

08/05

## SCRUM

### Filosofía de Scrum

"En un Scrum, hay que actuar como una unidad, no como 8 individuos. Todos tienen un rol. Nunca debemos olvidar que cuando trabajamos juntos como una unidad, el todo es más que la suma de las partes"

The On-line Rugby Coaching Manual

### Autores Scrum

- Ken Schwaber = desarrolló y formalizó Scrum para el desarrollo de sistemas.
- Jeff Sutherland = pensamiento y prácticas iniciales, previo a su formalización con Ken.
- Mike Beedle = Utilizó y mejoró Scrum, integrándolo a XP.

### ¿Qué es SCRUM?

- Scrum es un framework que permite crear su propio proceso para crear nuevos productos.

• Scrum es simple, puede ser implementado en pocos días, pero puede tomar mucho tiempo perfeccionarlo.

"Scrum no es una metodología - es un camino"

Ken Schwaber

Scrum es un proceso que está en el contexto de procesos empíricos para poder crear nuevos productos a través del desarrollo de JWS. Es un proceso simple, empírico pero que trabaja con ciertas definiciones concretas. Scrum define lineamientos de cómo trabajar, elementos claros que deben cumplirse.

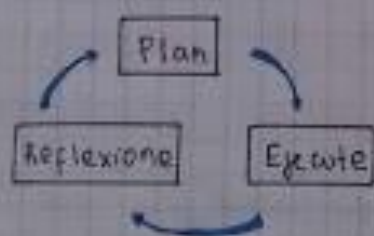
Scrum es empírico: las metodologías rigurosas se basan en métodos definidos, con la idea de línea de ensamble. El control en Scrum se alcanza con inspecciones frecuentes y correspondientes ajustes/inspección de partes.

¿Cuáles son los valores de Scrum?

- ✓ Coraje
- ✓ Foco
- ✓ Compromiso
- ✓ Respeto
- ✓ Apertura

Están vinculados a los valores del manifiesto ágil y van en línea con las metodologías ágiles.

### El ritmo de Scrum



Tiene que ver con planificar, ejecutar, analizar cómo fue esa ejecución, tomar las acciones correctivas necesarias y volver a planificar, con acciones correctivas que hagan más planificado y reflexionado.

Y así iterar las veces que hagan falta para crear el producto deseado.

### Cimientos de Scrum

- 1) **EMPIRISMO** = ya que se trata de un framework que trabaja bajo las características de un proceso empírico, no de procesos definidos o de metodologías tradicionales.
- 2) **AUTO-ORGANIZACIÓN** = entre los miembros del equipo y donde los equipos tienen características de autoorganizarse y esto implica que las personas que forman los equipos deben poder ser autónomas y poder autoorganizarse para poder cumplir con los objetivos que se plantean.
- 3) **COLABORACIÓN** = entre los miembros del equipo, entre el Product Owner y todas las personas que participan en la construcción y tienen algún rol dentro del framework.
- 4) **PRIORIZACIÓN** = hace referencia en concentrarse en aquello que se define con mayor prioridad. Se presta mucha atención en eso de la mayor prioridad.
- 5) **TIME BOXING** = donde tiene que ver lo de las iteraciones y el trabajo que se hace se realiza planteando límites de tiempo concretos.

En el proceso de Scrum lo que está fijo es el alcance y lo que se modifica es el tiempo que nos va a llevar cubrir ese alcance.

Esto nos sirve para asegurarnos que cada tanto el cliente recibirá esa entrega.

### EMPIRISMO

El empirismo es una teoría filosófica que enfatiza el papel de la experiencia, ligada a la percepción sensorial, en la formación del conocimiento, no sólo en cuanto a su origen sino también en cuanto



a su origen sino también en cuanto a su contenido. Se parte del mundo sensible para formar los conceptos y estos encuentran en lo sensible su justificación y su limitación.

Procesos definidos y planificación detallada en las primeras fases son reemplazados por ciclos de inspección y revisión just in time y ciclos adaptativos.

