

# Metodologías Ágiles: Scrum y técnicas de estimación ágil

PreparaTIC - Junio 2009

Jorge Manrubia Díez  
[jorge.manrubia@giss.seg-social.es](mailto:jorge.manrubia@giss.seg-social.es)



¿Por qué?



Hacer un programa es cómo...

*Can you get a design that is capable of turning the coding into a **predictable** construction activity? And if so, is cost of doing this sufficiently small to make this approach worthwhile?*

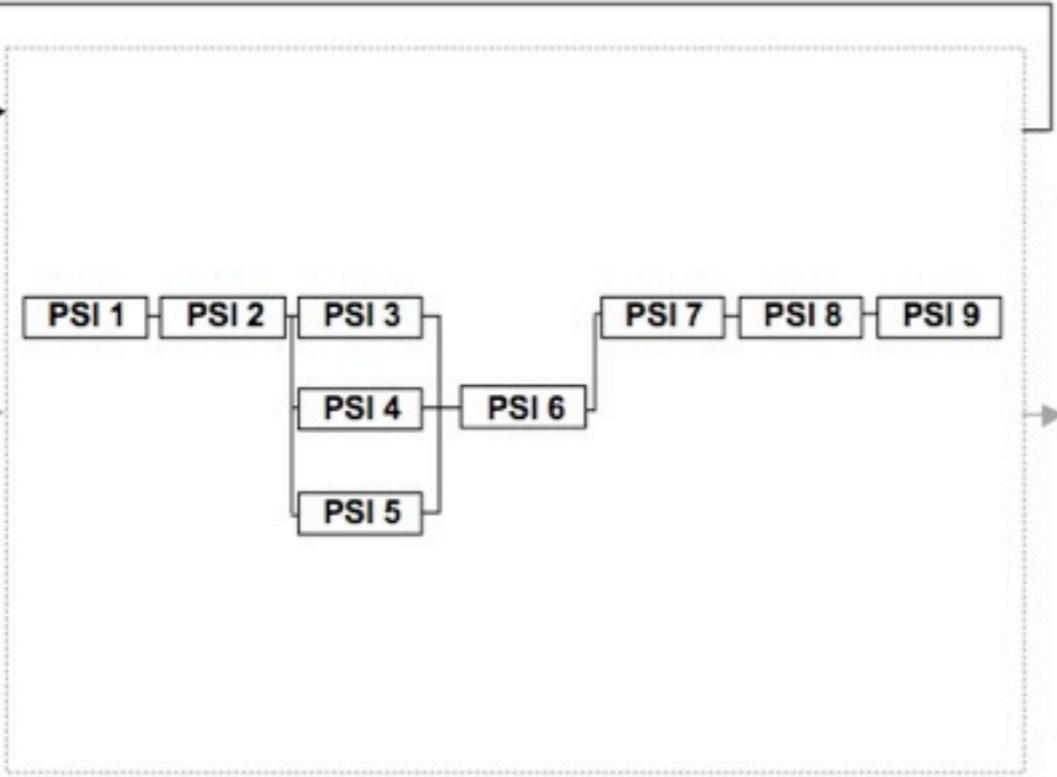
Martin Fowler, "The new methodology" (2005)

*They were amazed that my industry (software development) was trying to do its work using a completely inappropriate process control model. (...) They said that all complex processes that weren't completely understood required the **empirical model** (not the **defined process control model**)*

Conclusiones de expertos en teoría de control de procesos liderados por Babatunde Organnaiké tras analizar los procesos de desarrollo software. Ken Schwaber, "Agile Software Development with Scrum" (2001)

**Entradas Externas**

- Solicitud Formal del PSI
- Estructura Organizativa
- Información Relevante
- Entorno Tecnológico Actual y Estandar



-Requisitos del PSI

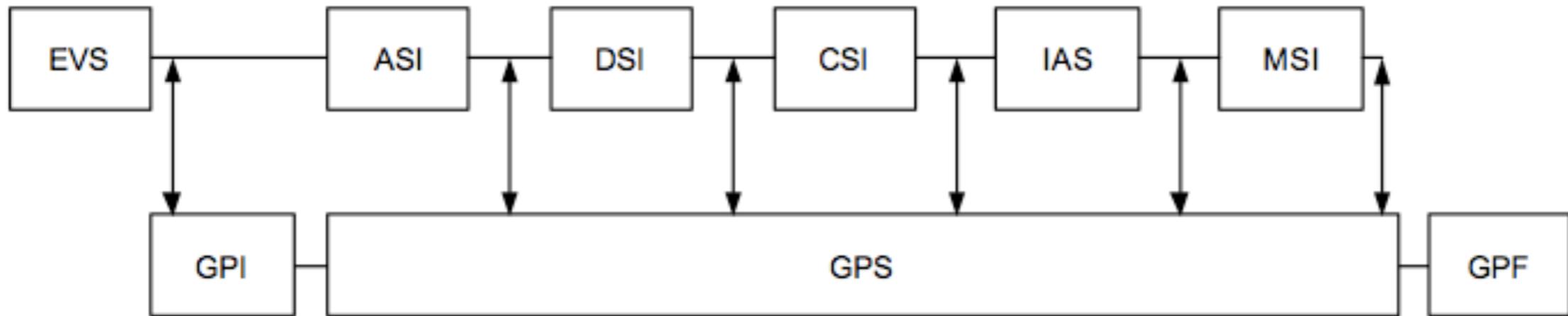
-Arquitectura de Información:

- oModelo de Información
- oModelo de Sistemas de Información
- oArquitectura Tecnológica

-Plan de Acción:

