

## **4.4 Metodologías ágiles de desarrollo de SW SCRUM**

# Indice

1. ¿Qué es SCRUM?
2. Características
3. Tipos de participantes
4. Roles en SCRUM
5. Reuniones
6. Artefactos y herramientas
7. Valores
8. Errores comunes y malas interpretaciones
9. Resumen
10. Bibliografía



# Scrum

“El enfoque de ‘carrera de relevos’ en el desarrollo de productos ... puede entrar en conflicto con los objetivos de máxima velocidad y flexibilidad. En su lugar, un enfoque estilo ‘rugby’ - donde un equipo intenta ir a la distancia como una unidad, pasando la pelota hacia adelante y hacia atrás -pueden servir mejor a los actuales requisitos competitivos”.

*Hiroataka Takeuchi and Ikujiro Nonaka, “The New New Product Development Game”, Harvard Business Review, January 1986.*

# ¿Qué es Scrum?

Metodología desarrollada por:

Ken Schwaber, Jeff Sutherland y Mike Beedle

- ¿De dónde viene el término Scrum?
  - El término Scrum viene de una jugada de rugby donde los jugadores se apiñan para conseguir el balón (melé)
  - Todo el equipo trabaja para conseguir el mismo objetivo



# Características

- Metodología iterativa e incremental
- Equipo auto-dirigido y auto-organizado
- Prohibido añadir trabajo extra a una iteración
- Reuniones diarias
- Iteraciones duración máxima 30 días
- Incrementos funcionales en cada iteración
- Demo al cliente al final de cada iteración
- Iteraciones conducidas por los clientes

# Desarrollo secuencial vs superpuesto

Requisitos

Diseño

Código

Test

En lugar de hacer sólo una cosa a la vez ...

...los equipos Scrum hacen un poco de todo todo el tiempo

# Scrum Framework

## Roles

- Product Owner
- Scrum Master
- Equipo
- Otros

## Reuniones

- La planificación del Sprint
- El Scrum diario
- La revisión del Sprint
- La retrospectiva del Sprint

## Artefactos y herramientas

- Backlog de Producto
  - Gráfica Burn-up
- Backlog de Sprint
  - Gráfica Burn-down

# Participantes

- Hay una historia que describe gráficamente los 2 tipos principales de participantes en un proyecto





# Participantes

- Una **gallina** y un **cerdo** paseaban por la carretera . La gallina le dijo al cerdo: “¿Quieres abrir un **restaurante** conmigo?”. El cerdo consideró la propuesta y respondió: “Sí, me gustaría. ¿Y cómo lo llamaríamos?” La gallina respondió: “**Huevos con jamón**”
- El cerdo se detuvo, hizo una pausa y contestó: “Pensándolo mejor, creo que no voy a abrir un restaurante contigo. Yo estaría realmente **comprometido**, mientras que tú estarías sólo **implicada**.”

# Participantes

- Scrum diferencia claramente estos dos grupos para garantizar que **quienes tienen la responsabilidad, también tienen la autoridad** necesaria para poder lograr el éxito y que quienes no tienen la responsabilidad no producen interferencias innecesarias



# Roles en SCRUM

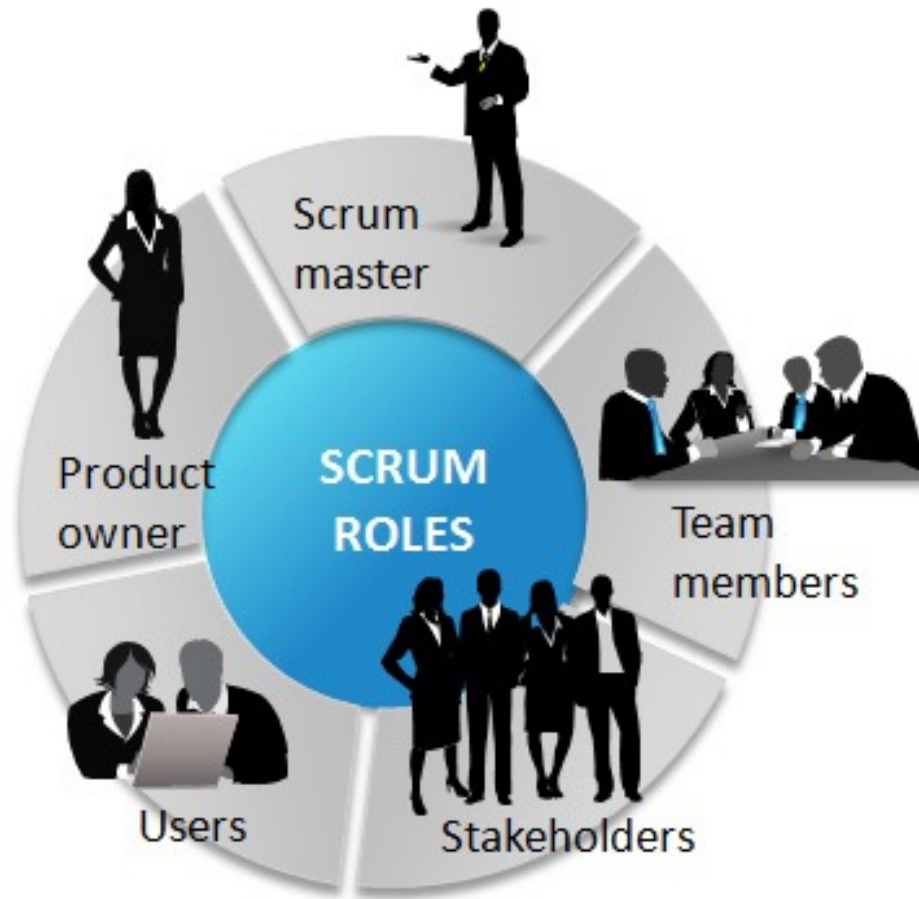
## Roles (COMPROMETIDOS):

- **Product Owner:** responsable de que el desarrollo iterativo del producto cumpla las necesidades del cliente
- **Equipo de desarrollo:** el equipo encargado de desarrollar el producto
- **Scrum Master:** encargado de que el proceso Scrum se realiza correctamente

## Roles (IMPLICADOS):

- Usuarios de la aplicación
- Clientes
- Managers
- Marketing

# Roles de Scrum



# Dueño del producto (Product Owner)

- Define las funcionalidades del producto
- Decide sobre las fechas y contenidos de los releases
- Es responsable por la rentabilidad del producto (ROI)
- Prioriza funcionalidades de acuerdo al valor del mercado/negocio
- Ajusta funcionalidades y prioridades en cada iteración si es necesario
- Acepta o rechaza los resultados del trabajo del equipo

# Scrum Master

- Gestiona y supervisa el proyecto
- Responsable de promover los valores y prácticas de Scrum
- Elimina impedimentos
- Se asegura de que el equipo es completamente funcional y productivo
- Permite la estrecha cooperación en todos los roles y funciones
- Escudo del equipo ante interferencias externas

# Equipo de desarrollo

- Típicamente de 5 a 9 personas
- Multi-funcional:
  - Programadores, testers, analistas, diseñadores, etc
- Los miembros deben ser full-time
- Los equipos se auto-organizan
- Sólo puede haber cambio de miembros entre los Sprints

# Reuniones

- Scrum establece unas cuantas reuniones obligatorias
- La principal característica es que tienen un límite de tiempo
  - Obligan a los involucrados a ser eficientes y hacer un uso inteligente del tiempo
- Tipos de reuniones:
  - La planificación del Sprint (SM, PO, Equipo y otros(opc))
  - El Scrum diario (SM, Equipo, PO(opc) y otros (opc))
  - La revisión del Sprint (SM, Equipo, PO y otros)
  - La retrospectiva del Sprint (SM, Equipo)



# Planificación del Sprint

- Scrum es un proceso cíclico de Sprints: una iteración de trabajo
- El resultado de cada Sprint debe ser una pieza de software funcionando
- Se planifica únicamente el esfuerzo que se puede realizar durante la duración de un Sprint
- Objetivos:
  - Definir el objetivo del Sprint
  - Hacer una estimación del tiempo de los requerimientos de más alta prioridad y seleccionar cuántos se pueden lograr terminar en el Sprint

# Planificación del Sprint

- Resultados:
  - La meta del Sprint
  - El backlog del Sprint
- El tiempo de la planificación debe ser suficiente para poder hacer el análisis y las estimaciones de los requerimientos
- El dueño del producto puede repriorizar el backlog del producto

# Planificación del Sprint

- Consta de dos partes:
  - **Primera parte:** Duración de 2 a 4 horas. Se seleccionan los elementos de la pila de producto que se van a desarrollar
  - **Segunda parte:** Se desglosan para determinar las tareas necesarias, estimar el esfuerzo de cada una y asignarlas a las personas del equipo

# Planificación del Sprint

- Un día completo de trabajo permite obtener una buena planificación separada en dos partes:
  - Análisis, evaluación, repriorización y estimación del backlog del producto
  - Desglose y estimación de las tareas de los requerimientos aceptados por el equipo.
- No se recomienda extender esta planificación más allá de un día laboral completo

# Planificación del Sprint

- Precondiciones:
  - La organización tiene determinados los **recursos disponibles** para llevar a cabo el Sprint
  - El **PO** tiene preparada la **pila de producto**, con su criterio de **prioridad** para el negocio y un nº suficiente de elementos para desarrollar en el Sprint
  - Siempre que sea posible el **PO** debe haber **trabajado antes con el equipo**. De esta forma, su estimación previa del trabajo que se puede realizar en el Sprint será bastante ajustada
  - El **equipo** tiene un **conocimiento previo de las tecnologías** empleadas **y del negocio del producto** suficiente para realizar estimaciones basadas en “juicios de expertos”

# Planificación del Sprint

- Entradas:
  - Pila del producto
  - El producto desarrollado hasta la fecha a través de los sucesivos incrementos (excepto si se trata del primer Sprint)
  - Circunstancias de las condiciones de negocio del cliente y del escenario tecnológico empleado (información del entorno)

# Planificación del Sprint

- Resultados:
  - Pila del Sprint
  - Duración del Sprint y fecha de la reunión de revisión
  - Objetivo del Sprint
- La reunión es conducida por el responsable del funcionamiento de Scrum (**Scrum Master**) a la que deben asistir el **Product Owner**, todo el **Equipo** y opcionalmente **otros implicados** en el proyecto

# Planificación del Sprint

- Primera parte (de 1 a 4 horas):

## Product Owner:

- Presenta las funcionalidades de la pila de producto que tienen mayor prioridad y que estima se pueden realizar en el Sprint
- Nivel de detalle suficiente para transmitir al equipo toda la información necesaria

## Equipo

- Realiza las preguntas y solicita las aclaraciones necesarias
- Propone sugerencias, modificaciones y soluciones alternativas



# Planificación del Sprint

- Segunda parte (hasta el final de la jornada):
  - El equipo desglosa cada funcionalidad en tareas
  - Estima el tiempo para cada tarea
  - Determina las tareas de la pila del Sprint
  - Se tienen en cuenta los elementos de diseño y arquitectura que se deberán incorporar
  - Los miembros del equipo se auto-asignan las tareas
  - El Product Owner atiende dudas y comprueba que el equipo comprende el objetivo
  - El Scrum Master actúa de moderador de la reunión
    - Evitar que el equipo comience a profundizar en trabajos de análisis o arquitectura propios del Sprint

