



UNIDAD 03

SCRUM

SCRUM

- ▶ Scrum es una metodología ágil para gestionar proyectos de software. Aunque surgió como práctica en el desarrollo de productos tecnológicos, resulta válido en los entornos que trabajan con requisitos inestables, y necesitan rapidez y flexibilidad; situaciones habituales en el desarrollo de algunos sistemas de software.

SCRUM

- Scrum controla de forma empírica y adaptable la evolución del proyecto, con las siguientes prácticas de la gestión ágil:
 - Revisión de las Iteraciones
 - Desarrollo Incremental
 - El desarrollo incremental implica que al final de cada iteración se dispone de una parte del producto operativa que se puede inspeccionar y evaluar.
 - Desarrollo Evolutivo
 - Como modelo ágil, es útil en entornos con incertidumbre e inestabilidad de requisitos. Intentar predecir en las fases iniciales cómo será el resultado final, y sobre dicha predicción desarrollar el diseño y la estructura del producto no es realista, porque las circunstancias obligarán a remodelarlo muchas veces.

SCRUM

- Auto-organización
 - La gestión predictiva confía la responsabilidad de su resolución al gestor de proyectos. En Scrum los equipos son auto-organizados, con margen de decisión suficiente para tomar las decisiones que consideren oportunas.
- Colaboración
 - Las prácticas y el entorno de trabajo ágiles facilitan la colaboración del equipo, que es necesaria y debe basarse en la colaboración abierta entre todos según los conocimientos y capacidades de cada persona, y no según su rol o puesto.

SCRUM ELEMENTOS

- ▶ *Product Backlog*
- ▶ *Sprint Backlog*

- **Product Backlog:**

- Requisitos del sistema. Se parte de la visión del resultado que se desea obtener; y evoluciona durante el desarrollo.
- Es el inventario de características que el propietario del producto desea obtener, ordenado por orden de prioridad.
- Es un documento “vivo”, en constante evolución. Es accesible a todas las personas que intervienen en el desarrollo. Todos pueden contribuir y aportar sugerencias.

SCRUM – PRODUCT BACKLOG

[illegible]

PRODUCT BACKLOG

Story ID	Story name	Status	Size	Sprint	Priorit y	Story Type	Comments	Additional Comments
1	Crear nueva opción en la cinta de opciones de Excel	Done	3	1	3	Desarrollo	Se incluyó un (1) punto en la tarea 2 del Sp1. No hubo impacto por el cambio	Sin Impacto
2	Cargar datos en un archivo Excel desde la tabla worklog de base de datos de Jira	Done	8	2	3	Desarrollo	Se realizó una revisión de la estimación y se aumentó de 8 a 21 puntos	La diferencia de esfuerzo entre lo estimado y el esfuerzo realizado se debió al cambio de lenguaje de C# a VB. Este esfuerzo no se encontraba estimado.
3	Cargar datos usando el query MMA	Done	20	2	2	Desarrollo	Se realizó una revisión de la estimación y se aumentó de 20 a 16 puntos	Cambio de esfuerzo al hacer revisión de las tareas a ejecutar durante el desarrollo de la historia teniendo en cuenta el avance realizado en la HU_2
4	Cargar datos filtrados por Usuario	Done	13	3	4	Desarrollo	Se realizó el Sprint según lo esperado en la planeación	Sin Impacto
5	Cargar datos filtrados por usuario y semana	Done	13	3	1	Desarrollo	Se incluyó un (1) punto en la tarea 2 de la HU_5. No hubo impacto por el cambio	Sin Impacto
6	Cargar datos filtrados por proyecto y semana	Done	13	4	4	Desarrollo	Se realizó según la planeación	Sin Impacto
7	Cargar datos filtrados por proyecto y mes	Done	13	4	1	Desarrollo	Se realizó según la planeación	Sin Impacto
8	Actualizar datos desde Excel a Jira	Removed	20			Desarrollo	Se elimina al verificar la no viabilidad de la implementación	

SCRUM – HISTORIAS DE USUARIO

- Las metodologías ágiles como **Scrum** utilizan las historias de usuario como el instrumento principal para identificar los requerimientos de usuario.