### AUTO-ORGANIZACIÓN

Es un proceso en el que la organización interna de un sistema, generalmente abierto, aumenta de complejidad sin ser guiado por ningún agente externo. Normalmente, los sistemas auto-organizados exhiben propiedades emergentes.

La auto-organización es objeto de estudio interdisciplinar, pues es una propiedad característica de los sistemas complejos, ya sean estos matemáticos, físicos, químicos, biológicos, sociales o económicos.

Pequeños grupos de trabajo que manejan su propia carga de tareas y se organizan entre ellos alrededor de un objetivo claro y tomando en cuenta las restricciones.

### COLABORACIÓN

Se refiere abstractamente a todo proceso donde se involucre el trabajo de varias personas en conjunto. También cuando ayuda a una persona a hacer algo que se le dificulta, o en caso de que no pueda hacerlo.

Líderes de surum, diseñadores de productos y clientes colaboran con los desarrolladores - ellos no lo gerencian o dirigen.

## Priorización

Da prioridad a alguna cosa.

Trabajar en lo más importante - no perder el tiempo haciendo  
foco en el trabajo que no tiene y lo agregar valor.

## Time boxing

Es una técnica de planificación en proyectos, típicamente de software, donde el schedule (programación) es dividido en un número determinado de periodos de tiempo (time box), normalmente entre 2 y 6 semanas, los cuales tienen sus propias entregables, fechas y costos.

Timeboxing crea el ritmo que guía el desarrollo.

Entonces, cómo trabaja Scrum?

- ✓ Equipos pequeños (<10 personas, ideal 7+-2)
- ✓ una serie de Sprints (30 días) → duración fija
- ✓ Incrementos visibles y usables
- ✓ Time-boxing
- ✓ Auto-organización
- ✓ Colaboración
- ✓ Priorización

Scrum plantea una forma de trabajo diferente...

- Menos tiempo planeando y definiendo tareas
- Menos tiempo creando y leyendo reportes
- Se pasa más tiempo con el equipo investigando la solución
- Scrum → supone dominios impredecibles = no se debe definir el alcance completo desde un principio, sino que

varial priorizando con el PO a medida que avanza.

→ No supone un proceso repetible.

### Roles en Scrum

#### ● PRODUCT OWNER

- Controla y gestiona el Backlog.
- Una persona, no un comité.
- Nadie puede decirle al equipo una prioridad diferente.
- Acciones visibles.

Es el experto en el dominio y se encarga principalmente de el product backlog, quien es el que prioriza los items en función del valor que aporta al negocio. El SO tiene que ser una persona que esté capacitada para tomar decisiones y para definir prioridades.

#### ● SCRUM MASTER

- Es responsable de que las prácticas, valores y reglas se realicen.
- Nexo entre la gerencia y el equipo.
- Dirige los Scrum diarios, comparando el progreso planeado con lo real.
- Se asegura que se resuelven los impedimentos y se toman decisiones

rápidas.

- Trabaja con la gerencia y el cliente para identificar el dueño del producto.

- El Scrum Master, el dueño del producto y el equipo producen un Backlog de producto.

Actividades del Scrum Master:

- ✓ Toma decisiones en reuniones de Scrum
- ✓ Registro y resuelve problemas
- ✓ Mantiene el equipo enfocado



✓ Realiza el seguimiento de avance

Responsabilidades:

✓ Mantener las reuniones diarias cortas mediante la aplicación de las reglas:

- Hora y lugar fijos en lo posible

- Los gerentes pueden asistir pero sólo los desarrolladores participan

- Una reunión de Scrum no es una reunión de diseño.

La persona que cumple este rol de Scrum master es uno más del equipo. Es quien mejor conoce el framework y ayuda al equipo a salvar impedimentos, a establecer como tienen que trabajar. Es un facilitador.

### ● EL EQUIPO

- 7+2 personas. Si hay más de 8 personas, hacer varios equipos trabajando sobre el mismo backlog.

- Autórganos y autoorganizados

- Compromiso de entregar un conjunto de items del Backlog al final de un Sprint.