- oPlan de Proyectos
- oPlan de Mantenimiento

**ESTUDIO DE VIABILIDAD DEL SISTEMA**



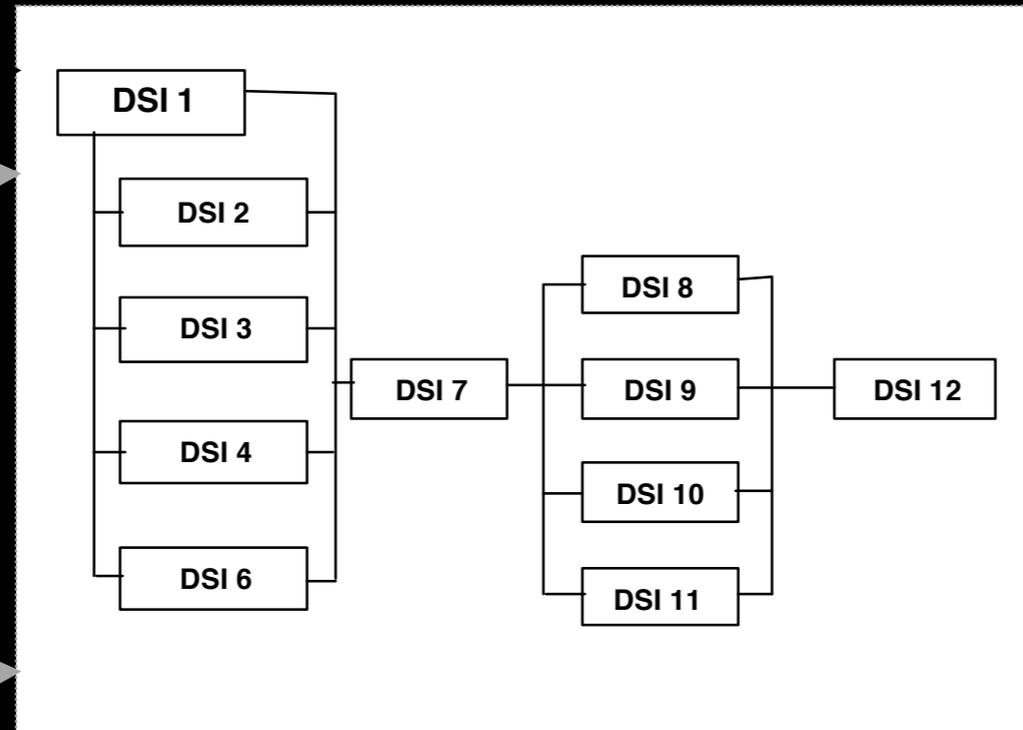
## DISEÑO DEL SISTEMA DE INFORMACIÓN (ORIENTACIÓN A OBJETOS)

### Resultados del Análisis de Sistema de Información (Orientación a Objetos)

- Catálogo de Requisitos
  - Contexto del Sistema
  - Modelo de Casos de Uso
  - Modelo de Clases de Análisis
  - Modelo de Procesos
  - Descripción de subsistemas
  - Resultado del Análisis de Consistencia
  - Interfaz de usuario
  - Plan de Pruebas
- Especificación de Requisitos Software (ERS)*

### Entradas Externas

- Estándares y normativas de la instalación
- Características Específicas del SGBD o Sistemas de Ficheros a Utilizar
- Estructura de Datos del Sistema Origen



- Diseño de la Arquitectura del Sistema
- Entorno Tecnológico, Seguridad, Operación y Administración
- Diseño Detallado de Subsistemas
- Diseño de la Realización de Casos de Uso
- Diseño de la Interfaz de Usuario
- Modelos de Clases de Diseño
- Modelo Físico de Datos
- Resultado Análisis de Consistencia
- Especificaciones de Construcción
- Plan de Migración y Carga inicial
- Especificación del Entorno, Niveles y Planificación de las Pruebas
- Requisitos de Implantación

### CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN

Adaptación continua  
del proceso

+

Excelencia técnica



WE'RE GOING TO TRY SOMETHING CALLED AGILE PROGRAMMING.

www.dilbert.com scottadams@aol.com



THAT MEANS NO MORE PLANNING AND NO MORE DOCUMENTATION. JUST START WRITING CODE AND COMPLAINING.

11-24-07 © 2007 Scott Adams, Inc./Dist. by UFS, Inc.



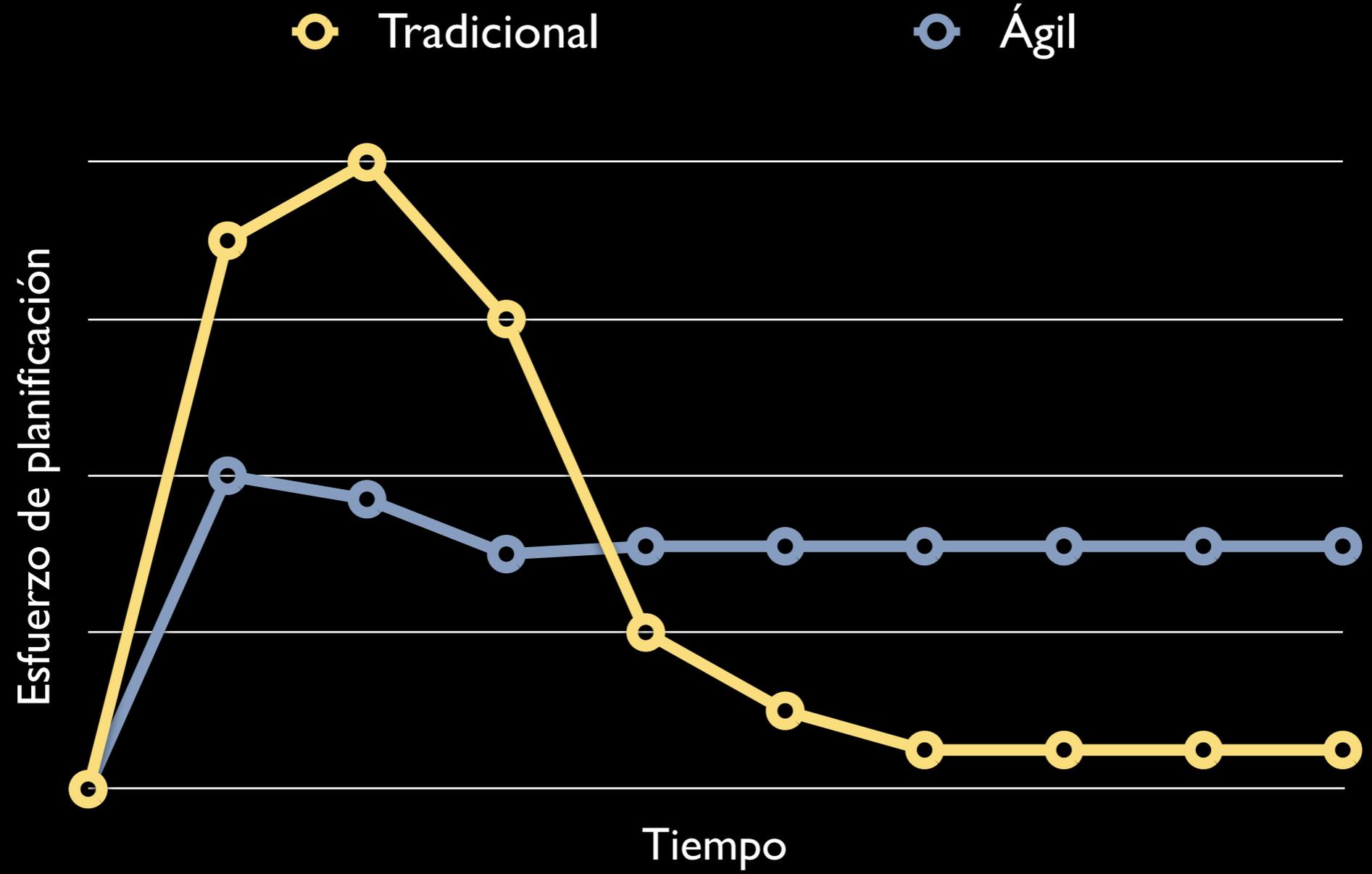
I'M GLAD IT HAS A NAME.