Las historias de usuario son descripciones cortas y simples de una funcionalidad, escritas desde la perspectiva de la persona que necesita una nueva capacidad de un sistema, por lo general el usuario, área de negocio o cliente.

SCRUM – HISTORIAS DE USUARIO

- Típicamente las historias siguen una plantilla simple: **Yo como un [Rol], necesito [Descripción de la funcionalidad], con la finalidad de [Descripción de la consecuencia].**

SCRUM – HISTORIAS DE USUARIO

- **Rol:** Es el rol que está desempeñando el usuario cuando utiliza la funcionalidad que se está describiendo. Debe ser lo más específico posible, describiendo el rol o actor que se está desempeñando. El enunciado puede escribirse como se sigue: Yo como un [Rol], Desempeñando el rol de [Rol], Como un [Rol], entre otros. Por ejemplo:
 - Yo como cliente registrado.
 - Desempeñando el rol de cliente registrado.
 - Como un cliente registrado.

SCRUM – HISTORIAS DE USUARIO

- **Característica / Funcionalidad (Feature):** Representa la función que el rol quiere o necesita hacer en el sistema que se está desarrollando. Puede diferenciarse entre acciones obligatorias u opcionales, utilizando la palabra puede o necesita para describir la acción. Por ejemplo:
 - Necesito realizar búsquedas de productos por categorías.
 - Puedo seleccionar una categoría para ver el número de productos que tiene asociado.

SCRUM – HISTORIAS DE USUARIO

- **Razón / Resultados:** Lo que el rol necesita lograr al ejecutar la acción. Este es el resultado de ejecutar la acción desde el punto de vista del rol. Este punto puede ser opcional, pues la historia puede documentarse sólo con la definición del rol y la acción (sin definir la consecuencia).

HISTORIA - EJEMPLO

Historias de usuario	Tareas
Como un usuario de la app, quiero poder crear transferencias periódicas, de forma que pueda programarlas mensual o anualmente	Codificar el...
	Diseñar el...
	Reunirse con el dueño del producto para...
	Diseñar la interfaz gráfica
	Crear los tests automatizados
	Codificar los otros...
Como usuario de la app, quiero poder contratar fondos de inversión, de forma que pueda invertir mi dinero en ellos	Modificar vistas para...
	Actualizar los tests...
	Codificar el...
	Ampliar el diseño de la base de datos...

HISTORIA EJEMPLO

Historia de Usuario	
Número: 4	Nombre: Calcular recurso.
Usuario: Especialista	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 4.4
Riesgo en Desarrollo: Alto	Puntos Reales: 4.8
Descripción: Se calculará el recurso hídrico subterráneo disponible mediante la utilización del método de balance de la ventana de trabajo activa a partir de las mediciones insertadas en la rejilla de datos, de ser necesario algún valor adicional o existir variantes para el cálculo se indicará al usuario que introduzca la información requerida. En el caso del Método de Balance, para algunas variantes del cálculo del recurso se necesitará el valor de Coeficiente de Almacenamiento del período analizado, este se podrá indicar mediante la inserción directa de su valor por pruebas de bombeo o determinarlo mediante fórmulas de balance. Los resultados del balance serán mostrados en un reporte con opciones de impresión y exportación a múltiples formatos.	
Observaciones: El cálculo de recurso hídrico se podrá realizar desde las ventanas de trabajo de los métodos de balance al pulsar sobre la función Balance de la barra de herramientas.	

SCRUM – SPRINT BACKLOG

- Lista de tareas que realizará el equipo durante el sprint para generar el incremento previsto.
- El equipo asume el compromiso de la ejecución. Las tareas están asignadas a personas, y tienen estimados el tiempo y los recursos necesarios.
- **Incremento:** es la suma de todos los elementos de la Lista de Producto completados durante un Sprint y el valor de los **incrementos** de todos los Sprints anteriores. Utilizable, potencialmente entregable.