# Planificación del Sprint



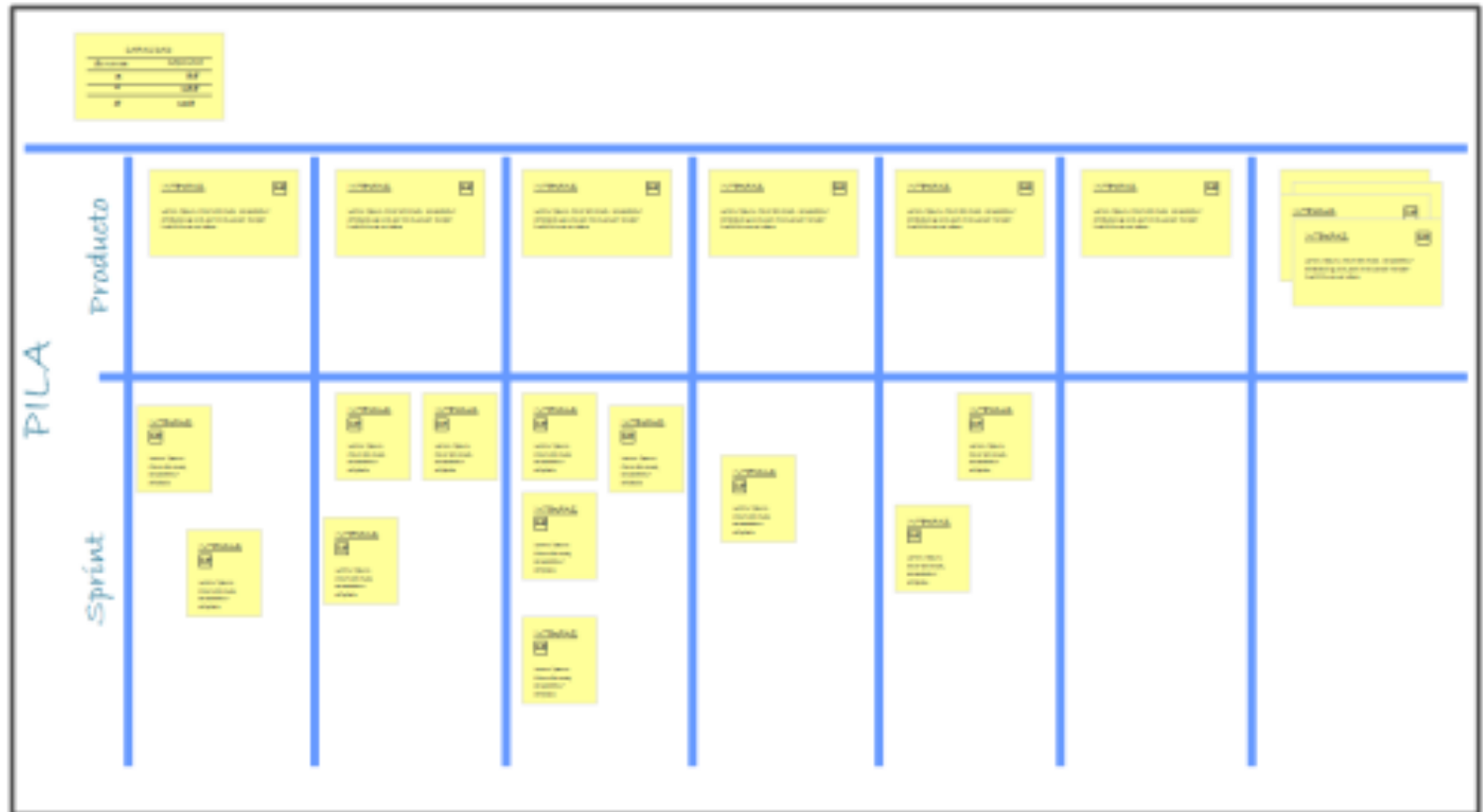
**Fuente: Scrum Manager. Gestión de proyectos.**

# Tablero de trabajo

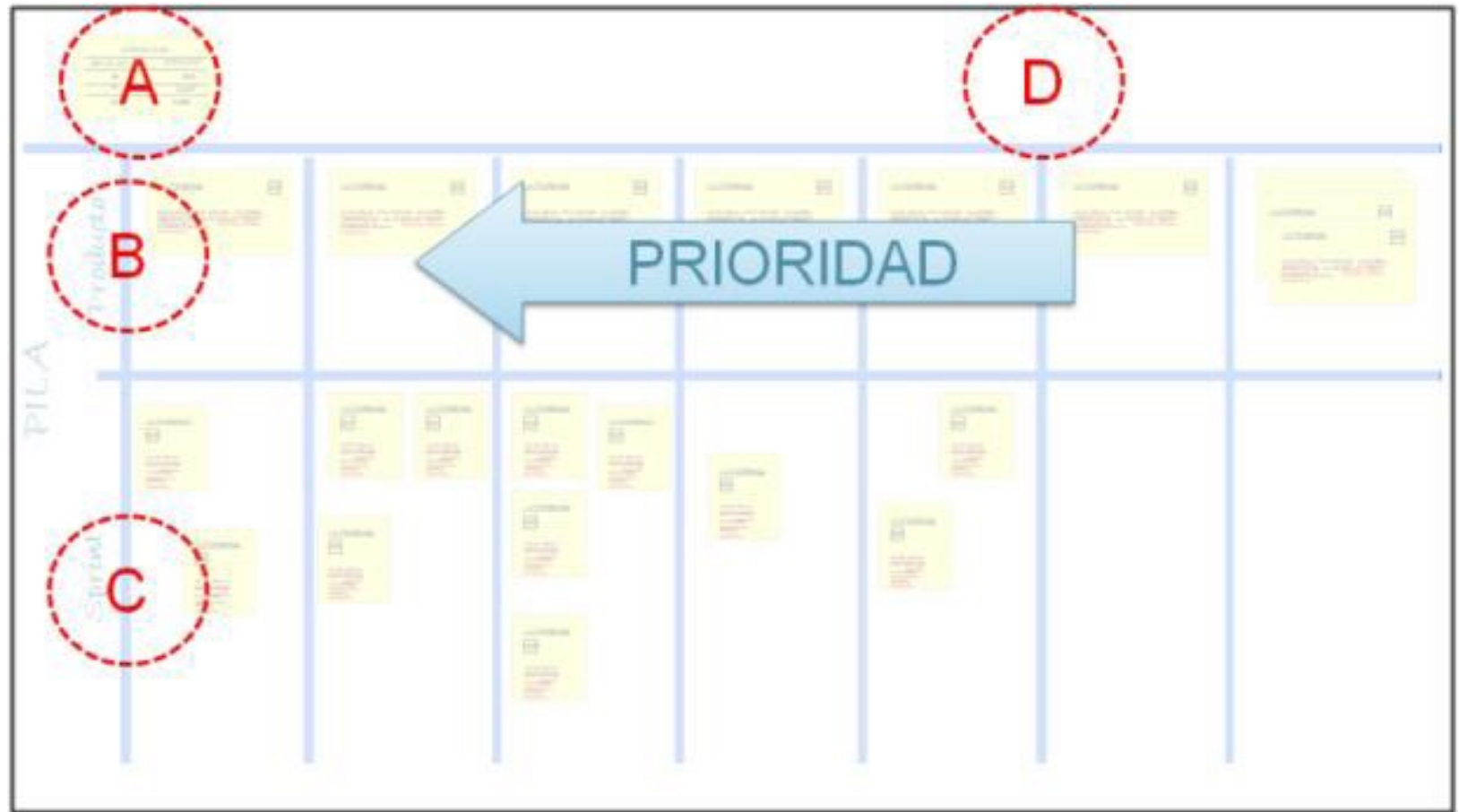
- El Product Owner debería emplear una hoja de cálculo o alguna herramienta para guardar en formato digital la pila del producto
- Pero no se aconseja utilizarla para trabajar con ella en la reunión
- Es mucho mejor y más eficiente trabajar con objetos físicos. Usar una pizarra y objetos movibles (adhesivos, chinchetas, imanes)

# Tablero de trabajo

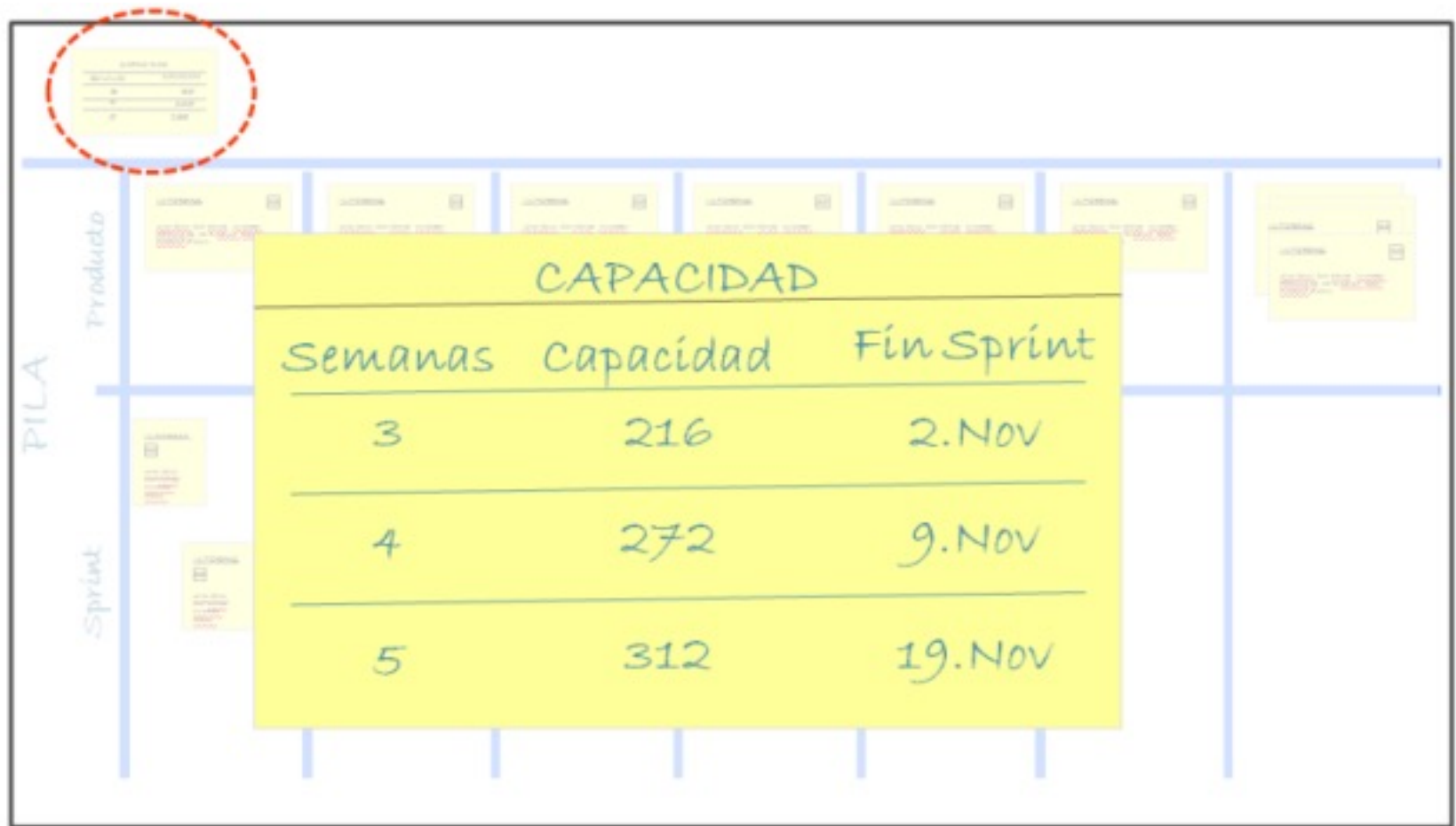
- Ejemplo de tablero de trabajo:



# Tablero de trabajo



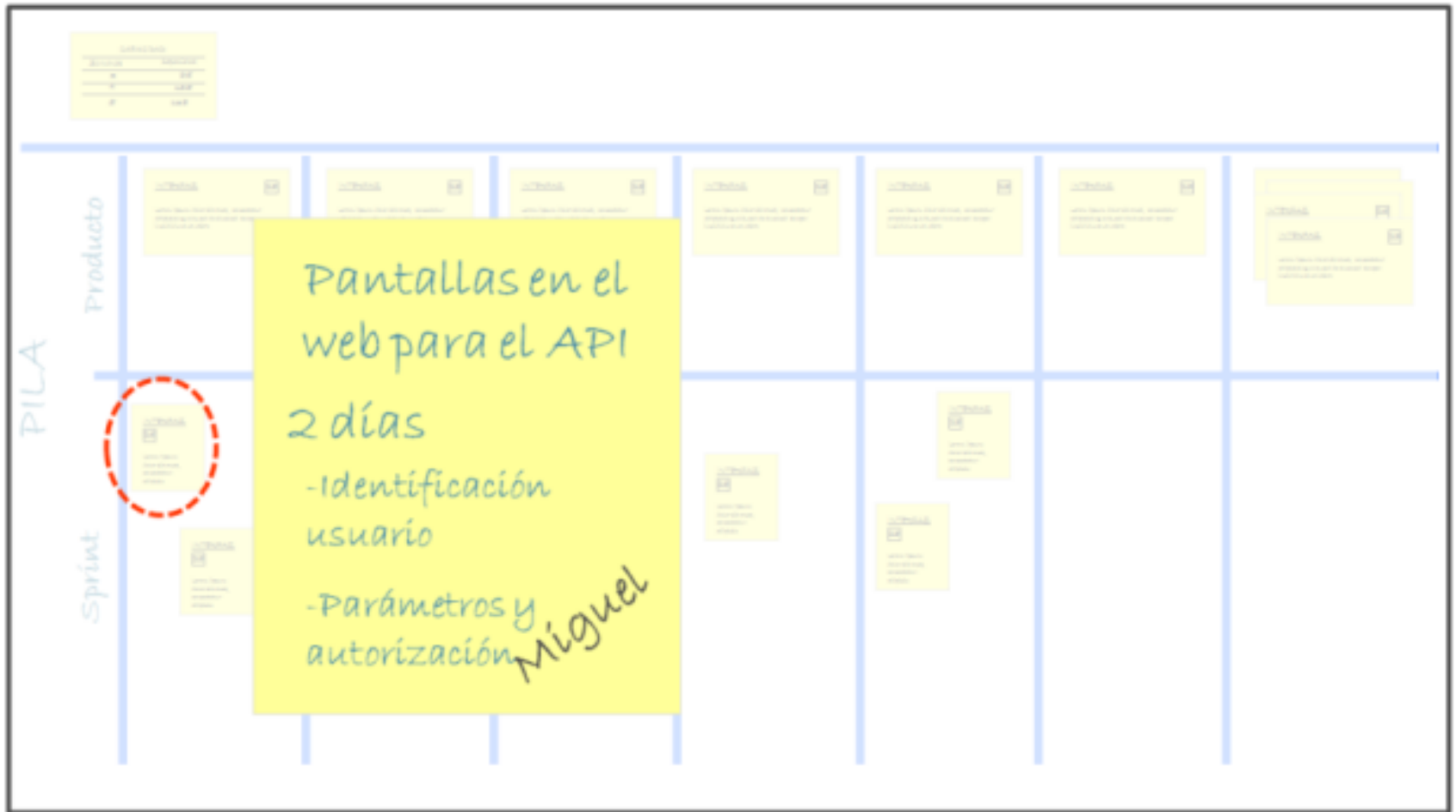
# Tablero de trabajo



# Tablero de trabajo

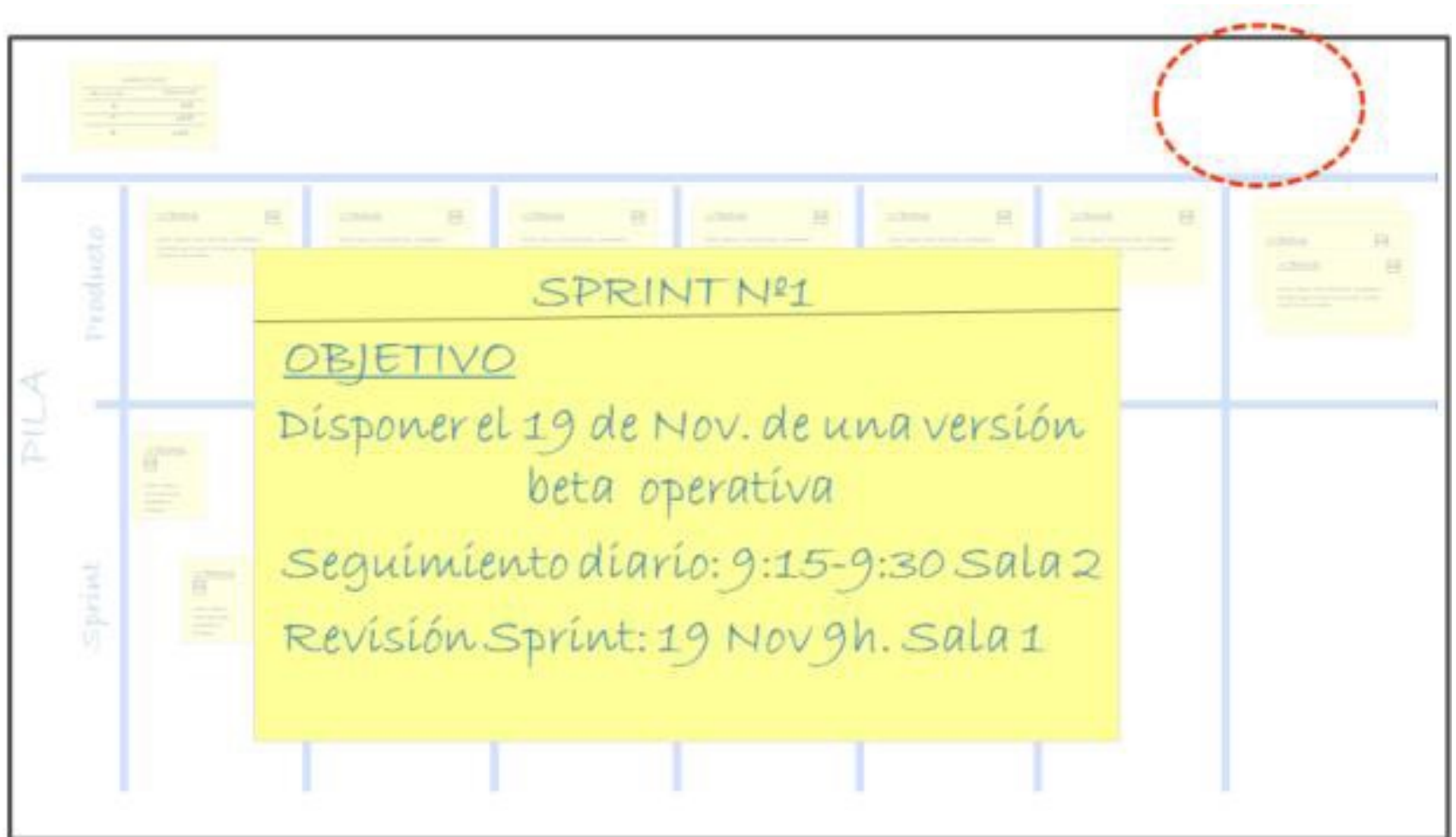


# Tablero de trabajo





# Tablero de trabajo



# El juego de la estimación

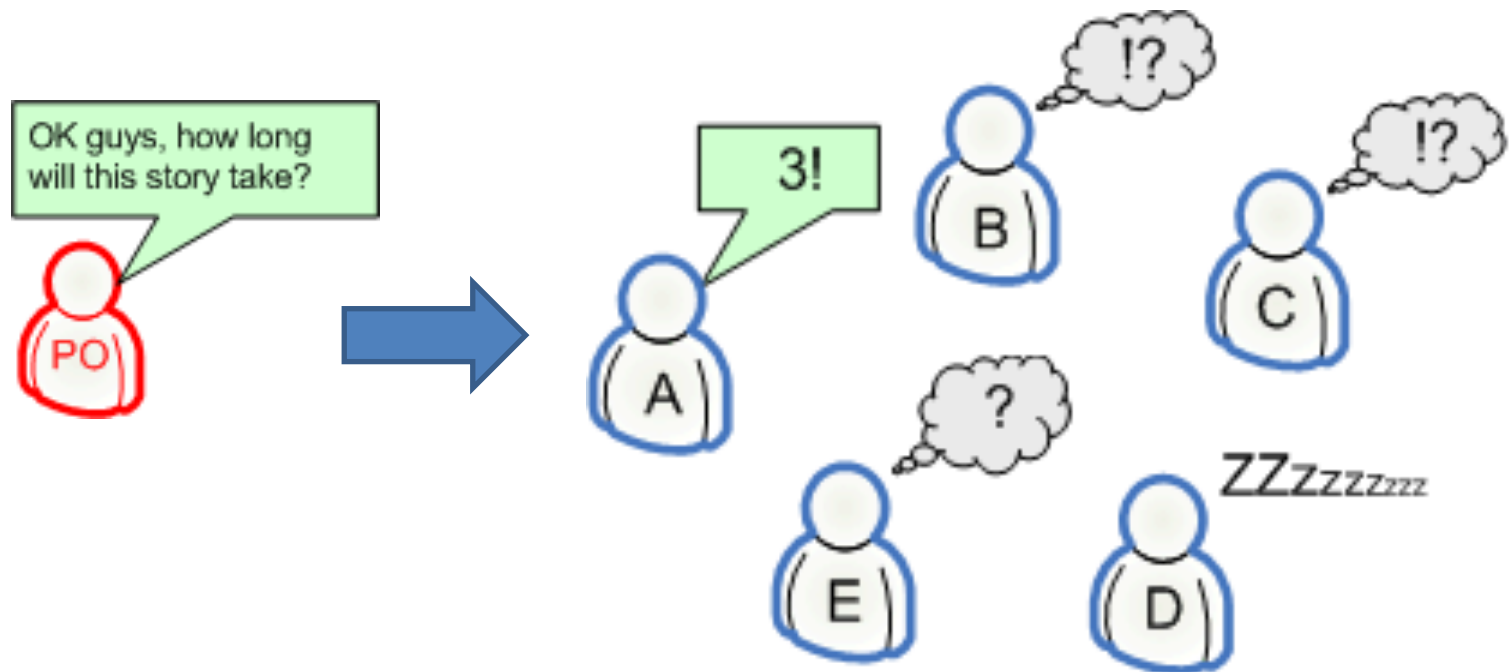
- La estimación es una de las actividades principales de las metodologías ágiles
- La **estimación** es el proceso de **asignar el tamaño a una historia**, es decir, cuánto tiempo llevará, cuánto trabajo costará, etc
- En Scrum la estimación es una actividad que se realiza en equipo
- Para cada historia **todo el equipo participa** en el proceso de estimación

# El juego de la estimación

- El juego de **estimación Planning Poker** (Scrum Poker) es muy simple y permite que la estimación sea más rápida, más precisa y más divertida
- El Planning Poker se utiliza para estimar el **esfuerzo** o el **tamaño relativo** de las **tareas** en el desarrollo de software de manera fiable
- Los **miembros del equipo** del proyecto se reúnen y en unas pocas rondas mediante el Planning Poker llegan a un consenso sobre el tamaño de cada tema o tarea

# El juego de la estimación

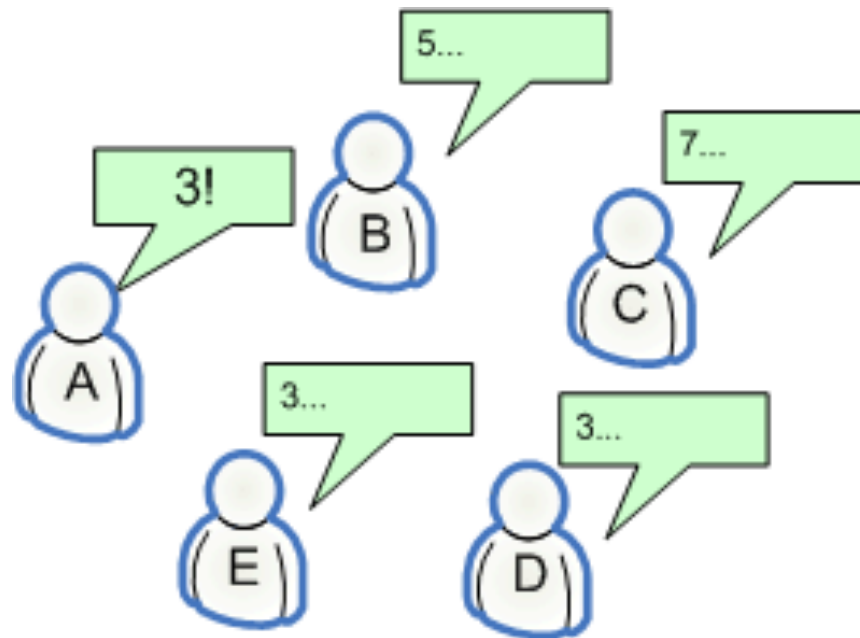
- Estimación sin Planning Poker:



Fuente: <http://www.crisp.se/planningpoker/>

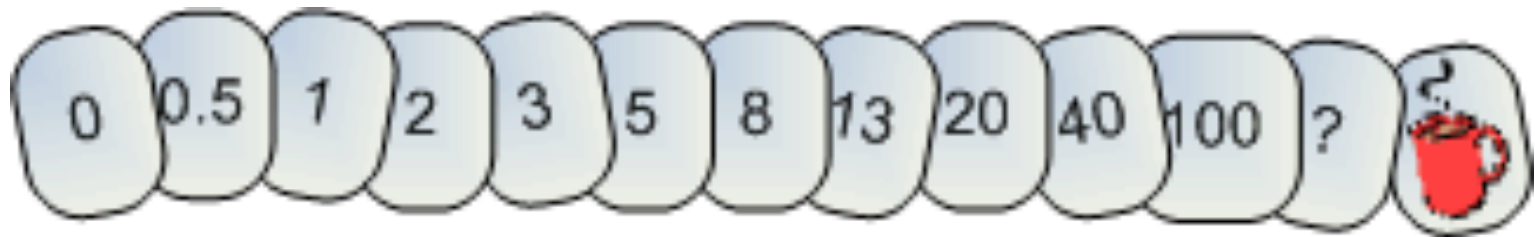
# El juego de la estimación

- Resultado:

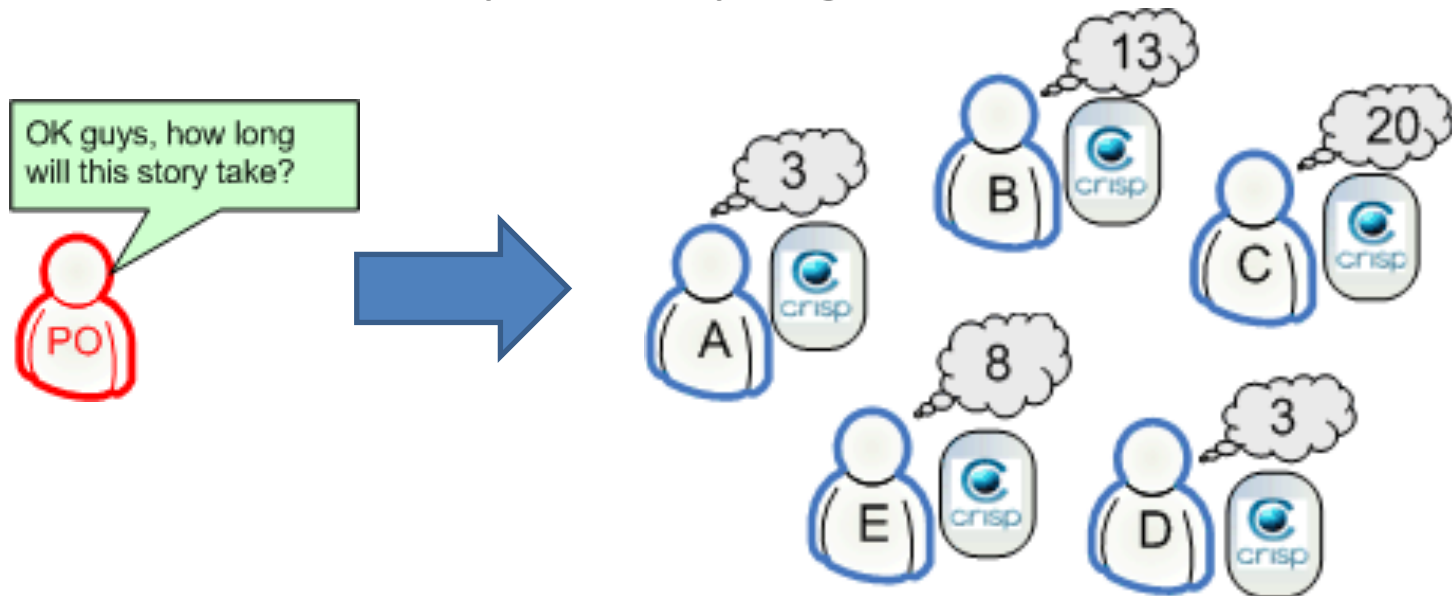


# El juego de la estimación

- Ahora imagina que cada miembro tiene un juego de cartas:

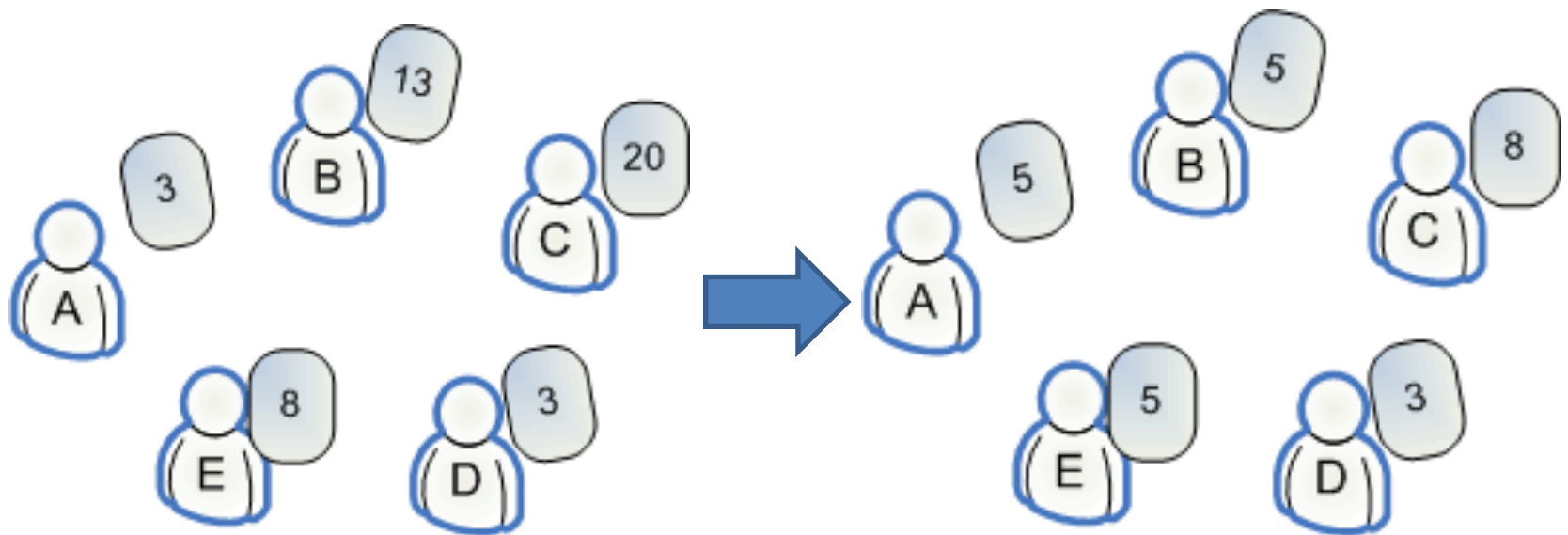


- Volvamos a repetir la pregunta:



# El juego de la estimación

- Resultado:



# Scrum diario

- Las bases teóricas de Scrum tienen su origen en el control empírico de procesos
  - Inspección frecuente y adaptabilidad
- Esta reunión se realiza con **todos los miembros del equipo** frente a una pizarra o pared de control del proyecto
- Esta reunión se recomienda que no exceda los **15 minutos**
- Se realiza **todos los días laborables a la misma hora y en el mismo lugar**



## To Do

Installer Update

Upgrade Production Server

Database Query Profiling

Improve Retention

Merge New Engine Build

Promotion Discounts

Optimize Loading Time

Test New Workflows

Upgrade Automatic Build System

Design Authentication System

Export Item Shop List Feature

Predictive Analytics

Improve Face Recognition

Integrate New Anti-Hacking System

Improve Server Response Times

Optimize Stats Queries

Switch to New Graphic Format

## In Progress

Server Load Balancing

Add VR Overlay

Revenue Dashboard

Localize to German

Item Shop Optimization

New UI Icons

Detailed Server Logs

New Login Dialog

Switch to new Database

## Testing

Migrate Analytics Module to Cloud

Review Digital Signatures

UI Redesign

Fix Automatic Build

Test New Performance Profiler

New Search Panel

Automatic Crash Logs Upload

## Done

Cloud System Configuration

Final UI Review

System Integration

# Scrum diario

- Todos los **miembros del equipo** y el **Scrum Master** están **obligados a asistir** a estas reuniones
- El **dueño del producto** también puede asistir pero **no está obligado** a ello
- También **cualquier interesado** en el proyecto **puede asistir** pero **no puede interrumpir** el trabajo del equipo
  - Los interesados en el proyecto tienen diversos mecanismos por los que pueden conocer el progreso del proyecto y agregar sus propios requerimientos al backlog del producto

# Scrum diario

- Cada miembro del equipo debe responder a 3 preguntas:
  - ¿Que hiciste/lograste ayer?
  - ¿Que vas a hacer/lograr hoy?
  - ¿Que impedimentos tienes para lograrlo?
- Las primeras 2 preguntas son tareas del backlog del Sprint
- El Scrum diario no debe convertirse en un mecanismo de control sino en un compromiso por parte de los miembros del equipo

# Scrum diario

- El **Scrum Master** tiene la responsabilidad de **registrar los impedimentos** del equipo **y eliminarlos**
- Si la lista de impedimentos no disminuye o se hace más grande, el Scrum Master no está haciendo bien su trabajo
  - En este caso excepcional, el equipo puede solicitar que se interrumpa el Sprint de forma extraordinaria
- Cada miembro del equipo debe conocer los requerimientos y auto-asignarse tareas comprometiéndose a terminarl

# Revisión del Sprint

- La revisión del Sprint se realiza **al final de cada Sprint**
- Todo el equipo está enfocado a lograr la meta que se definió en el Sprint
- Al final de los 30 días el equipo debe hacer una **demostración de la funcionalidad** del software que terminaron en el Sprint
- Esta demostración se hace **en los ordenadores del equipo de desarrollo**

# Revisión del Sprint

- Asistentes:
  - El dueño del producto (Product Owner)
  - El Scrum Master
  - Todo el equipo
  - Cualquier interesado en el proyecto
- La demostración debe realizarse entre 4 y 8 horas como máximo
- El equipo de desarrollo realiza las **pruebas de funcionalidad para cada uno de los requerimientos** aceptados para el Sprint