THAT WAS YOUR TRAINING.

© Scott Adams, Inc./Dist. by UFS, Inc.

© Scott Adams, Inc./Dist. by UFS, Inc.

Desarrollo ágil: lo que **no** es



El propio proceso se adapta durante el desarrollo

# Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

the right, we value the items on the left more.

- Liberación **temprana y continua** del software de **software funcional**
- Integración total de la **gente de negocio** en los equipos de desarrollo
- **Excelencia técnica**, buenos diseños y simplicidad
- Reuniones cara a cara, **comunicación oral**
- **Autoorganización** de los equipos de desarrollo

• • •



# SCRUM y Técnicas de Estimación Ágil

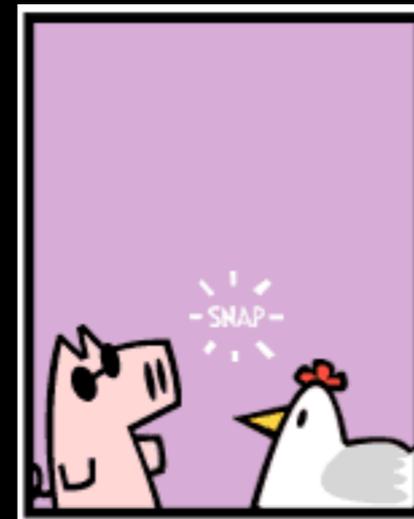
# Scrum

- Creada por Ken Schwaber, Mike Beedle y Jeff Sutherland (mediados de los 90)
- Equipos de ~7 personas
- Iteraciones (*sprints*) de 30 días



# Roles

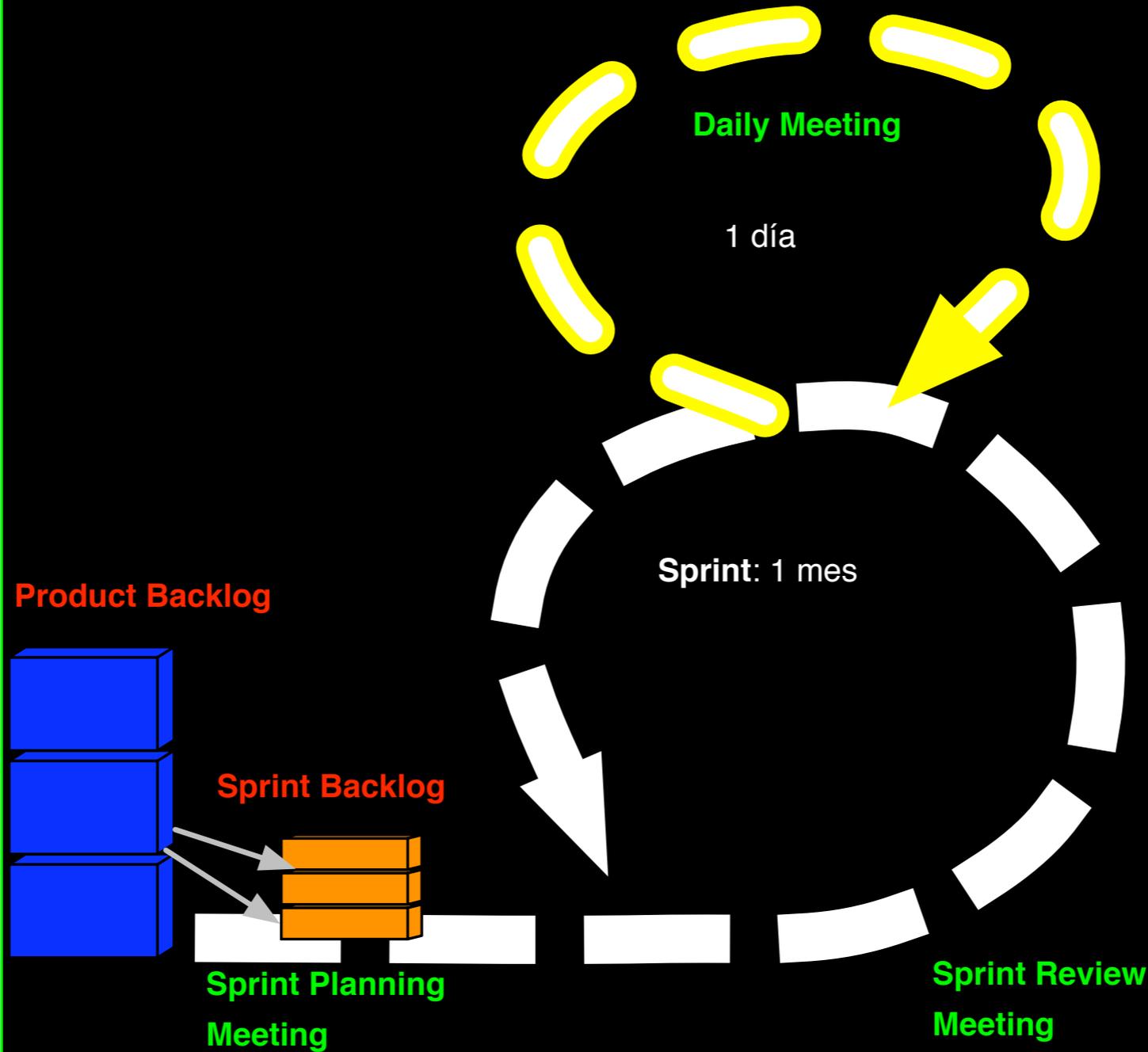
- Product Owner: representa al cliente
- Scrum Master (50% desarrollador)
- Scrum Team
- Chicken (otros)



## PRE-GAME

- Product Backlog Inicial
- Release Backlog
- Diseño alto nivel/arquitectura

## DESARROLLO (CICLO DE SPRINTS)



## POST-GAME

- Documentación de usuario
- Material de formación
- ...