SPRINT BACKLOG

- El sprint backlog es la lista que descompone las funcionalidades del product backlog en las tareas necesarias para construir un incremento: una parte completa y operativa del producto.
- Lo que se debe terminar al final de un sprint son historias de usuario completas, no tareas.
- En el sprint backlog se asigna a cada tarea la persona que la va a llevar a cabo, y se indica el tiempo de trabajo que se estima, aún falta para terminarla.

SPRINT BACKLOG

- Realizado de forma conjunta por todos los miembros del equipo.
- Cubre todas las tareas identificadas por el equipo para conseguir el objetivo del sprint, incluso bugs y testing.
- Sólo el equipo lo puede modificar durante el sprint.
- El tamaño de cada tarea está en un rango de 4 a 16 horas de trabajo.
- Es visible para todo el equipo. Idealmente en una pizarra o pared en el mismo espacio físico donde trabaja el equipo.

SPRINT BACKLOG

[illegible]

SPRINT BACKLOG

[illegible]

SPRINT – TAREAS (TASK SLIPS)

2		Cargar datos en un archivo Excel desde la tabla worklog de base de datos de Jira	
Story ID:		Story:	
Task:	Crear conexión a base de datos local		
Responsible Person: Oscar López			
Initial Estimate	2	Work Done	Work Left

2		Cargar datos en un archivo Excel desde la tabla worklog de base de datos	
Story ID:		Story:	de Jira
Task:	Crear consulta a la tabla Jira.Worklog		
Responsible Person: Oscar López			
Initial Estimate	1	Work Done	Work Left

2		Cargar datos en un archivo Excel desde la tabla worklog de base de datos de Jira	
Story ID:		Story:	
Task:	Crear Prueba TDD de extracción de datos		
Responsible Person: Oscar López			
Initial Estimate	2	Work Done	Work Left

ESTIMACIÓN DE ESFUERZO

- Una de las técnicas más efectivas y conocidas del mundo ágil para estimar es el Planning Poker (cartas numeradas). Se trata de una dinámica ágil en la que se reúne el equipo con una baraja de Poker modificada y se hacen rondas de estimación con ayuda de estas cartas.
- Para una sesión de estimación ágil será necesario que cada participante tenga una baraja de Planning Poker.

PLANNING POKER

- En cada baraja hay una pseudo-secuencia de Fibonacci. Es recomendable quedarse con solo las primeras cartas para evitar estimar tareas demasiado grandes. Así cada participante tendrá las siguientes cartas: 0, 1, 2, 3, 5, e infinito. El cero significa que la historia ya está hecha o no requiere ningún esfuerzo, el interrogante significa que nos falta información para estimar esa historia o tarea y el infinito es que es demasiado grande y hay que dividirla.

PLANNING POKER

Antes de iniciar la sesión hay que tener claro en qué unidades se va a estimar. Es posible que para estimaciones alto nivel se prefiera estimar en semanas ideales, mientras que para historias más pequeñas y tareas de bajo nivel prefiera estimar en días/horas ideales.

Un día ideal es el trabajo que consigue una persona en un día en el que no tiene interrupciones de ningún tipo y todo le sale a la primera. También es posible que se decida estimar en una medida relativa como story points. En cualquier caso, lo importante es que todos sepan en qué medida se estima.

PLANNING POKER

- También será importante que todos sepan qué se incluye y qué no en la estimación, si se incluye alguna documentación, tests unitarios, tests de integración y cualquier otra cosa que forme parte del desarrollo.
- Es recomendable incluirlo todo en la estimación de las historias y separar solo la documentación al desglosarlo en tareas.

PLANNING POKER - DINÁMICA

- Todo el equipo “se encierra” para estimar, y todos conocen lo que se va a estimar. Si hay gente que no está al tanto de lo que se va a estimar la sesión debe comenzar con una explicación y una sesión de preguntas para despejar cualquier duda sobre las historias que se van a estimar.