- Libertad de acción

- Limitado por estándares y convenciones organizacionales.

- No hay roles

- Todos codifican.

Dinámica del equipo:

✓ Scrum pretende ser una sombrilla bajo la cual el equipo puede dar lo mejor

✓ Scrum es empírico, y a veces se alcanza el objetivo reduciendo funcionalidad.

Tiene que estar integrado por integrantes motivados. No hay roles dentro del equipo, sino que todos hacen todo. Tiene que ser autoorganizado y trabajar de manera colaborativa.

### Ambiente de trabajo

- Abierto. El silencio absoluto es un mal signo.
- Pizarrones.
- El equipo define sus horarios.

### Entregables

#### ● EL BACKLOG DE PRODUCTO (PRODUCT BACKLOG)

- "Es una lista priorizada de funcionalidades técnicas y de negocio, que necesitan ser desarrolladas".
- El Backlog del producto contiene la lista de requerimientos.
- Se listan características, funciones, tecnologías, mejoras, bugs, etc que serán aplicadas al producto.
- El Backlog está incompleto inicialmente, aunque sólo se necesita lo suficiente para realizar el primer Sprint de 30 días.

En la lista de requerimientos se tienen aquellos items que tienen mayor prioridad en función del valor de negocio. El product Backlog no necesariamente tiene que estar definido de manera completa, lo que sí debe definirse de manera completa es lo que está arriba de la pila y que es lo que se va a incluir en el próximo Sprint, lo demás puede estar como épicas, etc y se van a ir definiendo su prioridad en la lista.

El backlog se origina en





## ISW

## TEÓRICO

- Desarrollo
- Soporte al cliente

Características del Backlog

priorizado

también contiene problemas a ser solucionados, y que son dependencias de otros ítems del backlog.

Asteroides ... ¿cómo refina un product backlog?

- Trabajar con un backlog de producto es como jugar a Asteroides.
- Grandes rocas (epics) se rompen en pedruzcos de rocas más pequeñas (stories) hasta que son lo suficientemente pequeñas para ser cumplidas (desarrolladas y entregadas)



Lo más grande queda abajo, lo más pequeño y lo que está arriba son las tareas que tienen que cumplir con el modelo invest. Todas las user que quedan arriba del backlog deben cumplir con ese modelo invest.

Estimaciones del Backlog

- Las estimaciones son un esfuerzo colaborativo entre las partes.
  - Las estimaciones son iterativas.
  - Si un ítem no puede ser debidamente estimado, se debe dividir en el Backlog.
  - Las estimaciones sirven de base para la funcionalidad en el sprint.
- La user debe ser estimable y pequeña según el modelo invest. La técnica usada para estimar es el pocket planning, y como medida de estimación se usa la story point.

## PRODUCT BACKLOG

- Los requisitos (tareas de Product Backlog, de mayor prioridad.)
- Una lista de todos los trabajos deseados en el proyecto.
- Idealmente cada ítem tiene valor para el usuario o el cliente.
- Priorizada por el Product Owner
- Repriorizada al comienzo de cada Sprint

Una vez que se definen los requisitos o requerimientos que se harán en el próximo Sprint, se logra el segundo de los entregables que es el Sprint Backlog.

## ● BACKLOG DEL SPRINT

- El equipo determina que debe hacerse para cumplir el objetivo del Sprint
- El dueño del producto suele asistir
- Se realiza una lista de tareas que tomen de 4 a 6 horas para completarse
- El Backlog no se modifica durante el desarrollo
- Si el equipo descubre que el Backlog no puede realizarse, el Scrum Master y el dueño del producto determinan si:
  - Algun ítem puede ser removido del backlog
  - Alguna funcionalidad puede eliminarse

El Backlog del Sprint es una porción del Product Backlog que el equipo decide hacer o ejecutar en el próximo Sprint. En conjunto con el Product Owner definimos los users y los requerimientos.

Para poder cumplir con el Sprint Backlog se hace una lista de tareas para poder cumplir con cada history point. Esta lista de tareas no debe demorar más de 16 horas y menos de 4 para poder cumplirlos.