DEFINIDO

EMPÍRICO

DEFINIDO

# Backlogs

	<b>Descripción</b>	<b>Items</b>	<b>Estimación</b>
<b>Product Backlog</b>	Lista priorizada de <u>funcionalidades de negocio</u> y técnicas a desarrollar	<ul style="list-style-type: none"><li>• <u>User stories</u></li><li>• Características</li><li>• Bugs</li></ul>	Story points
<b>Sprint Backlog</b>	Selección de items del PB y tareas para desarrollarlos	<ul style="list-style-type: none"><li>• Tareas</li></ul>	Horas

# Historias de usuario

Descripción escrita de una funcionalidad concreta a desarrollar

Como <tipo de usuario> quiero < poder hacer algo >

Como desarrollador quiero poder transformar  
un modelo físico obtenido de ADABAS a un  
modelo lógico

Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS a un modelo logico

*epic*



Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS traduciendo sólo las estructuras relacionales

Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS admitiendo campos múltiples

Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS admitiendo campos periódicos

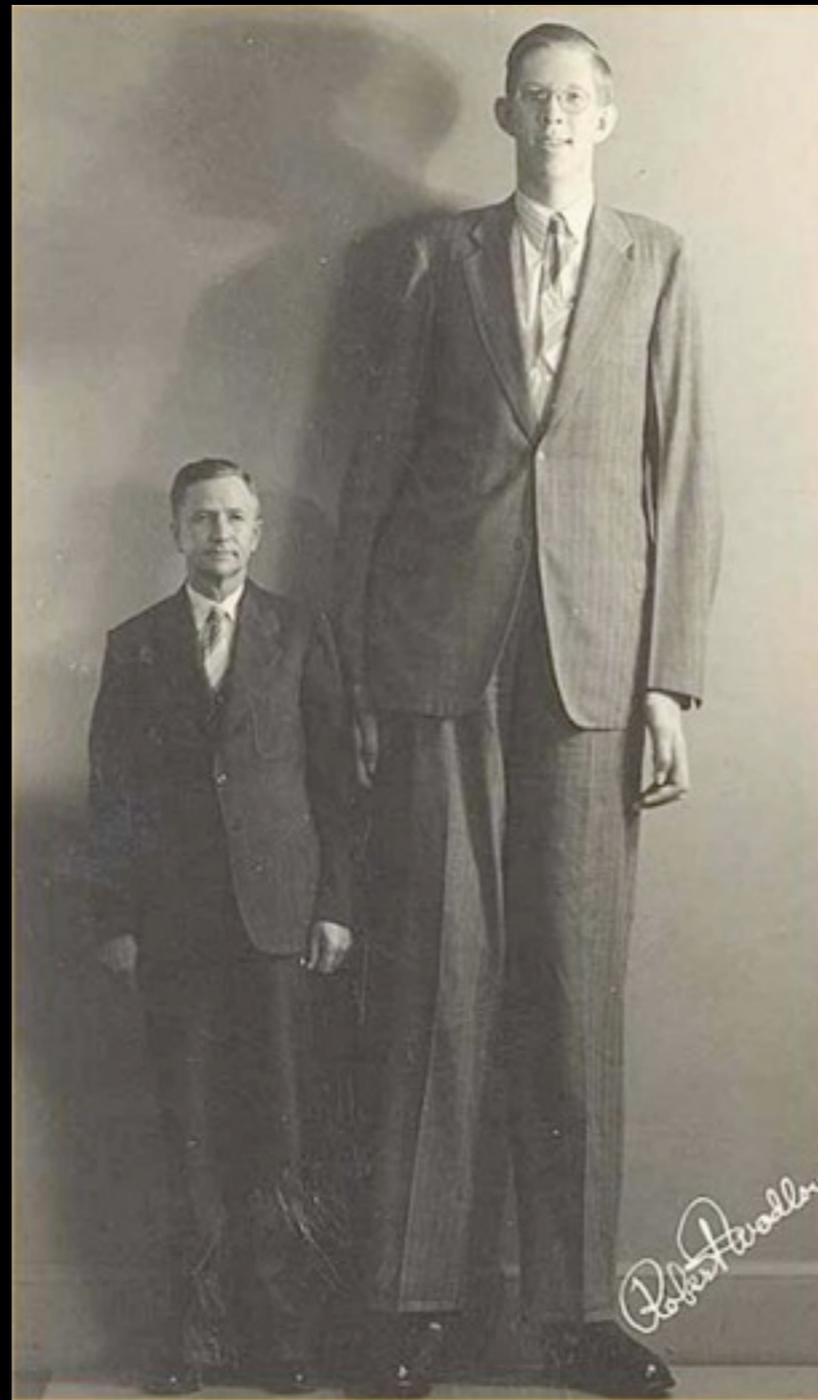
*theme*

Estimación de Tamaño

$\neq$

Estimación de Duración

Story Point = Esfuerzo necesario +  
complejidad + riesgo ...



¿Cuánto miden?  
¿Quién es más alto?

1, 2, 3, 5, 8, 13, 20, 30,  
50, 100

Escala de estimación

# Planning Poker

- La historia se presenta y se discute sobre ella
- Los miembros del equipo escogen carta con la estimación
- Todos dan la vuelta a la carta a la vez
- Los miembros con la estimación más baja y más alta exponen sus razones, y se repite el proceso de estimación



Velocidad del equipo =  
Story points / iteración



Características  
Deseadas

Estimar  
tamaño

Estimar  
duración

PLANIFICACIÓN

Historia 1	2
Historia 2	3
Historia 3	3
Historia 4	2
Historia 5	5
Historia 6	2
Historia 7	1
Historia 8	8

Product Backlog

Historia 1	2
Historia 2	3
Historia 3	3

Sprint Backlog 1

Historia 4	2
Historia 5	5
Historia 6	2

Sprint Backlog 2

Historia 7	1
Historia 8	8

Sprint Backlog 3

Tiempo

Velocidad = 8 story points / iteración

Cómo usuario del plug-in quiero poder obtener un LDM a partir de un modelo de la capa conceptual Rochade	18.0h
Implementar Asistente de selección de agrupación conceptual	8
Prueba funcional	4
Implementar Asistente de selección de fichero destino	4
Implementar la acción	2
Implementar la acción	5
Implementar Asistente de selección de fichero destino	4

Las historias seleccionadas para el sprint se desglosan en tareas (estimación en horas)

Diariamente los  
desarrolladores actualizan su  
estimación de esfuerzo restante



Gráfico de avance del proyecto (*Release Burndown Chart*)