PLANNING POKER - DINÁMICA

- Una por una se leen (dueño del producto) y discuten las historias de usuario. Una vez todos tienen claro en qué consiste cada uno elige una carta en función del esfuerzo que prevé requerirá esa historia. No es posible seleccionar un valor no incluido en la baraja. Solo estiman los que después desarrollan (ni el Scrum Master ni el Product Owner estiman, solo resuelven dudas).

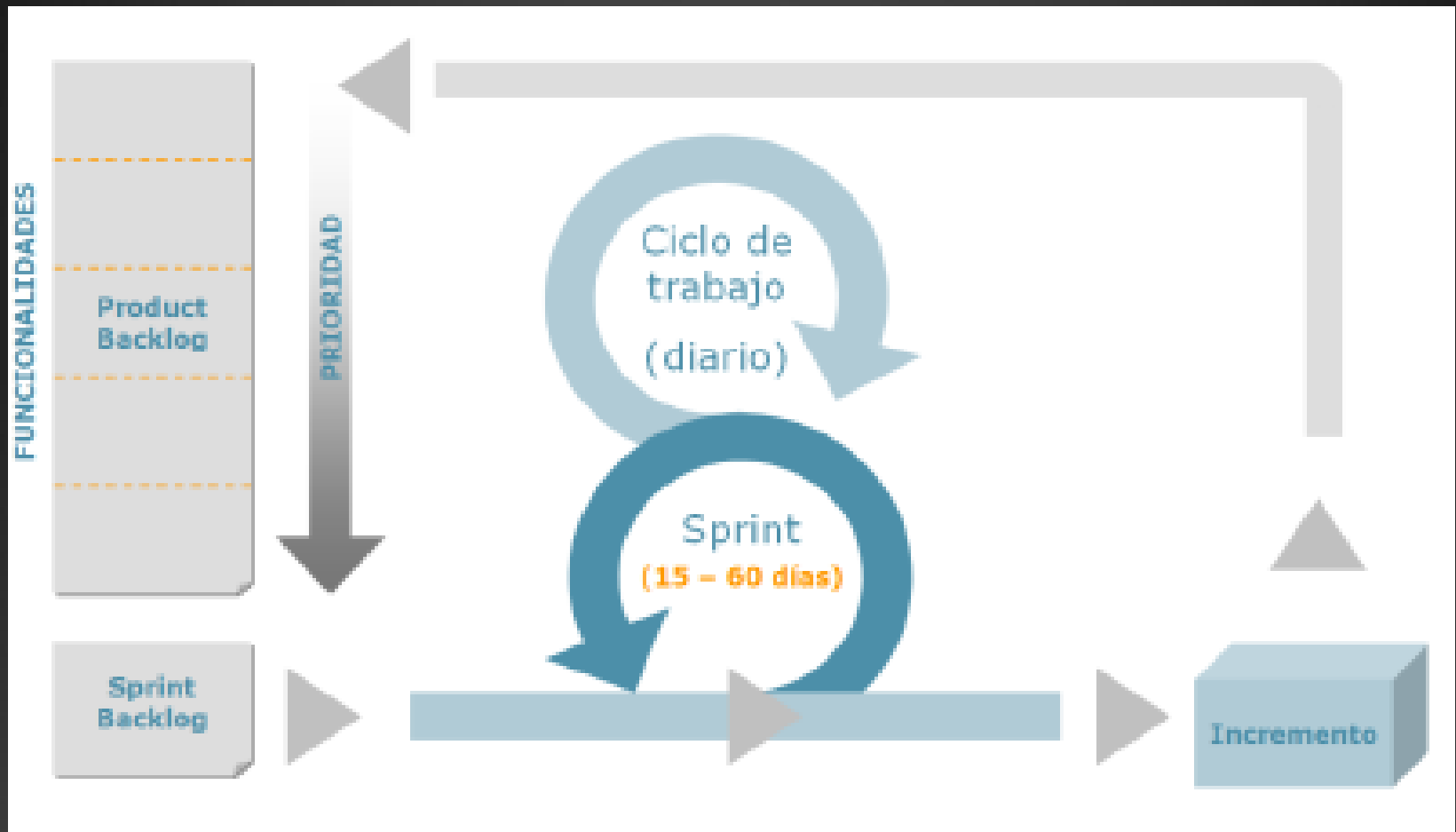
PLANNING POKER - DINÁMICA

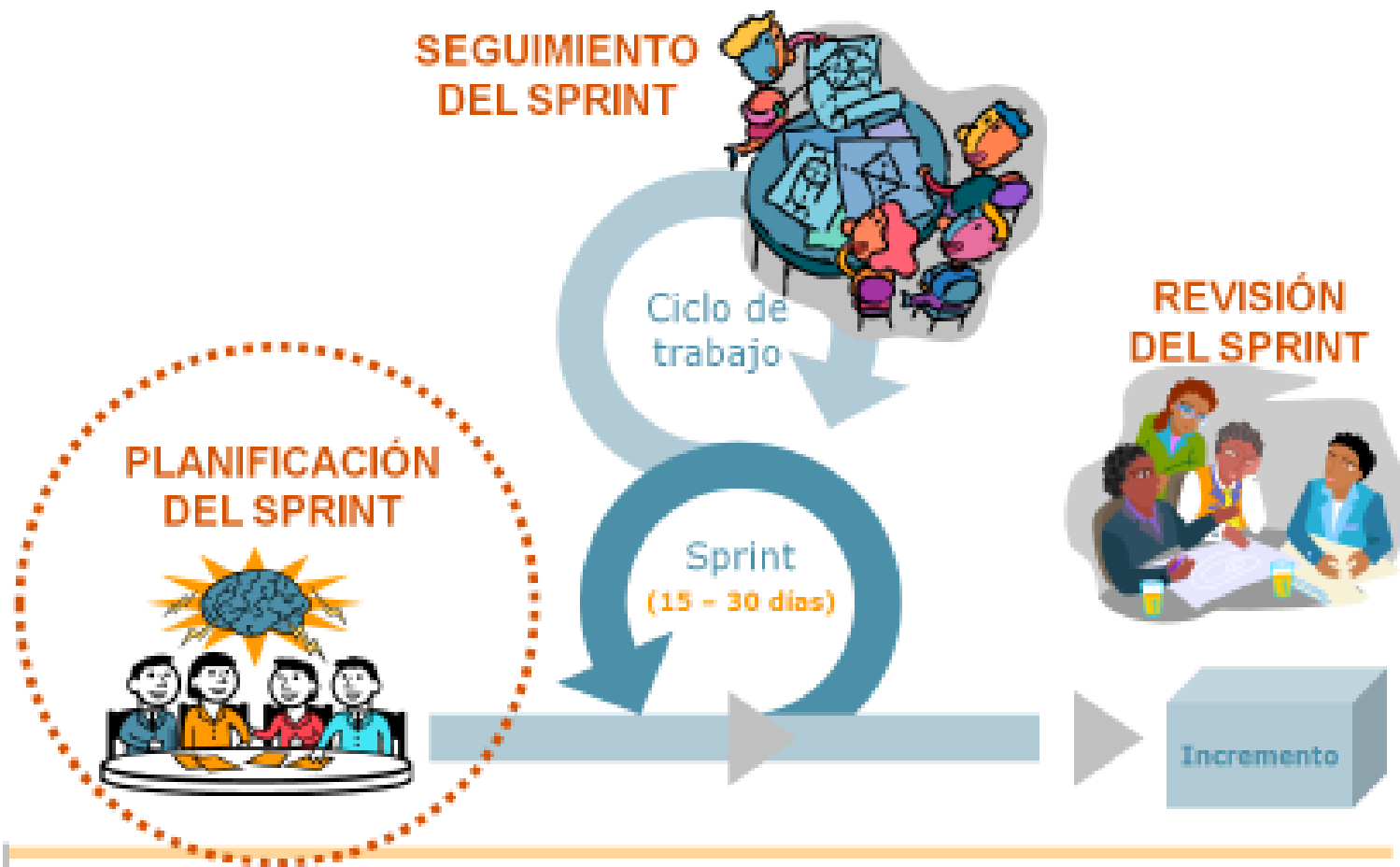
- Cada miembro del equipo selecciona una carta, la pone boca abajo en la mesa y cuando todos han seleccionado una carta se ponen boca arriba a la vez
- Una vez todos han mostrado su carta (su estimación), hay dos opciones:
 - Descartar las estimaciones mínimas y máximas y quedarse con la estimación media más repetida.
 - Buscar la unanimidad. Los que han diferido de la mayoría explican sus motivos y el resto explica los suyos. El objetivo es llegar a una estimación unánime.

