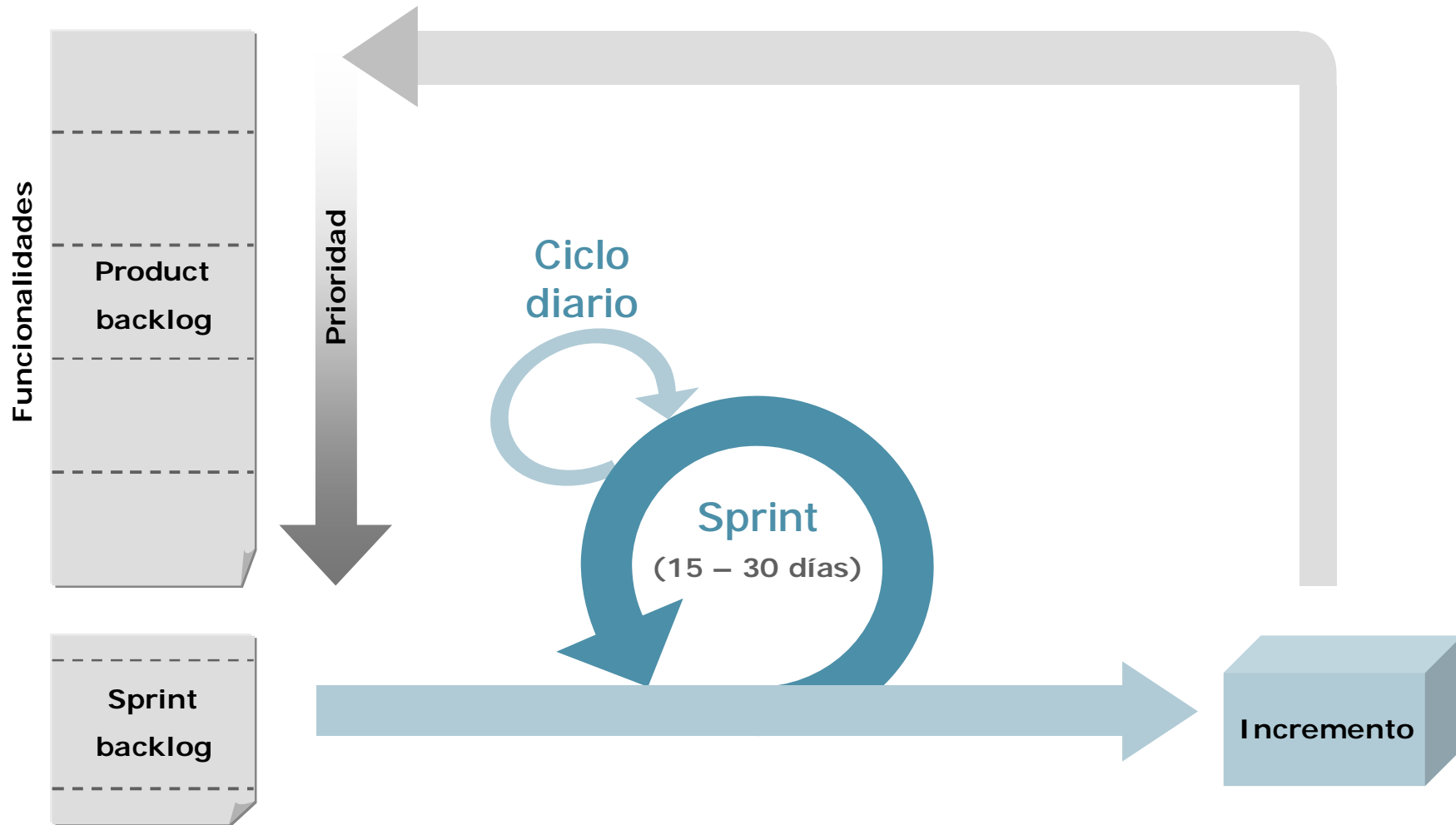
Se asegura del seguimiento de la metodología. Guía las reuniones. Ayuda ante problemas en el equipo.