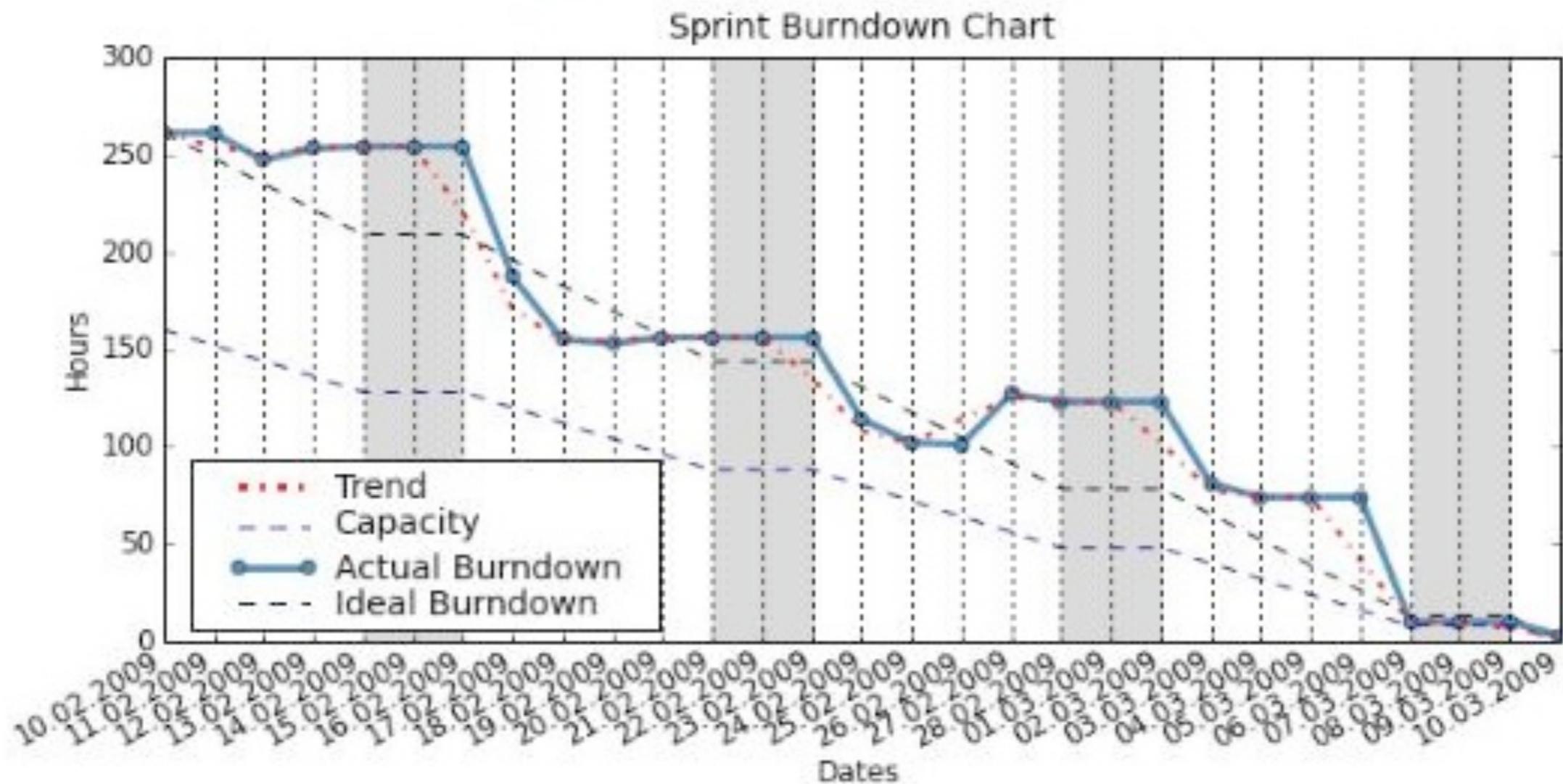


Gráfico de avance del sprint (*Iteration Burndown Chart*)

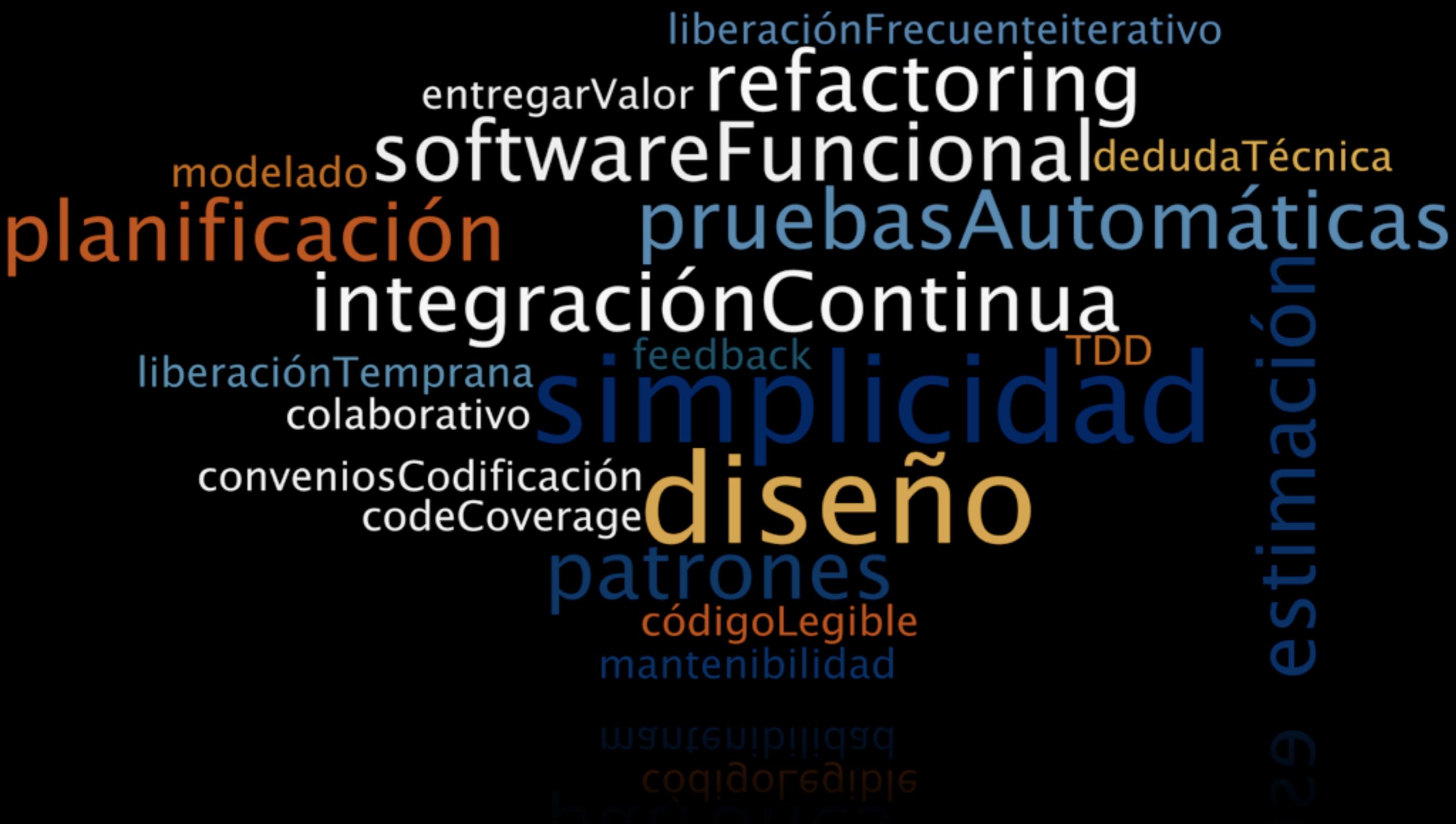
# Daily meeting

- ¿Qué has hecho desde ayer?
- ¿Qué vas a hacer hoy?
- ¿Qué problemas tienes?