PLANNING POKER - DINÁMICA

- Si no hay consenso (lo normal con más de 2 participantes) se abre la discusión.
- Se repite la estimación nuevamente en busca de consenso. Si no se consigue a la segunda se vuelve a discutir.
- A la tercera si no hay consenso se escoge o bien la media o bien el máximo.

SCRUM PROCESO





PLANIFICACIÓN DEL SPRINT

Reunión que consiste en 2 fases:

- En la primera, que puede tener una duración de una a cuatro horas, se decide qué elementos del product backlog se van a desarrollar.
- En la segunda se desglosan éstos para determinar las tareas necesarias, estimar el esfuerzo que necesita cada una y asignarlas a las personas del equipo.
- La planificación del sprint no debe durar más de un día

SEGUIMIENTO DEL SPRINT

- Reunión diaria breve, de no más de 15 minutos en la que todos los miembros del equipo dicen las tareas en las que están trabajando, si se han encontrado o prevén encontrarse con algún impedimento y actualizan sobre el sprint backlog las tareas ya terminadas o los tiempos de trabajo que les quedan.

REVISIÓN DEL SPRINT

- Reunión realizada al final del sprint en la que, con una duración máxima de 4 horas el equipo presenta al propietario del producto, clientes, usuarios, gestores... el incremento construido en el sprint.

GRÁFICO BURN DOWN

- Es un gráfico cartesiano que representa en el eje x los días laborables del sprint, y en el y la cantidad de esfuerzo estimada.
- En la reunión diaria cada miembro del equipo al referirse al trabajo que realizó ayer, y el que tiene previsto hacer hoy, actualiza en el sprint backlog si ha terminado alguna de las tareas en las que ha trabajado, o cuánto esfuerzo estima que les quedan.