## ***Equipo de desarrollo***

Responsables de implementar las funcionalidades escogidas por el product owner.

# Ciclo de SCRUM

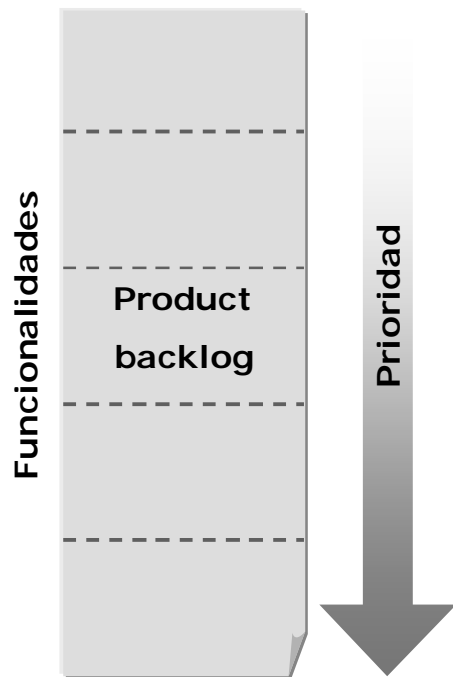


# Product backlog



Listado de todas las funcionalidades a implementar.

El product owner es el responsable de mantener esta lista ordenada (por prioridad) y actualizada.



	Item#	Description	Components	By
<b>Critical</b>				
	2	Update devices server	Server	JM
	2.7	* Install and test	Server	JM
	5	Download server	Server, Packager	Jaime
	6.20	* Auto download patch	Download wurfl patch	
	6.30	* Licenses server (web & services)		
	6.40	* Modify build process		
<b>Very High</b>				
	6	Publishers templates	Packager	JM
	4	Screens		
	4.1	Analysis	IDE, Engine	JM
	4.2	New screen wizard	IDE, Engine	JM
	4.3	Object management (load, save, remo	IDE, Engine	JM

# Sprint backlog

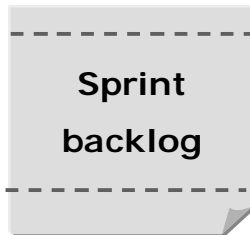


Listado de todas la funcionalidades que estarán listas para la próxima versión.

El product owner, ayudado por el equipo de desarrollo, es el responsable de elaborar esta lista.



Se marcan una serie de objetivos internos para el sprint.



Sprint backlog				
Backlog	Tarea	Tipo	Estado	Responsal
	Change building process to include all	Code	Completed	JM
70.9	Launch ANT scripts	Code	Working	JM
70.9	Change game wizard to add ANT scrip	Code	Pending	Jaime
70.9	Ask for available ANT targets	Code	Pending	Jorge
70.10	Select files to use	Code	Working	Jaime
	GUI - Check box list	Code	Cancelled	Jaime
	GUI - Extract from JAR	Code	Completed	JM
	GUI - Extract from engine	Code	Completed	JM
	Build properties file from A	Code	Completed	JM

# Sprint



Periodo de tiempo de que dispone el equipo para desarrollar la funcionalidad descrita en el sprint backlog.

Durante este tiempo, el Scrum Master debe congelar todo el trabajo extra para el equipo.

El equipo no puede salirse del sprint.



# Sprint: ciclo diario



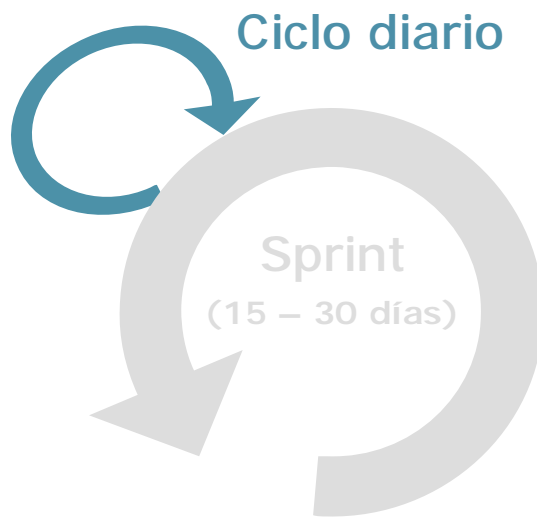
Cada día comienza con una reunión del equipo. El Scrum Master asiste de “oyente”.