## ISW

### TEORICO

#### ● PRODUCTO DE SW POTENCIALMENTE LISTO PARA PRODUCCIÓN

Parte del producto desarrollado en un sprint en condiciones de ser usado en el ambiente correspondiente

Cuando termina el sprint si así se debe tener un entregable del producto que se puede mandar a producción.

### Reuniones

#### ● Reunión de planeamiento de Sprint

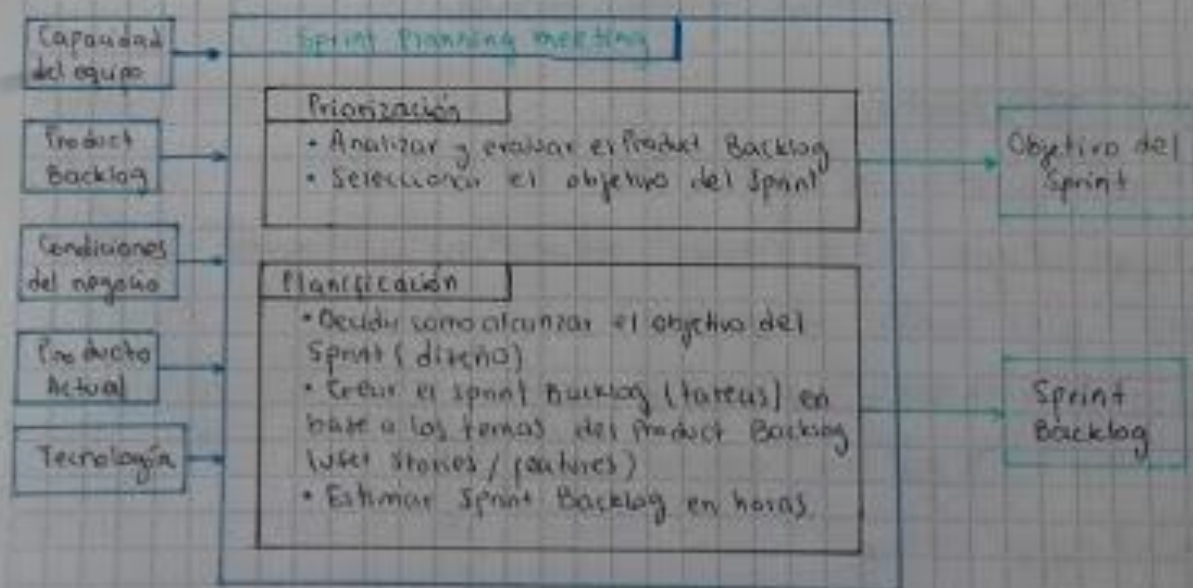
✓ Dos reuniones consecutivas

• El equipo se reúne con el dueño del producto, la gerencia y los usuarios para decidir la funcionalidad a implementar

• El equipo decide cómo llevarla a cabo

✓ Inputs: Backlog, último incremento, métricas

En resumen, un sprint planning meeting es:



Sprint Planning  
Daily  
Sprint Demo/review  
Retrospectivas

} Reuniones  
obligatorias

Story time /  
Product Backlog  
Refinement  
(optional)

} opcional



En la Sprint Planning se debe priorizar, elegir cuales son los requisitos de la User que vamos a incluir en el Sprint. Se define para el próximo Sprint: el objetivo, como se va a alcanzar ese objetivo, es decir, cuales son los users que van a estar involucrados, crear el Sprint backlog, las tareas y lograr una estimación del Sprint backlog en horas.