# Revisión del Sprint

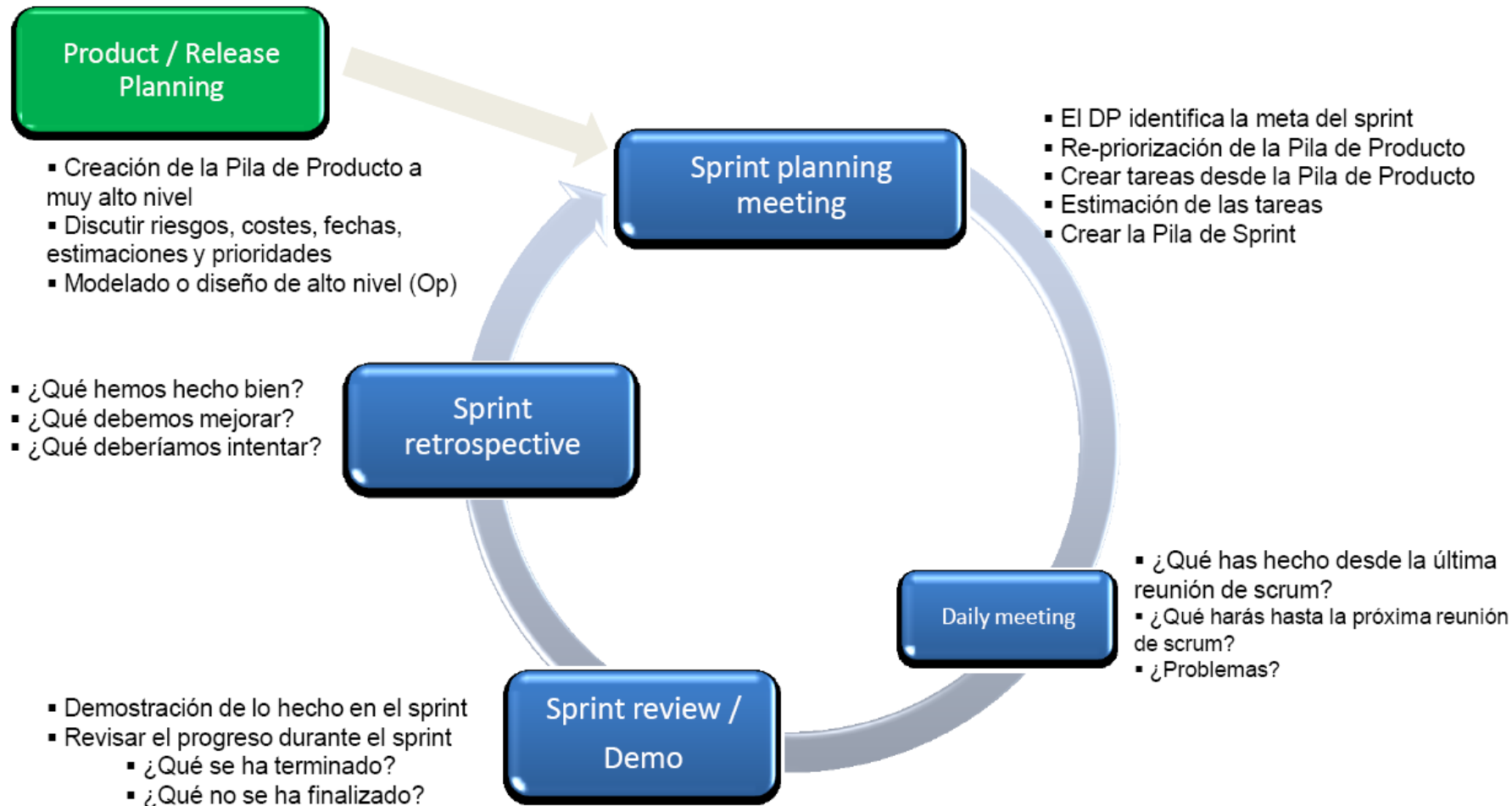
- Los asistentes hacen preguntas y discuten el resultado del Sprint
- Si se descubren **nuevos requerimientos** éstos **se añaden al backlog del producto** para ser priorizados por el dueño del producto antes de la siguiente reunión de planificación
- El dueño del producto debe aprovechar esta reunión para conocer las opiniones del resto de los interesados, integrar nuevos requerimientos y adaptar el proyecto a nuevas condiciones (si las hubiera) del entorno de negocio

# Retrospectiva del Sprint

- La retroalimentación, el aprendizaje y la experiencia son fundamentales
- **Después de la revisión del Sprint** el equipo se reúne para **evaluar los resultados durante el Sprint y cómo se puede mejorar**
- Cada miembro del equipo contesta a las siguientes preguntas:
  - Qué es lo que **SÍ funciona** para seguir haciéndolo...
  - Qué es lo que **NO funciona** para dejar de hacerlo...
  - Qué podemos empezar a hacer para que **funcione MEJOR**



# Reuniones



# Artefactos y herramientas



# Artefactos y herramientas

- Los procesos de desarrollo de software incluyen la creación y mantenimiento de diversos “artefactos” documentales que pretenden dar tangibilidad a un proceso de producción intangible
- Scrum obliga a **3 tipos de documentos** usados tanto para gestionar el proyecto y sus tareas como para dar seguimiento, visibilidad y transparencia al proceso:
  - El **Product Backlog** (pila del producto) o priorización de requerimientos
  - El **Sprint Backlog** (pila del Sprint) o priorización de tareas
  - La **gráficas de burn-up y burn-down**

# Product Backlog

- El **análisis y definición de requerimientos** es una tarea muy compleja
- Frecuentemente, el cliente no sabe lo que quiere
- Cuando el cliente ve el producto visualiza la solución de forma distinta
- Establecer unos requerimientos sin posibilidad de readaptarlos no es una buena idea
- Hay que estar preparados para **adaptarse a los cambios** que puedan surgir

# Product Backlog

- **Requisitos del cliente**
- Inventario de funcionalidades, mejoras, tecnología y corrección de errores que deben incorporarse al producto
- Todo lo que suponga un trabajo que debe realizar el equipo debe estar reflejado en esta pila
- Posibles entradas de una pila de producto:
  - Permitir a los usuarios la consulta de las obras publicadas por un determinado autor.
  - Permitir la consulta de una obra a través de un API web

# Product Backlog

- Listado de requerimientos expresados en forma de **historias de usuario** (User Stories) y **estimados en unidades de tiempo**
- Las historias de usuario deben expresarse en el **lenguaje del negocio** y deben aportar un valor real al mismo
- Este listado es **priorizado por el dueño del producto** y **estimado por el equipo** en cada reunión de planificación

# Product Backlog

- Debe incluir el mayor número de requerimientos que se puedan obtener de los usuarios y demás interesados del proyecto a través del dueño del producto
- En base a las estimaciones del equipo, **sirve para determinar el número de Sprints** necesarios para terminar el proyecto
- De este backlog **el equipo** selecciona los requerimientos de más alto nivel que puede completar en 30 días y **produce** el siguiente artefacto documental: **el backlog del Sprint**

# Product Backlog

- Se deben especificar los **criterios de aceptación** para cada ítem del Backlog del producto para considerar cumplido el requisito
- Los ítems no necesariamente deben tener la misma complejidad
- Los **menos complejos** se denominan **historias de usuario**
- Los **más complejos** se denominan **Temas o Epics**



# Product Backlog

- **Historia de usuario**

- Como un: Doctor a cargo de una emergencia
- Quiero: poder ver un mapa con los lugares de derivación candidatos
- De tal forma que: pueda saber a dónde llevar los afectados por la emergencia rápidamente, minimizando el tiempo de traslado de los accidentados

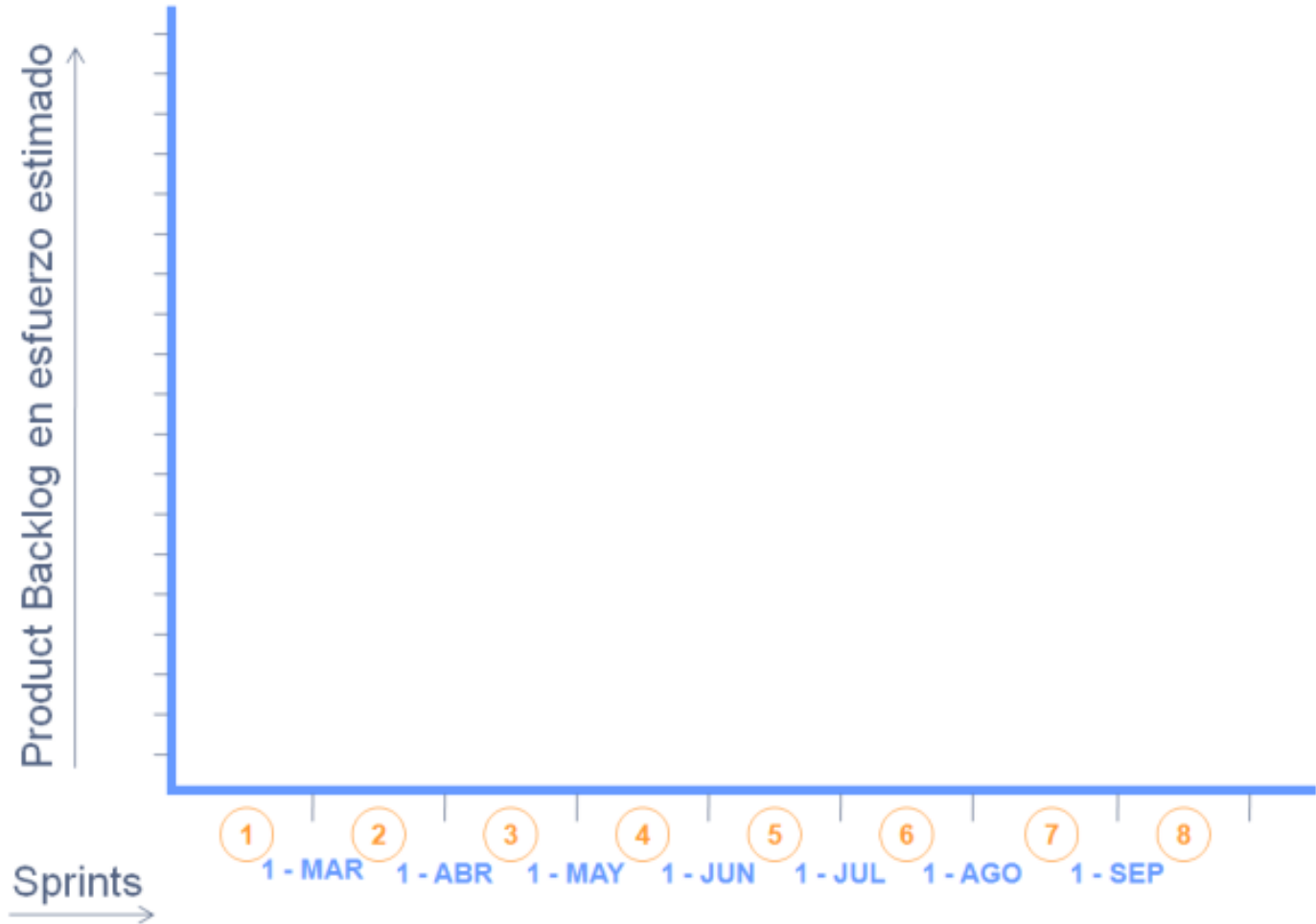
- **Casos de Aceptación**

- Existe una opción de “ver mapa”
- El mapa muestra la posición actual
- El mapa muestra los centros de derivación con una indicación de su tipo:
  - Hospitales municipales: rojo
  - Clínicas azul

# Gráfico burn-up

- **Herramienta de planificación y seguimiento del Product Owner**
- Muestra el plan general de desarrollo del producto y la traza de su evolución
- Se confecciona con:
  - La **estimación del esfuerzo** prevista en la pila del producto.
  - La **velocidad del equipo**
- Gráfico que representa:
  - **X: Fechas**
  - **Y: Estimación de esfuerzo** para desarrollar el producto