Cada integrante del equipo explica a sus compañeros:

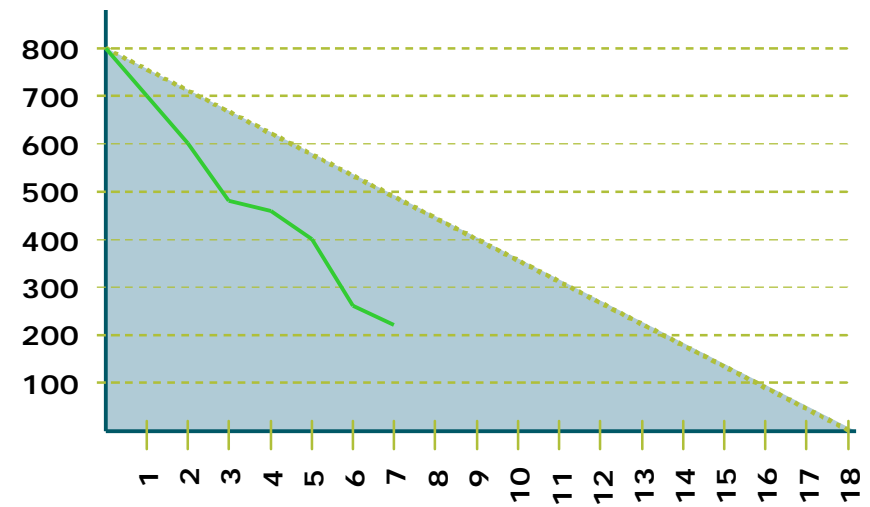
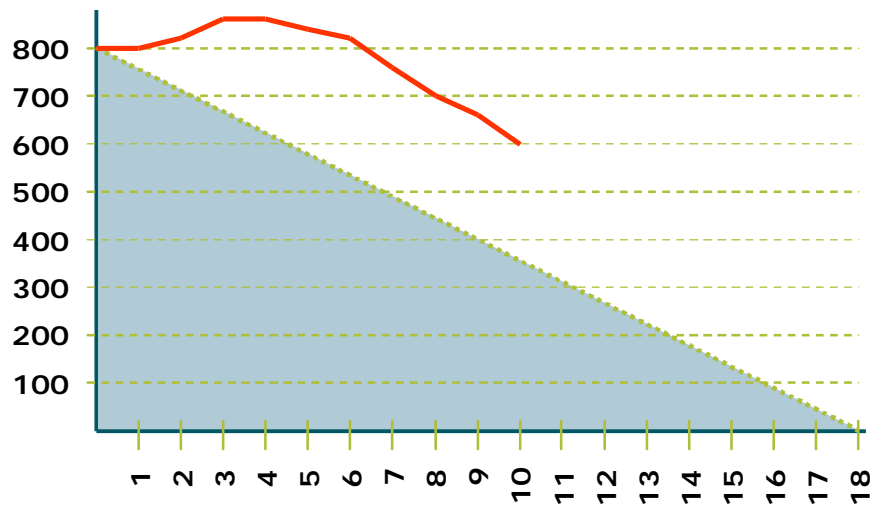