#### ● DAILY MEETING • Reuniones de Scrum diarias

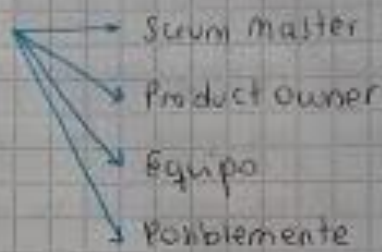
- Cortas (15 a 30 min). Se analiza el estado. Facilitador = Scrum Master.
- Asiste todo el equipo
- No para soluciones de problema.
- El Scrum Master pregunta 3 cosas a los asistentes =
  - ✓ ¿Qué se completó desde la última reunión?
  - ✓ ¿Qué obstáculos se presentaron?
  - ✓ ¿Qué va a hacer hasta la próxima reunión?
- No es dar un status report al Scrum Master
- Se trata de compromisos delante de pares.

#### ● SPRINT DEMO / REVIEW

- Reunión informativa de 4hs.
- El equipo presenta el incremento desarrollado a gerentes, clientes, usuarios y el dueño del producto.
- Informal
  - Regla de 2hs de preparación
  - No usar diapositivas.
- Todo el equipo participa
- Se invita a todo el mundo
- Se reportan los problemas encontrados.
- Cualquier item puede ser agregado, eliminado, re-priorizado.
- Se termina el nuevo Sprint y se asignan tareas.

### ● SPRINT RETROSPECTIVE

- Periódicamente, se echa un vistazo a lo que funciona y lo que no
- Normalmente 1hs a 4hs
- Se realiza luego de cada sprint
- Todo el equipo participa



Esta reunión abarca mas a la reflexión entre todo

el equipo se debe determinar

que cosas funcionaron en el sprint (para repetir las) y que cosas no funcionaron para poder corregirlas

Es importante encontrar la manera de que todo el equipo participe y cada uno de su punto de vista. Se usan ceremonias en anonimato para que nadie participe más

### ● STORY TIME

- El equipo se reúne con el PO para discutir items del backlog de alta prioridad, determinar su criterio de aceptación y asignar la priorización a cada nuevo item.
- Esto ocurre inicialmente antes del desarrollo y después iterativamente en cada Sprint

Se realiza antes de empezar a desarrollar los users

## SPRINT

- Período fijo de tiempo, sugiriendo 30 días
- Dentro del Sprint
  - No se puede cambiar el alcance
  - No se puede agregar funcionalidad
  - No se pueden modificar las reglas del equipo

Se puede hacer de menor días, eso lo decide el equipo. Una vez que se define el objetivo del Sprint backlog, no se puede cambiar ni modificar.

Todo lo que sucede luego del Sprint, permite hacer las modificaciones necesarias para el próximo Sprint, en función de lo aprendido en el Sprint anterior.

Todos los Sprints tienen un objetivo que se define en la planning y donde se define lo que se va a hacer.

- Objetivo del Sprint
- Una declaración
  - Un objetivo claro a alcanzar

### Reglas de un Sprint

- Foco en la tarea
- Sin interrupciones
- Sin cambios
- El mismo equipo puede descubrir más trabajo a ser hecho.

Mientras se ejecuta el Sprint, no puede haber ni cambios ni interrupciones.

### Elementos de un Sprint

- Reuniones de Scrum
- Se produce un incremento usable y visible (Daily)
- Los incrementos se basan en un producto anterior.
- Compromiso de los miembros a la asignación (Equipo)

### Mecánica de los Sprints

- En un Sprint se congelan 3 de las 4 variables de un proyecto:
  - Tiempo = 30 días



- Costo = salarios + ambiente
- Calidad = generalmente en estándares organizacional

• El equipo puede cambiar funcionalidad siempre y cuando cumpla con el objetivo del Sprint.

El ambiente se asegura y se establecen reglas para trabajar durante el Sprint. Pueden haber cambios mínimos que se coordinen con el Sprint siempre y cuando el objetivo del Sprint sea el mismo.

### Tareas obligatorias

Es un sprint hay

2 tareas mandatorias

- Reuniones Scrum donde participan todos los miembros del equipo.
- Backlog del Sprint - debe estar actualizado y con los últimos eliminados de los desarrolladores.