# Gráfico burn-up

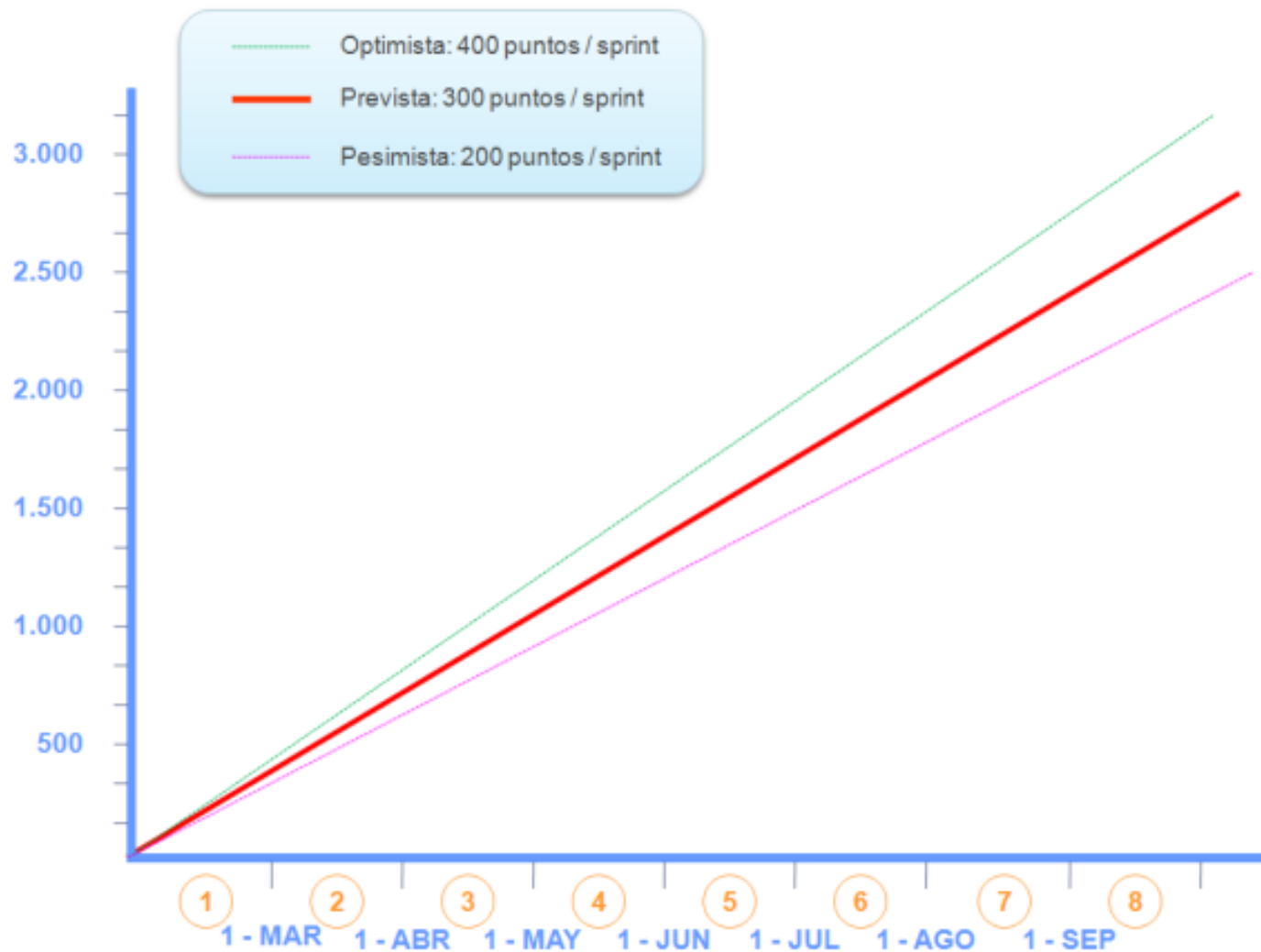


# Gráfico burn-up

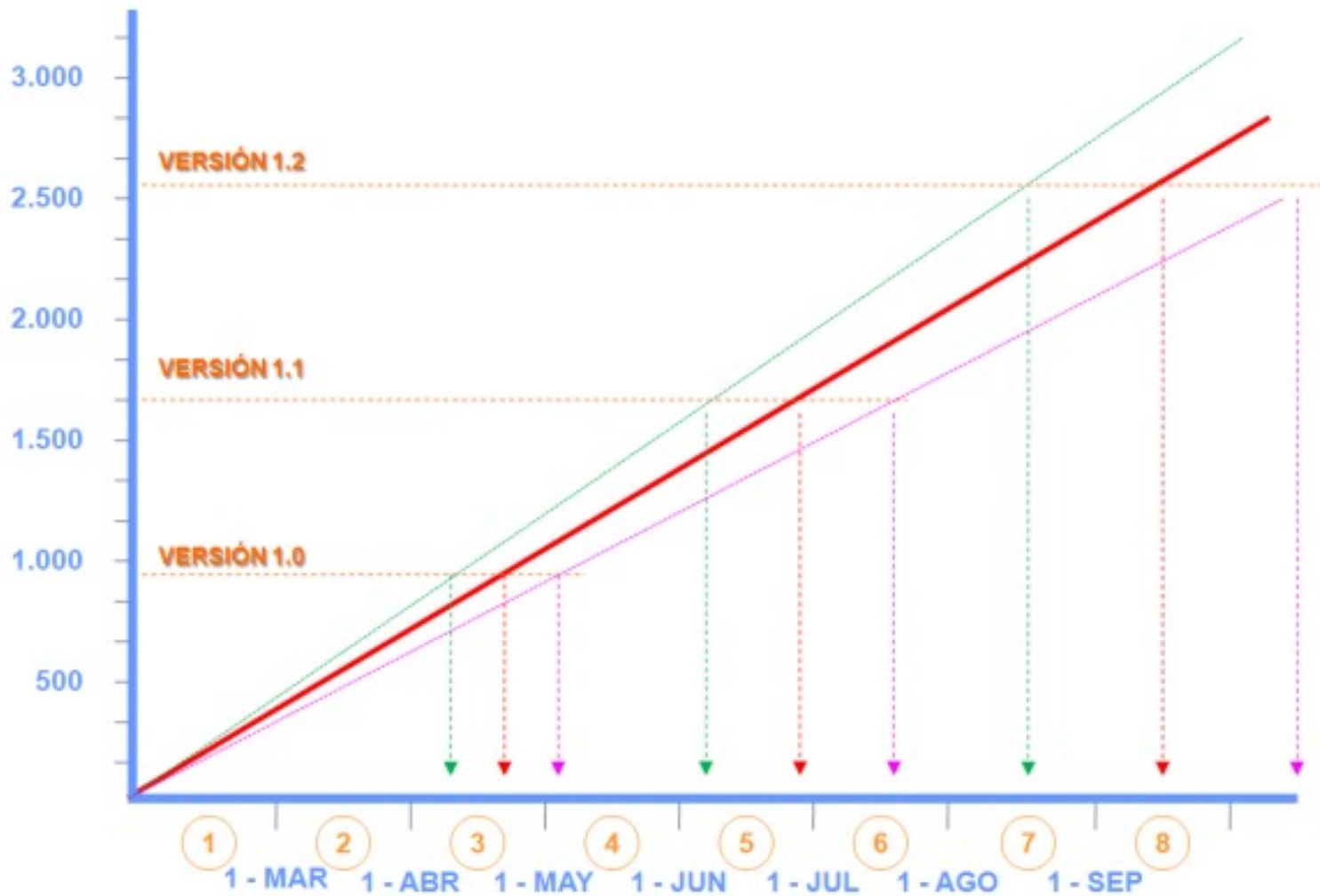
- Pila de producto:

Id	Temas	Trabajo	Criterio de validación	
1	Tema A 1.0	150	Lorem ipsum dolor sit amet	<b>Estimación: 950 PUNTOS</b>
2	Tema B 1.0	250	consectetur adipiscing elit	
3	Tema C 1.0	250	Aliquam vehicula accumsan tortor	
4	Tema D 1.0	300	Pellentesque turpis	
5	Tema A 1.1	250	Phasellus purus orci	<b>1.700 PUNTOS</b>
6	Tema D 1.1	350	penatibus et magnis dis parturient	
7	Tema E 1.0	150	Quisque volutpat ante sit amet velit	
8	Tema B 1.1	500	Cras iaculis pede eu tellus	<b>2.550 PUNTOS</b>
9	Tema C 1.1	150	Vestibulum vel diam sed pede	
10	Tema E 1.1	200	Suspendisse aliquam felis et turpis	
11	Tema F 1.0	TBD	Nullam imperdiet lorem vitae justo	<b>Versión 1.2</b>
12	Tema A.1.2	TBD	Suspendisse potenti. In nec nunc	
13	Tema B 1.2	TBD	Nam eros tellus, facilisis sed, pretium	
14	Tema F 1.1	TBD	Morbi arcu tellus, condimentum	

# Gráfico burn-up



# Gráfico burn-up



# Sprint Backlog

- **Definición y estimación de las tareas de programación** para cumplir los requerimientos del Sprint
- Participantes:
  - **Equipo de desarrollo**
  - Sólo el equipo lo puede modificar durante el Sprint
- Las estimaciones del backlog no son precisas, pueden ser revisadas si es necesario
- Las tareas de programación deberían ser atómicas para ser completadas por un solo miembro en un día laboral o entre 4 y 12 horas

# Sprint Backlog

- Si una tarea implica menos de 4 horas debería considerarse como parte de otra
- Si una tarea implica más de 12 horas debería considerarse dividirla en sub-tareas
- El objetivo es desglosar tareas a las que un miembro del equipo se puede comprometer a terminar de un Scrum diario a otro
- La suma del trabajo de las tareas de un elemento del backlog del producto debe completar el 100% del requerimiento al que pertenecen



# Sprint Backlog

- Formato:
  - Hoja de cálculo.
  - Pizarra física o pared.
  - Herramienta colaborativa o de gestión de proyectos.
- Escoger el formato más cómodo para todos

# Sprint Backlog

SPRINT	INICIO	DURACIÓN
1	1-mar-07	12

J
1-mar
23
276

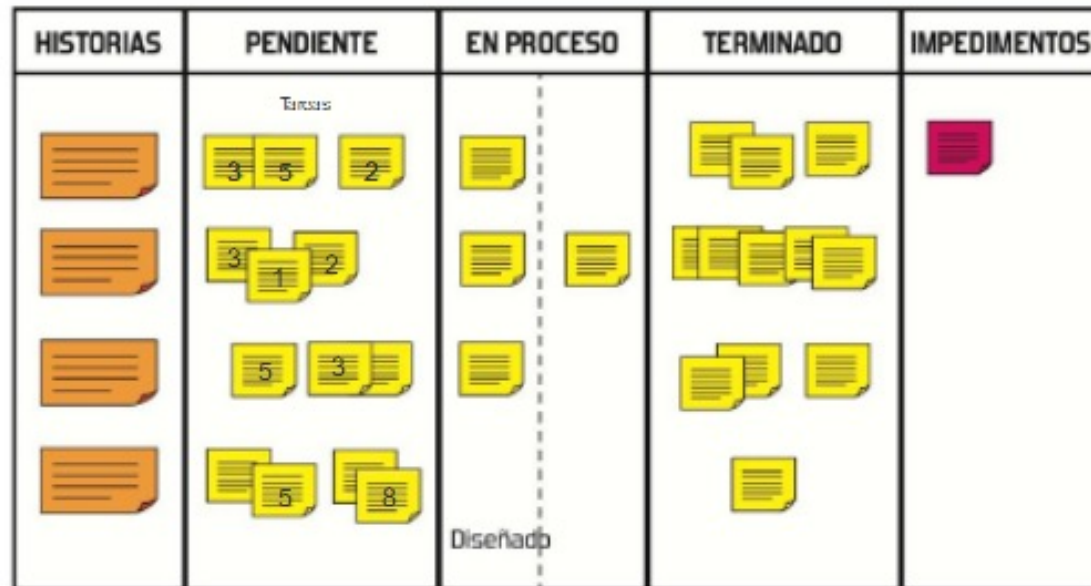
SPRINT BACKLOG			
Tarea	Estado	Responsal	
Descripción de la tarea 1	Terminada	Luis	16
Descripción de la tarea 2	Terminada	Luis	12
Descripción de la tarea 3	Terminada	Luis	4
Descripción de la tarea 4	Terminada	Elena	8
Descripción de la tarea 5	Terminada	Elena	16
Descripción de la tarea 6	Terminada	Elena	6
Descripción de la tarea 7	Terminada	Antonio	16
Descripción de la tarea 8	Terminada	Antonio	16
Descripción de la tarea 9	Terminada	Antonio	12
Descripción de la tarea 10	En curso	Luis	12
Descripción de la tarea 11	Pendiente	Luis	8

[illegible]