**15 min. máximo**



**Excelencia técnica**



liberación Frecuente iterativo

entregar Valor

refactoring

modelado

software Funcional

deduda Técnica

planificación

pruebas Automáticas

integración Continua

liberación Temprana colaborativo

feedback

TDD

simplicidad

convenios Codificación code Coverage

diseño

patrones

código Legible

mantenibilidad

mantenibilidad

código Legible

estimación

# Pruebas

- Exhaustivas
- Automáticas

# Pruebas automáticas

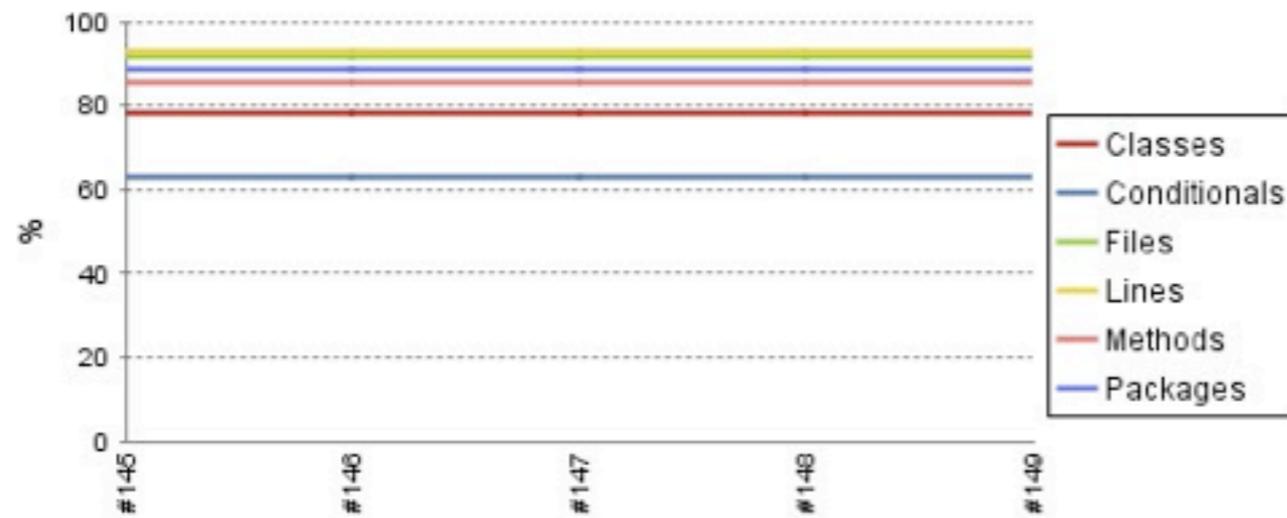


¿Cuánto? → Herramientas  
de cobertura de código  
(*code coverage*)

# Code Coverage

## Cobertura Coverage Report

### Trend



### Project Coverage Summary

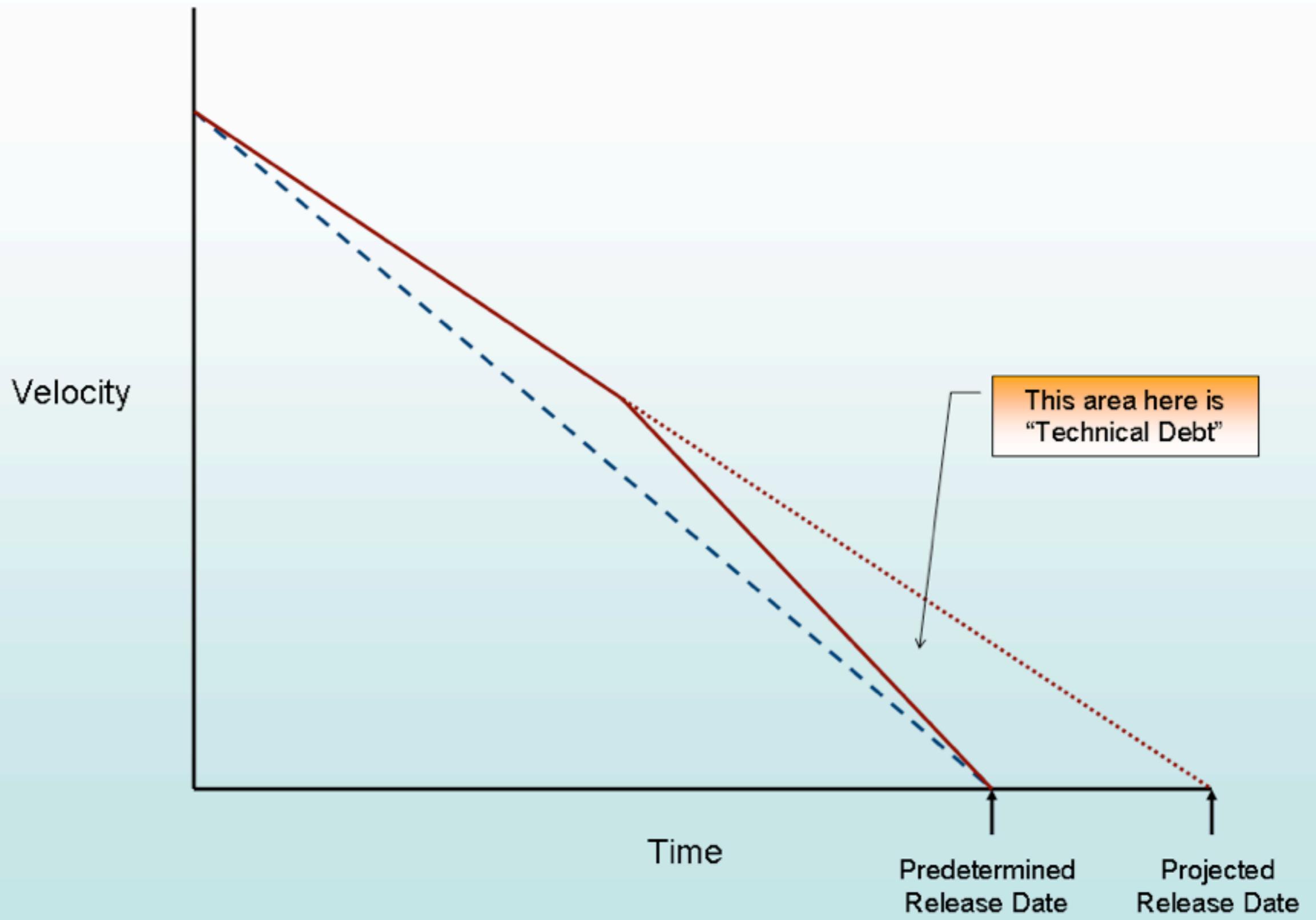
Name	Packages	Files	Classes	Methods	Lines	Conditionals
Cobertura Coverage Report	89% (8/9)	92% (23/25)	78% (57/73)	86% (139/162)	93% (864/930)	63% (1035/1639)