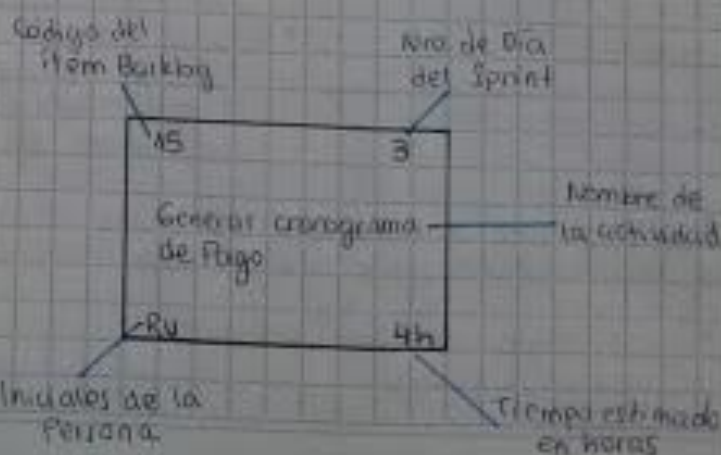
### Cancelación de un Sprint

- Se puede cancelar un Sprint si las circunstancias hacen que no sea necesario.
  - ✓ El objetivo se vuelve obsoleto.
  - ✓ Las condiciones técnicas o de mercado cambian.
- Decidido por el equipo.
  - ✓ Porque no se puede alcanzar el objetivo.
  - ✓ Se encuentra un problema muy grande.
- Cancelar un Sprint es un costo.

### Herramientas de Scrum

#### ③ TASKBOARD / TARJETA

- Código del Item Backlog
- Número del Sprint
- Nombre de la actividad
- Iniciales de la Persona
- Tiempo estimado en horas



## ③ TASKBOARD / TABLERO

- Story
- To do
- Work in Progress
- To verify
- Done

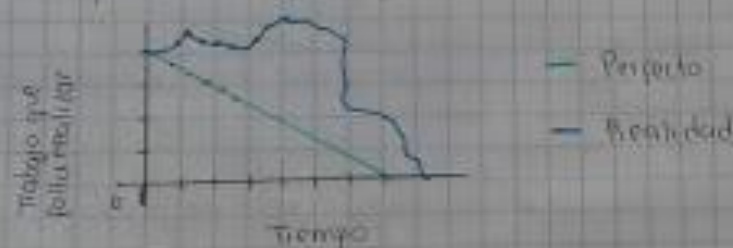
Story	To do	In Progress	To verify	Done
la que se va a hacer	la que se va a hacer	la que se va a hacer	la que se va a hacer	la que se va a hacer

Para que tenga buena cantidad de áreas descomponiendo entre 4 y 6.

## ④ GRÁFICOS DE BACKLOG

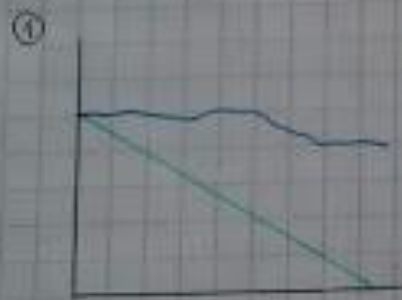
- la gerencia necesita datos acerca de:
  - Ingreso del Sprint
  - Progreso de Release
  - Progreso del Producto
- El backlog de trabajo es la cantidad de trabajo que queda por ser realizado.
- Tendría del Backlog = trabajo que queda vs tiempo.

Estos gráficos sirven para poder medir el progreso o cuanto queda por hacer



Cada punto muestra lo logrado o el trabajo realizado, cuanto queda por hacer y cuanto ya hemos. Todo esto forma el Burndown Chart.

### Burndown Chart



### Diagnostico?

- No todo lo incluido en el sprint será entregado en la fecha propuesta. El Backlog no fue modificado para alcanzar la fecha (remover requerimientos o modificar la fecha).
- Posibles causas:
  - Faltas técnicas de estimación
  - Mal manejo de riesgos
  - Rotación del equipo o renuncias

- Este gráfico muestra que el Scrum Master (SM) y el Product Owner (PO) no reaccionaron para evitar esta situación.
- Entre llegar a este estado.