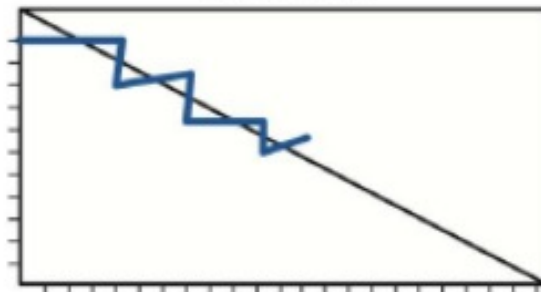
# Sprint Backlog

Panel Scrum

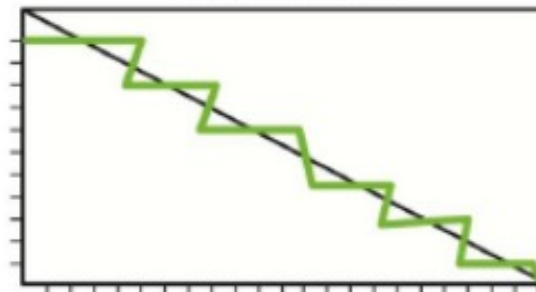
Gestión Visual



Release Plan



Sprint Burn-Down



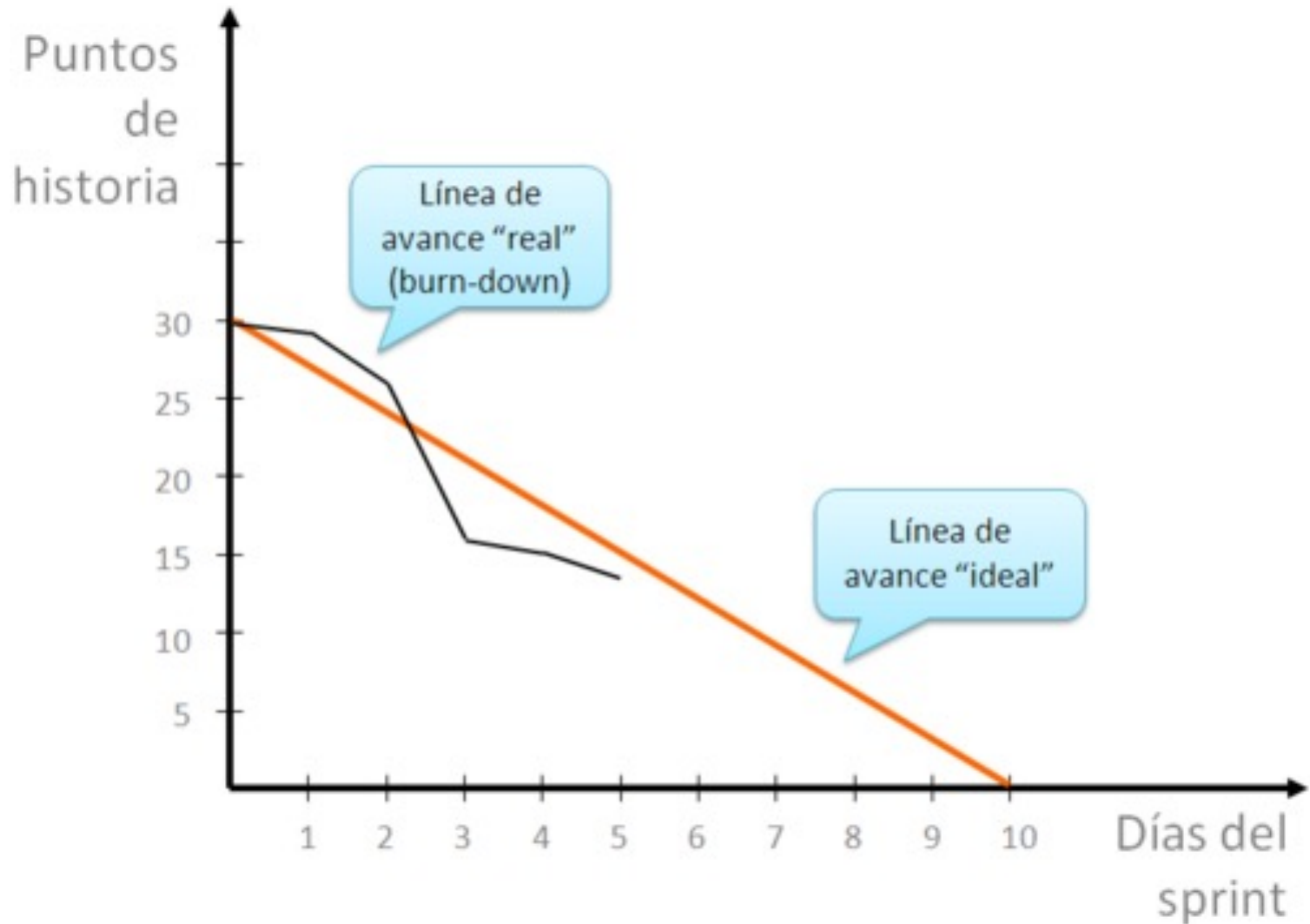
# Gráfico burn-down

- El **gráfico burn-down** en Scrum permite visualizar cuánto **tiempo** le queda al equipo **para finalizar el Sprint en curso**
- Sobre el **eje horizontal** se sitúan los **días hábiles** del Sprint
- Sobre el **eje vertical** se sitúan los **puntos de historia** (que tienen que quemar)
- La cantidad de puntos a quemar se realiza en la Planificación del Sprint ante el Dueño del producto

# Gráfico burn-down

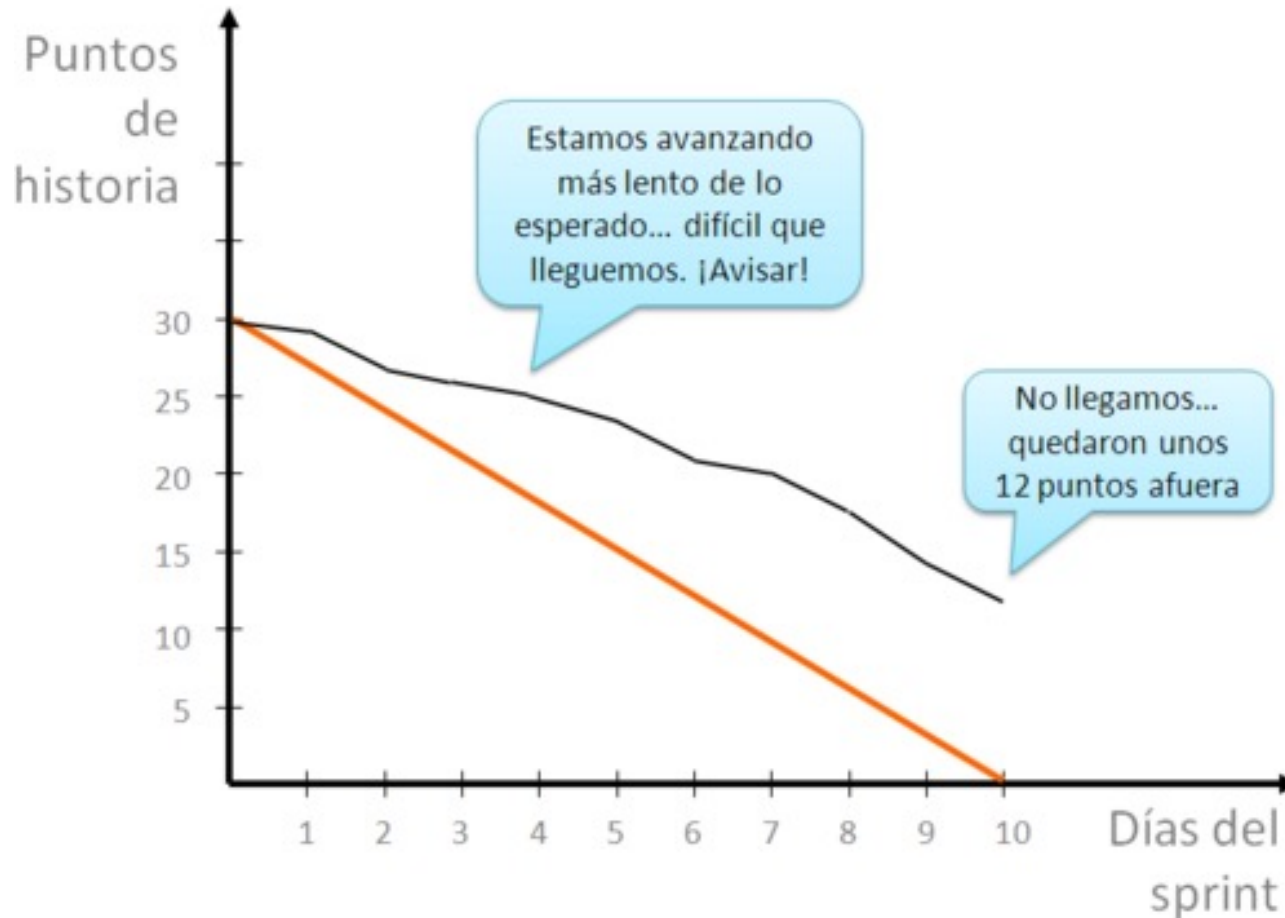
- Sabiendo la **duración del Sprint** y la cantidad de **puntos a quemar** se traza una línea recta que une ambos puntos
- Esa pendiente representará la **línea de quemado ideal**
- Idealmente el equipo deberá estar sobre esa línea para llegar a cumplir sus objetivos
- Este gráfico **se actualiza diariamente** durante la **reunión Diaria de Scrum**