### Coverage Breakdown by Package

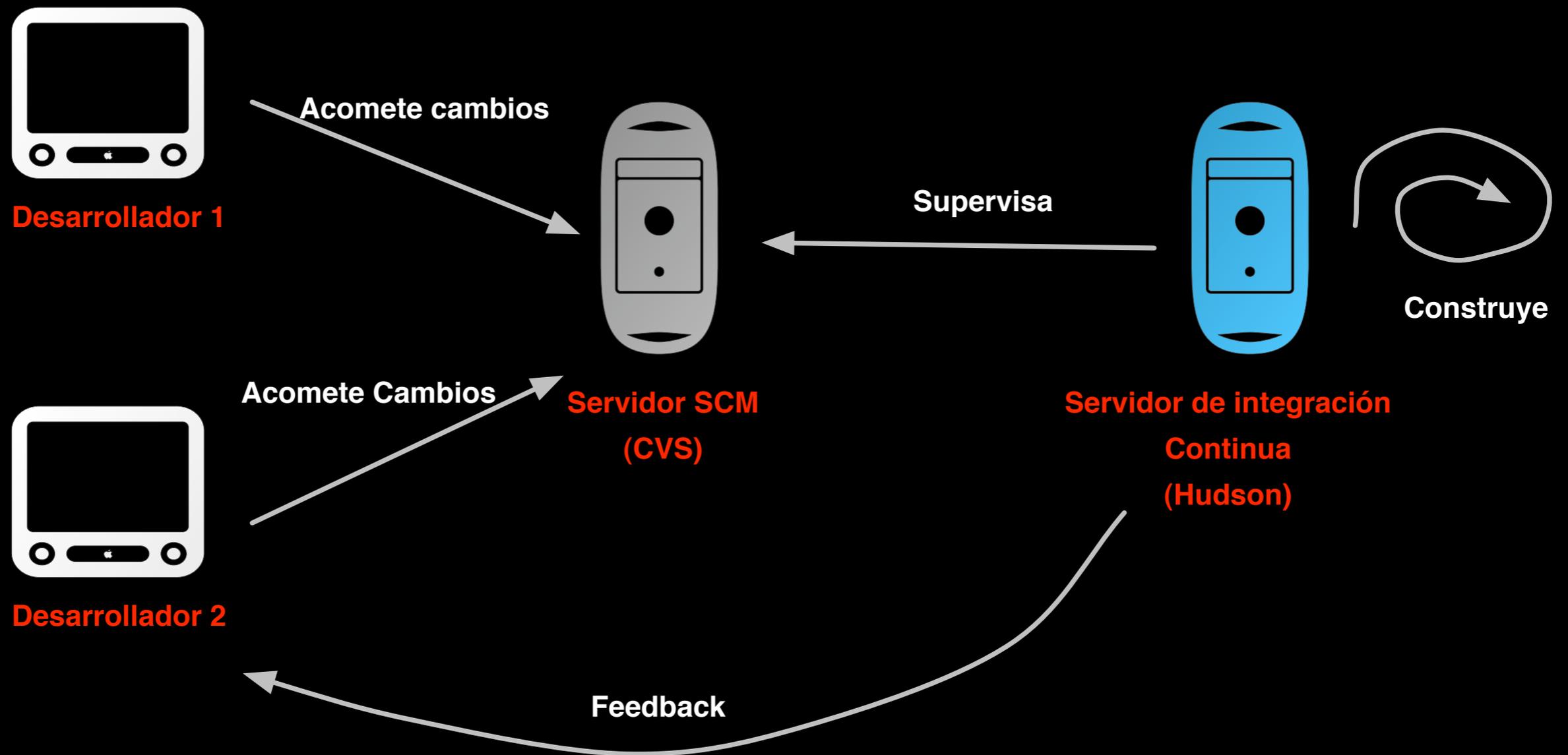
Name	Files	Classes	Methods	Lines	Conditionals
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.emf.util.mcd</a>	100% (7/7)	100% (7/7)	100% (22/22)	97% (63/65)	75% (41/55)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm</a>	0% (0/1)	0% (0/1)	0% (0/2)	0% (0/8)	100% (0/0)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.transformation.read.basic</a>	100% (1/1)	100% (16/16)	100% (16/16)	89% (49/55)	54% (62/114)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.transformation</a>	100% (1/1)	100% (20/20)	100% (62/62)	98% (618/630)	64% (846/1326)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.emf.util</a>	N/A	N/A	N/A	100% (0/0)	100% (0/0)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.emf.util.ldm.read</a>	100% (5/5)	100% (5/5)	94% (16/17)	94% (51/54)	85% (35/41)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.rochade.provider</a>	100% (1/1)	100% (1/1)	100% (4/4)	100% (12/12)	50% (2/4)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.utils</a>	50% (1/2)	6% (1/16)	10% (2/20)	18% (6/33)	7% (3/42)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.infrastructure</a>	100% (2/2)	100% (2/2)	100% (5/5)	100% (25/25)	100% (24/24)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.emf.util.ldm</a>	100% (5/5)	100% (5/5)	86% (12/14)	83% (40/48)	67% (22/33)

<a href="#">es.qiss.diccionarios.mcd.bus.ldm.emf.util.ldm</a>	100% (2/2)	100% (2/2)	88% (15/17)	83% (40/48)	61% (25/33)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.infrastructure</a>	100% (5/5)	100% (5/5)	100% (2/2)	100% (52/52)	100% (54/54)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.emf.util</a>	20% (1/5)	0% (1/17)	10% (5/50)	18% (6/33)	1% (3/54)
<a href="#">es.qiss.diccionarios.mcd.bus.ldm.rochade.provider</a>	100% (1/1)	100% (1/1)	100% (4/4)	100% (12/12)	50% (2/4)

Refactoring = mejora  
continua y temprana  
del diseño del código



# Integración continua



**Construcción** = Construcción + Tests +  
Despliegue

## Build Executor Status

No.	Status
1	Building <a href="#">es.qiss.diccionarios.busemfrochade.core #200</a> 

 [add description](#)