No se conoce bien y no se puede cumplir con el objetivo. Queda trabajo por hacer y parte que no haya habido un intento para plantear las consecuencias. El resultado del Product Backlog es que no se pudo cumplir.

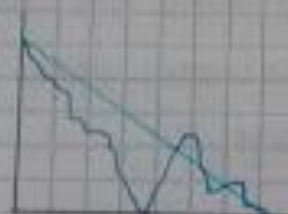
②



Diagnóstico?

- Este SC es mucho mejor que el anterior.
  - Al comienzo del sprint hubo problemas, pero en el medio el equipo de trabajo reaccionó y tomó medidas correctivas por lo que se nota que sí hubo requerimientos para mantener la fecha de release.
  - También pone en evidencia un mejor manejo de riesgos lo que facilitó aumentar la velocidad en la segunda parte del sprint.
- En este caso se tomaron las acciones correctivas a tiempo.

③



Diagnóstico?

- Este SC también tiene características negativas.
- Este no es un ejemplo de desarrolladores con las mejores habilidades técnicas. Demuestra pobres técnicas de estimación.
- El PO y el SM deberían haber agregado más requerimientos para el release.

Puede haberse sobreestimado, entonces se agregaron cosas y funcionalidad. Claramente hay problemas en la estimación. También pensar que lo que se había terminado en realidad aún no estaba listo.

④



Diagnóstico?

- Ideal
- Buenas técnicas de estimación
- Buena velocidad de desarrollo
- Buen manejo de impedimentos
- Todos los requerimientos prometidos son los entregados
- Buen manejo de la asignación de tareas (i drew cards, reporte diario)



- Quizás demasiado bueno para ser verdad.
- Nunca la proyección lineal es perfecta, se cumple con lo prometido.

### Beneficios de Scrum

- ✓ Se gestionan los cambios de requerimientos.
- ✓ Se inspira la unión de mercado.
- ✓ Los clientes ven incremental que refina los requerimientos en un tiempo razonable.
- ✓ Mejores respuestas con el cliente.

### Scrum en 10 palabras

Algunos usuarios de Scrum:

Microsoft, Yahoo, Google, Electronics Arts, High Moon Studios, Lockheed Martin, Philips, Siemens, Volvo, Capital One, BBC, Intuit, Omnicor Media, First American Real Estate, BMC Software, Ipswitch, John Deere, Lexis Nexis, Sabre, Salesforce.com, Time Warner, Turner Broadcasting, etc.

- ✓ Scrum es un proceso ágil que nos permite contrarnos en ofrecer el más alto valor de negocio en el menor tiempo.
- ✓ Nos permite rápidamente y en repetidas ocasiones inspeccionar software real de trabajo (cada 2 semanas o un mes).
- ✓ El negocio fija las prioridades los equipos se auto-organizan a fin de determinar la mejor manera de entregar las funcionalidades de más alta prioridad.
- ✓ Cada dos semanas o un mes cualquiera puede ver el software real funcionando y decidir si liberarlo o seguir mejorándolo en otro sprint.

### Se ha utilizado Scrum para:

- |                            |  |
|----------------------------|--|
| • Software comercial       | • Desarrollo de videojuegos                                |
| • Documentos internos      | • Software críticos de soporte vital, aprobados por la FDA |
| • Desarrollo bajo contrato |  |

## ISW

### TEÓRICO

- Proyecto Fixed-price
- Aplicaciones financieras
- Aplicaciones certificadas ISO 9001
- Sistemas embebidos
- Sistemas con requisitos 7x24 y 99.999% de disponibilidad
- Joint Strike Fighter

- Software de control satelital
- Intel web
- Software para handheld
- Teléfono portátil
- Aplicaciones de network switching
- Aplicaciones de ISE
- Algunas de las más grandes aplicaciones en WO

Ver 2 imágenes en el PDF.

Cambios en el Sprint? No, y que se entienda, muchas implementaciones exitosas se basan en diversos lastrados de los sprints para contener los cambios fuera del sprint.