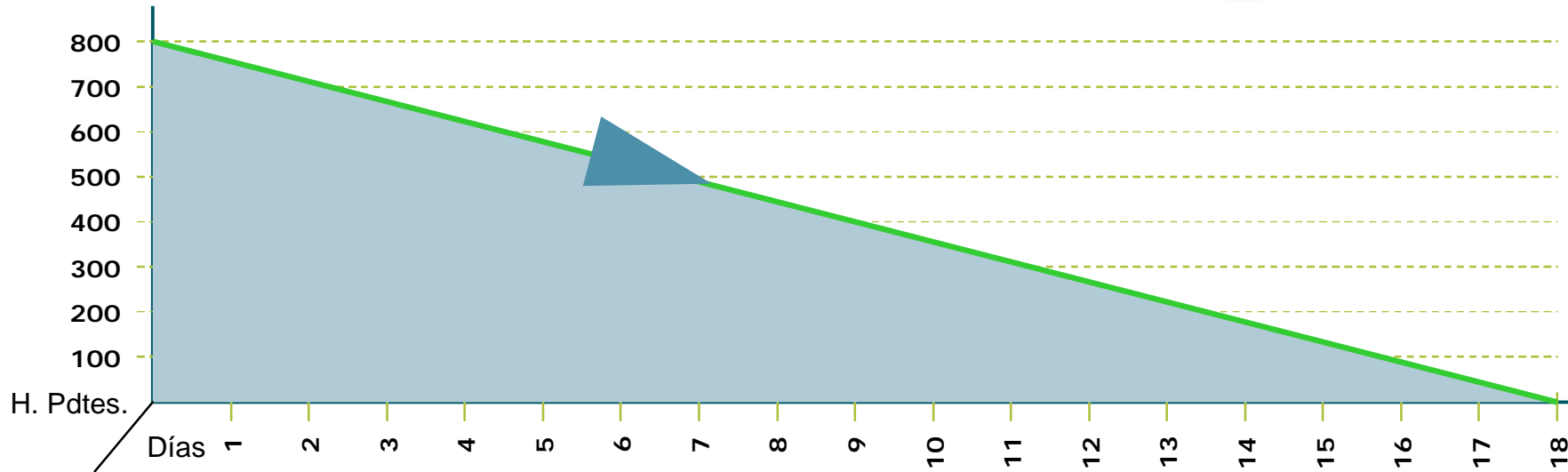
- n ¿Qué hice ayer?
- n ¿Qué voy a hacer hoy?
- n ¿Qué ayuda necesito?