All +

S	W	Job ↓	Last Success	Last Failure	Last Duration	
		<a href="#">es.qiss.diccionarios.apirochade.groovy</a>	1 day 1 hr ( <a href="#">#102</a> )	1 day 22 hr ( <a href="#">#100</a> )	1 min 31 sec	
		<a href="#">es.qiss.diccionarios.busemfrochade.common</a>	1 hr 2 min ( <a href="#">#119</a> )	N/A	28 sec	
		<a href="#">es.qiss.diccionarios.busemfrochade.core</a>	2 hr 0 min ( <a href="#">#188</a> )	N/A	5 min 58 sec	
		<a href="#">es.qiss.diccionarios.feature</a>	1 day 22 hr ( <a href="#">#82</a> )	N/A	3,3 sec	
		<a href="#">es.qiss.diccionarios.mcd.bus.dbm.oracle</a>	1 hr 2 min ( <a href="#">#54</a> )	N/A	2 min 54 sec	
		<a href="#">es.qiss.diccionarios.mcd.bus.excel.conceptual</a>	59 min ( <a href="#">#122</a> )	N/A	17 min	
		<a href="#">es.qiss.diccionarios.mcd.bus.excel.mapping</a>	1 day 22 hr ( <a href="#">#108</a> )	16 min ( <a href="#">#123</a> )	2 min 14 sec	
		<a href="#">es.qiss.diccionarios.mcd.bus.ldm</a>	42 min ( <a href="#">#149</a> )	N/A	8 min 51 sec	
		<a href="#">es.qiss.diccionarios.mcd.bus.msl</a>	6 days 9 hr ( <a href="#">#119</a> )	22 min ( <a href="#">#152</a> )	18 min	
		<a href="#">es.qiss.diccionarios.mcd.capaconceptual.ui</a>	13 min ( <a href="#">#87</a> )	N/A	1 min 9 sec	
		<a href="#">es.qiss.diccionarios.mcd.transformation.adabasdbm2ldm</a>	33 min ( <a href="#">#291</a> )	23 hr ( <a href="#">#284</a> )	11 min	
		<a href="#">es.qiss.diccionarios.mcd.transformation.ldm2msl</a>	1 day 22 hr ( <a href="#">#152</a> )	13 min ( <a href="#">#169</a> )	23 sec	
		<a href="#">es.qiss.diccionarios.mcd.transformation.ldm2namedldm</a>	59 min ( <a href="#">#70</a> )	2 hr 41 min ( <a href="#">#68</a> )	20 sec	
		<a href="#">es.qiss.diccionarios.metamodels.mcd</a>	13 hr ( <a href="#">#89</a> )	N/A	12 sec	
		<a href="#">es.qiss.diccionarios.metamodels.rochade</a>	2 hr 0 min ( <a href="#">#94</a> )	13 hr ( <a href="#">#93</a> )	7,1 sec	
		<a href="#">es.qiss.diccionarios.update-site</a>	1 day 22 hr ( <a href="#">#99</a> )	1 day 22 hr ( <a href="#">#97</a> )	2 min 4 sec	

Icon: [S](#) [M](#) [L](#)

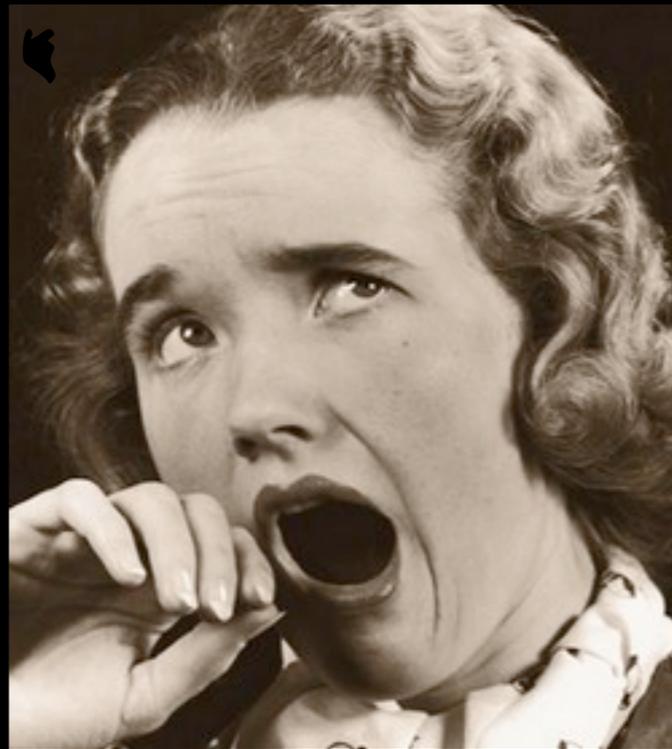
Legend  for all  for failures  for just latest builds

Icon: [S](#) [M](#) [L](#)

Legend  for all  for failures  for just latest builds



Y para el cuarto...



Diferenciarse



No aburrir



Sin pasarse

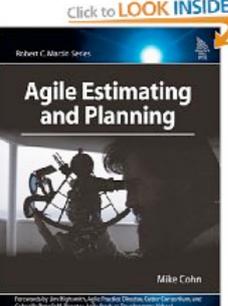
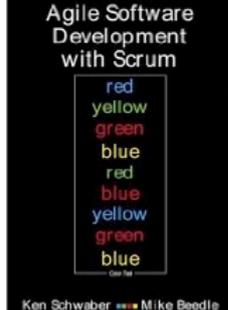
# Proceso de Desarrollo

Elemento	Ejemplos
Descripción del proceso	<ul style="list-style-type: none"><li>• Iteraciones cortas, completas</li><li>• Niveles de planificación: de lanzamiento, de iteración, diario</li><li>• Planificación priorizada</li><li>• Historias de usuario, <i>story points</i>, velocidad</li></ul>
Configuración del equipo de desarrollo	<ul style="list-style-type: none"><li>• Implicación del cliente</li></ul>

# Aspectos técnicos

<b>Elemento</b>	<b>Ejemplos</b>
Uso de un servidor de integración continua	<ul style="list-style-type: none"><li>•Detección temprana de errores</li><li>•Software siempre listo para desplegar</li></ul>
Pruebas automáticas	<ul style="list-style-type: none"><li>•Exhaustividad</li><li>•Automatización de tests de aceptación</li></ul>
Uso de herramienta de cobertura de código	<ul style="list-style-type: none"><li>•Definición de umbrales de cobertura a exigir: ejemplo, 90% código cubierto</li></ul>
Uso de herramientas de análisis de código	<ul style="list-style-type: none"><li>•Detección de código sin documentar</li><li>•Nombres de identificadores con tamaño suficiente</li><li>•Detección de métodos demasiado largos</li></ul>

# Referencias

	<p><i>Agile Estimating and Planning.</i> Mike Cohn. Prentice Hall, 2005</p>
	<p><i>Agile Software Development with Scrum.</i> Ken Schwaber. Prentice Hall, 2001</p>
	<p><i>The art of Agile Development.</i> James Shore and Shane Warden. Prentice Hall, 2007</p>
	<p><i>User Stories Applied: For Agile Software Development.</i> Mike Cohn. Addison-Wesley Professional, 2004</p>