Pensamiento Foco: "Scrum no es para todos, sino para aquellos que tienen que trabajar con sistemas funcionando con la complejidad de tecnologías inestables y el surgimiento de requerimientos".

### MÉTRICAS ÁGILES

Regla de Oro Ágil sobre Métricas:

- La medición es una salida, no una actividad.

- Una filosofía minimalista sobre las métricas = Medir lo que es necesario y nada más.

El contenido: "El software funcionando" nos va a servir para definir las métricas que se van a usar. Este contenido es del manifiesto ágil.

Los principios ágiles que guían la elección de las métricas:

"Nuestra mayor prioridad es satisfacer al cliente por medio de entregas tempranas y continuas de software valioso."

"El software funcionando es la principal medida de progreso."

### Capacidad

Es una estimación de cuánto trabajo puede completarse en un periodo de

tiempo dado. Basado en la cantidad de tiempo ideal disponible del equipo.

- Se puede medir en:
  - Esfuerzo (horas)
  - Puntos de Historia (Story Points)

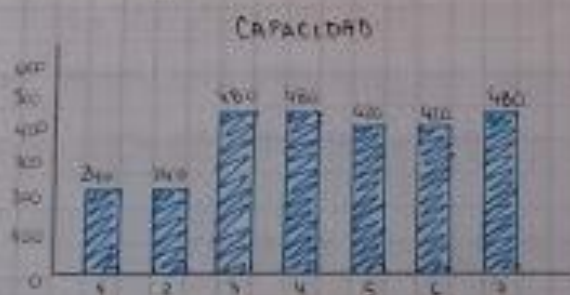
La capacidad se define como métrica para esto. basados en la estimación. lo ideal sería definir la capacidad en Story Points, porque como las user y features las tenemos definidas en Story Points, si se quiere decir cuanto trabajo se va a hacer la ideal es usar Story points.

- Horas de Trabajo Disponibles por día (WH) x Días Disponibles Iteración (DI) = Capacidad

$$WH \times DI = \text{Capacidad}$$

- Ejemplo:
  - ✓ Equipo de 6 miembros
  - ✓ 4 miembros disponibles los 2 primeros sprints
  - ✓ 1 miembro se casa en sprints 5 y 6
  - ✓ 6 horas de trabajo

Sprint	1	2	3	4	5	6	7	Total
Horas	240	240	480	480	420	420	480	2360
Puntos de Historia	30	30	45	60	58	51	60	335



la capacidad se ratifica en la Planning.

#### Consideraciones:

- Incluir la estimación capacidad histórica
- Aplicar estimaciones honestas a sus tareas.
- Considerar una capacidad máxima de 50% - 70%
- Comparar la capacidad a largo plazo con la velocidad y los puntos de historia



## ISW

### TÉCNICO

- ¿Cuál es el promedio de puntuación de un punto de historia para un equipo/individuo?

### Velocidad

- Es una observación empírica de la capacidad del equipo para completar el trabajo por iteración. Es concretamente una medida.
- Comparable entre iteraciones para un equipo dado y un proyecto dado.
- No es una estimación, un objetivo a alcanzar, comparable entre equipos, y comparable entre proyectos.

Se va a obtener en la ceremonia de la Review, y para medir la velocidad se debe sumar la cantidad de puntos de historia de las features o de las user stories que han sido probados por el Product Owner.

### Unidad de medida de la velocidad

Cómo planea el equipo	Unidad en medida
Compromiso con las historias	Historias
Tamaño relativo (puntos)	Puntos de historia
Estimación (horas ideales)	Horas ideales

¿Qué cuenta para la velocidad? Solo cuenta el trabajo completado para la velocidad.

- la capacidad es una estimación, entonces se trata de predecir cuánto trabajo vamos a poder realizar en el próximo sprint.
- la velocidad sirve como input para refinar la capacidad pero además sirve para poder ver cuál es la medida de trabajo que el equipo va realizando. Sirve de retroalimentación para la capacidad y además para ver niveles de diagramas como va trabajando el equipo. El equipo además debería en algún momento ser estable.