- n El Scrum Master elimina obstáculos



# Avance diario



# Fin del sprint



El equipo muestra las nuevas funcionalidades al product owner y a los interesados.

El product owner apunta nuevas sugerencias, mejoras o funcionalidades que puedan surgir.



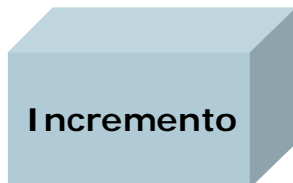
# Reunión retrospectiva



Product owner y equipo revisan los objetivos marcados para el sprint.



- n Se apuntan los aspectos **positivos** del sprint: intentar **repetirlos**.
- n Se apuntan los aspectos **negativos** del sprint: intentar **evitarlos**.



# Claves



n Responsables últimos del sprint

n Saben hacer su trabajo: libertad total



n Protege al equipo de presiones externas

n Ayuda en la logística



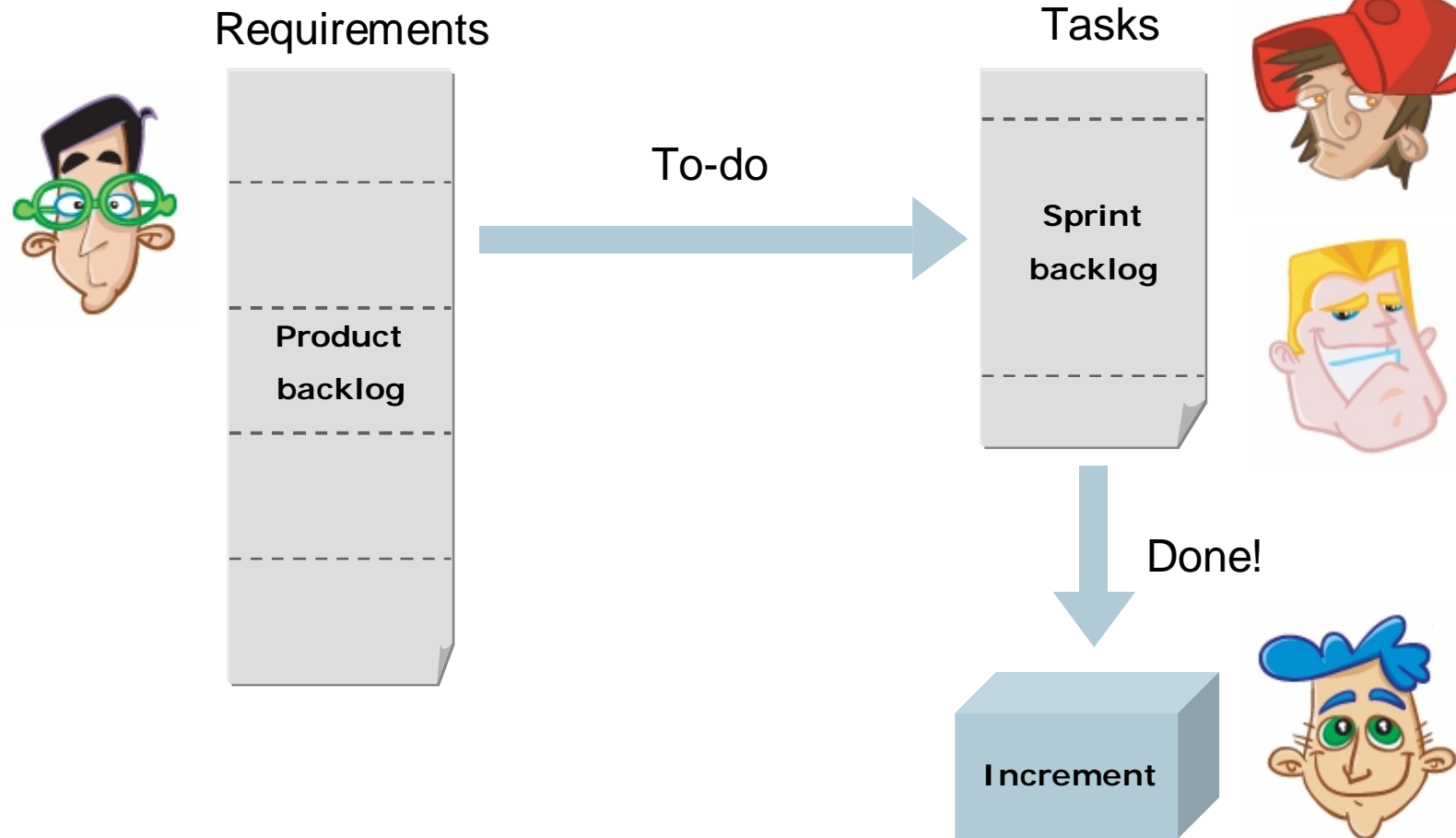
n Conoce y marca las prioridades del producto