# Ejemplo de gráfico burn-down



# Situaciones en un Sprint

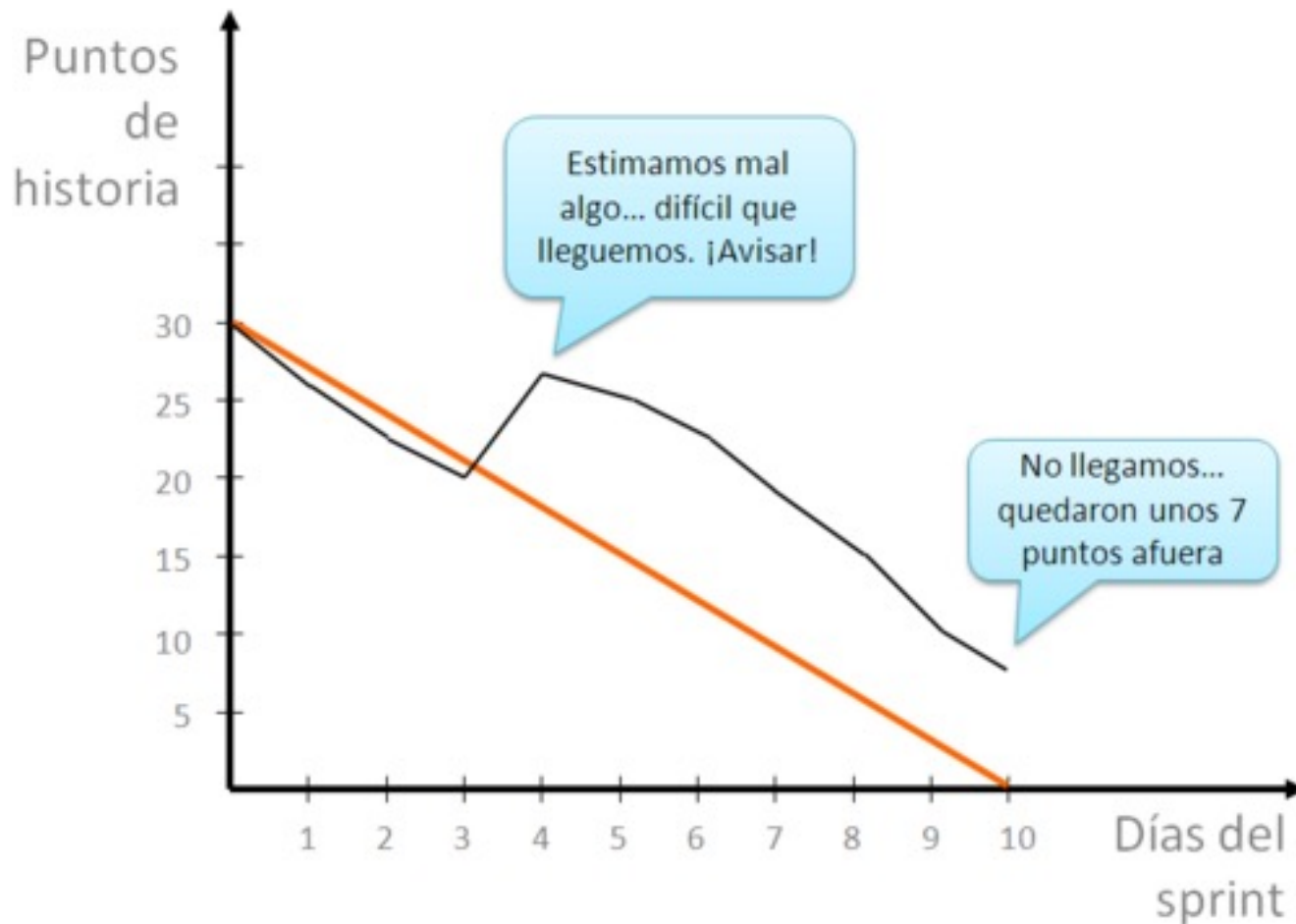
- Avanzando más lento a lo esperado





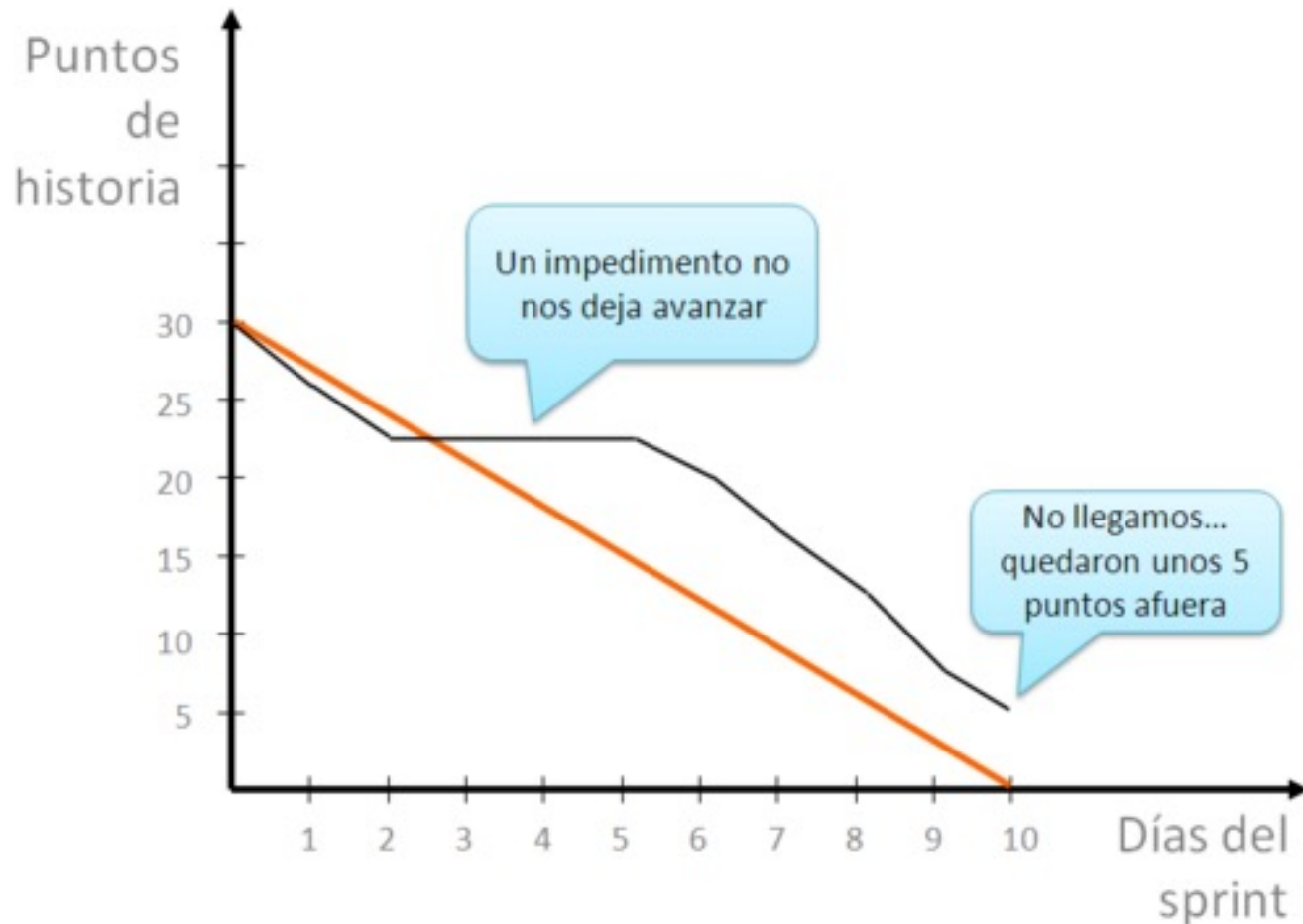
# Situaciones en un Sprint

- Historia mal estimada



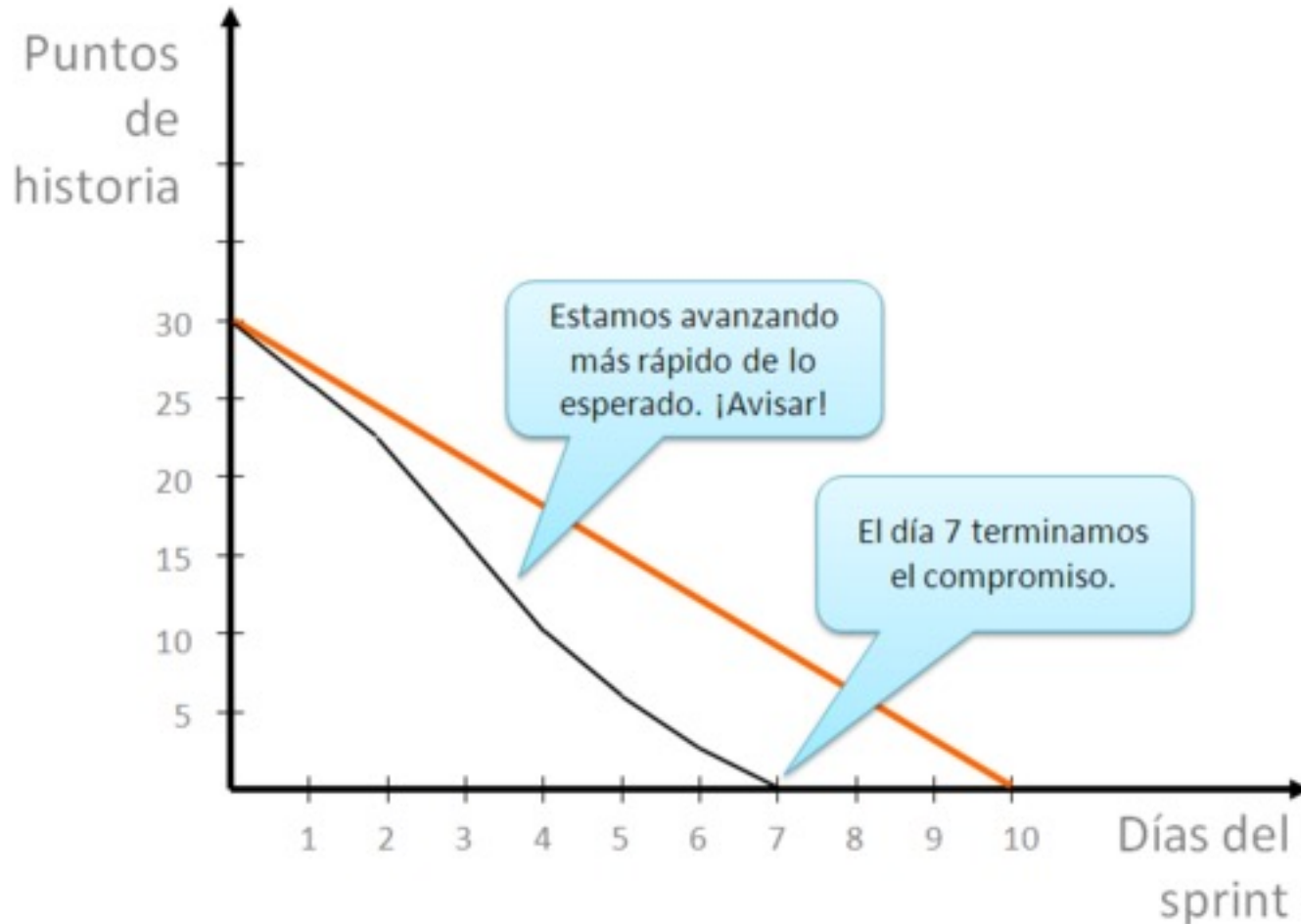
# Situaciones en un Sprint

- El efecto de un impedimento



# Situaciones en un Sprint

- Avanzando más rápido de lo estimado



# Valores

## Compromiso

- La metodología otorga a las personas la autoridad para cumplir sus compromisos

## Enfoque

- Haz tu trabajo. Enfoca todos tus esfuerzos y habilidades para trabajar en lo que te comprometiste

## Apertura/honestidad

- Todo lo relacionado con el proyecto es visible para todos

# Valores

## Respeto

- Es importante respetar a todas las personas del equipo y sus formas de pensar

## Coraje

- Tener el coraje para comprometerse, actuar, ser honesto y esperar respeto

# Resumen: Scrum



# Datos de uso de Scrum

- Empresas:
  - Microsoft, Yahoo, Google, Honda, Telefonica, Xerox, HP, Philips, Siemens, Nokia, Vodafone, BBC, Easyjet...
- Aplicaciones:
  - Software comercial, desarrollos internos, aplicaciones financieras, desarrollo de videojuegos, sitios Web, teléfonos móviles

# Errores comunes y malas interpretaciones

- Los equipos no se auto-dirigen, es el Scrum Master el que dirige y organiza al equipo.
- No hay actualizaciones diarias del Sprint Backlog por los miembros del equipo
- Se puede añadir trabajo nuevo a una iteración una vez comenzada
- El Product Owner no está involucrado o no tiene poder de decisión



# Errores comunes y malas interpretaciones

- No existe una revisión del Sprint
- La documentación no es necesaria
- Diseñar y realizar diagramas no es necesario
- Las reuniones Scrum pueden alargarse el tiempo que sea necesario
- Una iteración no termina con un producto parcial implementado y testeado
- Cada iteración debe terminar con un producto listo para comenzar su producción



# RESUMEN

# SCRUM

- Metodología centrada en **entregar software funcional en cada iteración**
- El trabajo se planifica en **Sprints de duración limitada**
- Primero realizar el **Sprint 0**
  - Formar el equipo
  - Visión y objetivo del proyecto
  - Visión del dueño del producto
  - Definir horario para reunión diaria de Scrum

# SCRUM

- **Planificación del Sprint**
  - **Parte1:** El dueño del producto prioriza las tareas obteniendo el **backlog del producto**
  - **Parte2:** El equipo divide las tareas creando el **backlog del Sprint**
- **Reuniones diarias de Scrum**
  - **Sincronización** de actividades
  - **Actualización** del estado del proyecto

# SCRUM

- **Revisión del Sprint**

- Al finalizar el Sprint se proporciona un **incremento del producto** mostrado a cualquier interesado en el proyecto (dueño del producto, usuarios del negocio,...)
- Se añade al **backlog del producto** el feedback que de nuevo es priorizado por el dueño del producto

- **Retrospectiva del Sprint**

- Se realizan **mejoras en la forma de trabajo**

- **Sprint de release**

- Tras varios Sprints el equipo obtiene un **producto preparado para ser puesto en producción**

# SCRUM

- **Artefactos y herramientas:**
  - **Pila de producto:** lista de funcionalidades que desea el **cliente**, ordenadas según la **prioridad** para él. Es un documento vivo, en constante evolución durante el desarrollo del sistema.
  - **Pila de Sprint:** lista de tareas realizada por el **equipo** de desarrollo en las que se han descompuesto las funcionalidades de la pila del producto que se van a desarrollar en un Sprint . Para cada tarea de la pila del Sprint se indica la persona que la tiene asignada y el tiempo de trabajo previsto.
  - **Gráficas burn-up y burn-down**

# Bibliografía

- Succeeding with Agile : software development using Scrum. Mike Cohn. Addison Wesley
- Agile project management with Scrum. Ken Schwaber. Microsoft
- The SCRUM Primer. Pete Deemer et al. Scrum Training Institute
- SCRUM y XP desde las trincheras. Henrik Kniberg. Info Q. Enterprise Software Development Series