GRÁFICO BURN DOWN

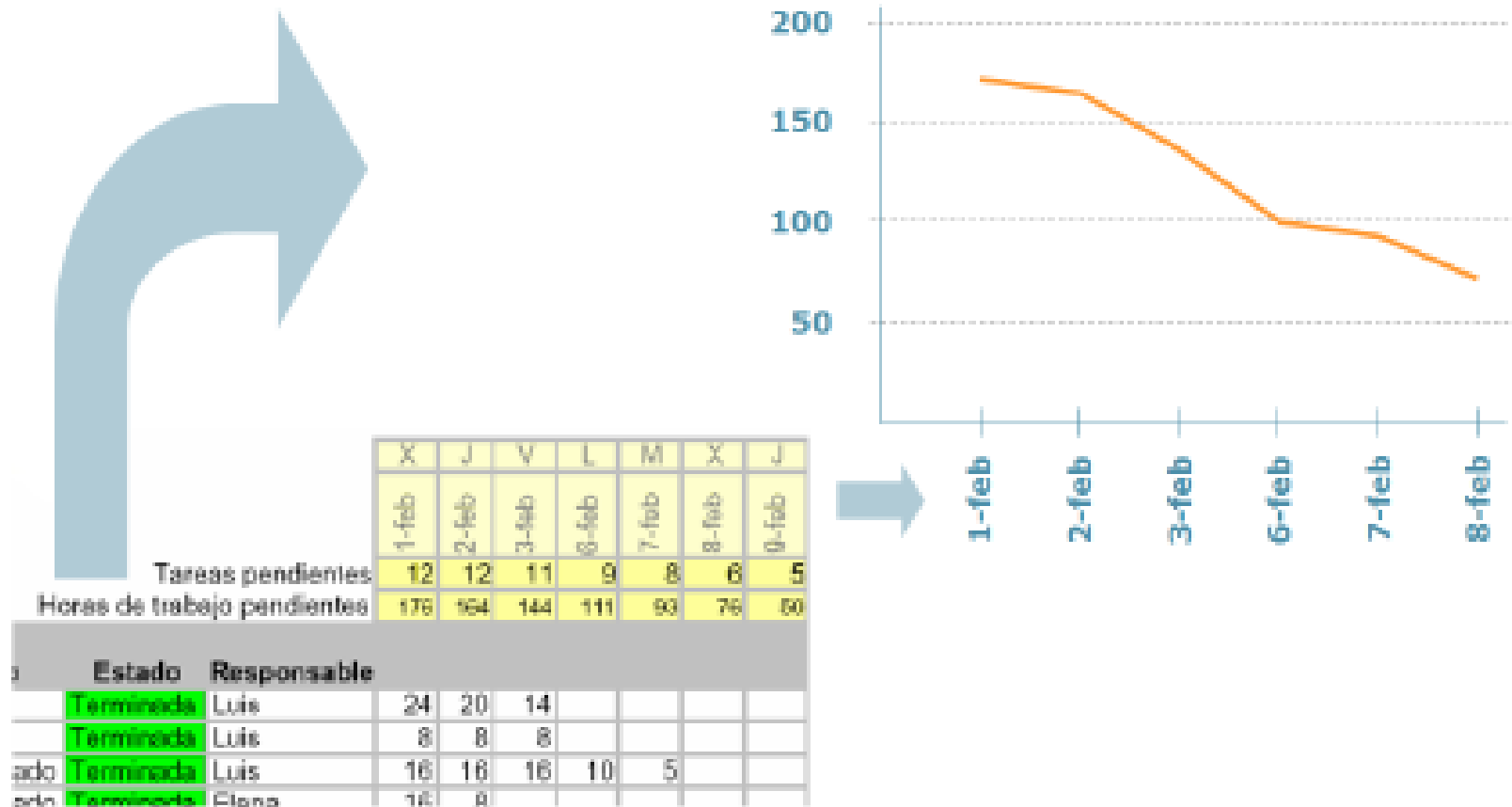
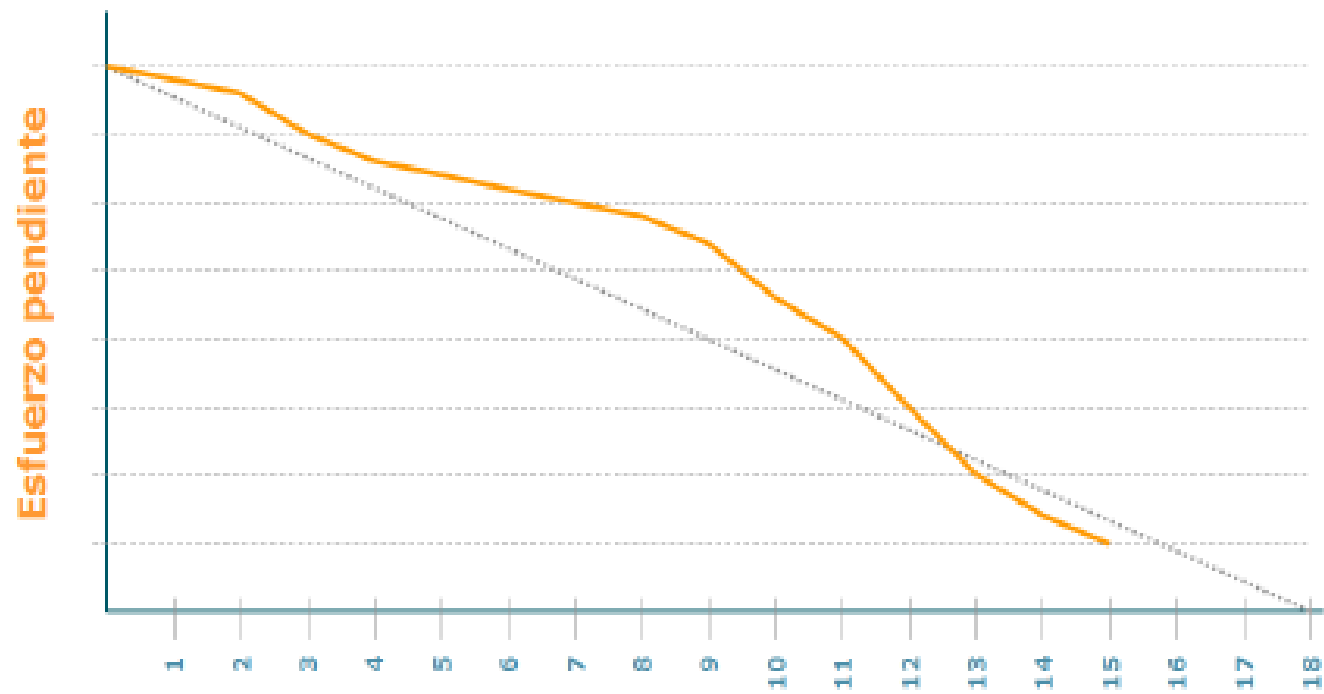