n Valida el resultado de cada sprint



n Aporta ideas y sugerencias

# Ciclo de SCRUM



# Agenda



n Introducción

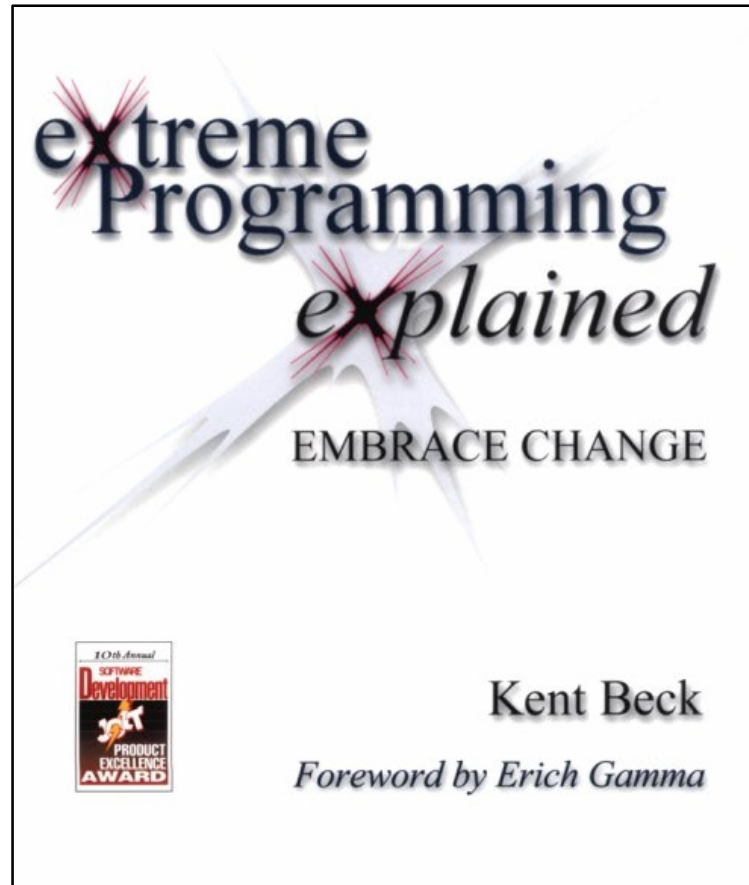
n La corriente ágil

n Gestión y planificación con SCRUM

n **Desarrollo de software con XP**

n Ejercicios

# eXtreme Programming



**Metodología ágil para el  
desarrollo de software**

**Consta de 12 prácticas  
de programación**

<http://www.extremeprogramming.org>

## Codificación

- n Diseño sencillo
- n Refactorizaciones
- n Pruebas unitarias
- n Estándares

## Desarrollo

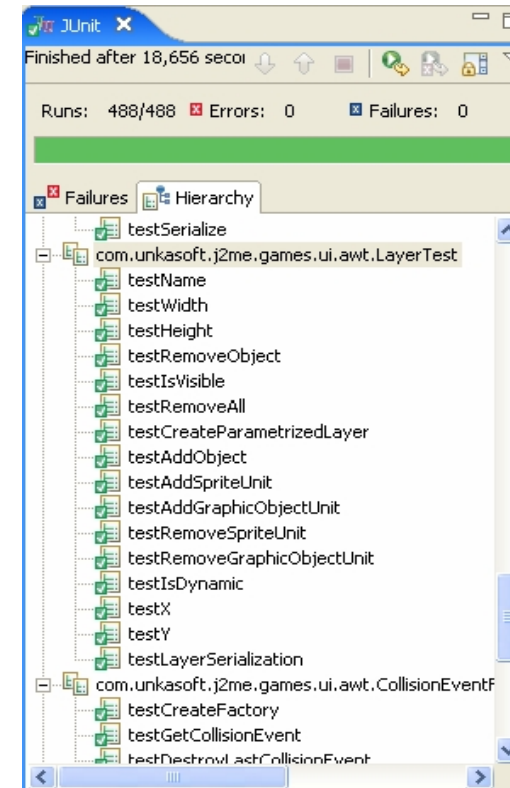
- n Programación por pares
- n Propiedad colectiva de código
- n Integración continua
- n Metáfora común

# eXtreme Programming



## Negocio

- n Cliente in-situ
- n Juego de la planificación
- n Entregas frecuentes
- n 40 horas semanales



# SCRUM + XP



+ Planificación y seguimiento con SCRUM  
+ Programación con las prácticas de XP

---

Solución más adoptada en la industria



# Información adicional



<http://www.unkasoft.com>

<http://eskasiunblog.blogspot.com>

---

Q&A



¿Dudas?

¿Sugerencias?

¿Ideas?

¿Aclaraciones?

¿Críticas?

¿Preguntas?

<http://www.lawebdejm.com>

# Agenda



n Introducción

n La corriente ágil

n Gestión y planificación con SCRUM

n Desarrollo de software con XP

n **Ejercicios**