GRÁFICO BURN DOWN

- Al final de la reunión la columna del día del sprint backlog muestra el esfuerzo que según el equipo falta para terminar el sprint, y el equipo marca en el gráfico el punto que tiene como ordenada ese valor, y como abscisa la fecha del día.
- La evolución ideal del sprint se representaría por la línea punteada en gris de la figura. La línea naranja muestra la evolución real diaria.
- El recorrido sobre la diagonal es síntoma de problemas o sub-estimación del sprint backlog. El recorrido bajo la diagonal es síntoma de sobre-estimación del backlog.

GRÁFICO BURN DOWN

Indicador de progreso del sprint



SEQUENCE

Sprint	Remain.Work	Planned Work	Realized Work
1	99	3	4
2	95	28	37
3	58	26	27
4	31	26	26
5			
6			
7			
8			
9			
10			

ROLES – PRODUCT OWNER

El Product Owner es el responsable de optimizar y maximizar el valor del producto. Es quien decide qué trabajo hay que hacer y también es quien debe encargarse de la interlocución con los stakeholders. Es, en definitiva, la persona que se focaliza en la parte de negocio y es responsable del ROI del proyecto.

Sus tareas se podrían resumir en 3 funciones clave:

- **Gestiona prioridades:** contrata al equipo de desarrollo, gestiona los presupuestos y explica qué valor produce el producto en el que se está invirtiendo.
- **Representante del negocio:** El Product Owner debe tener la capacidad de tomar decisiones por sí mismo. Al formar parte del negocio, podrá aportar valor a su trabajo y al producto.
- **Intraemprendedor:** Debe ser un Product Manager ágil, que sea capaz de medir el valor generado y utilizar la flexibilidad de cada sprint para aumentar el valor.

Es fundamental que el Product Owner tenga el poder necesario para poder tomar cualquier decisión que afecte al producto sin tener que consultarlas previamente con otra persona. Además, también debe ser el interlocutor con el cliente, y quien transmite sus demandas y su feedback.

ROLES – SCRUM MASTER

El Scrum Master es la persona que lidera al equipo con el objetivo de que entiendan exactamente qué es la metodología Scrum y puedan cumplir las reglas y usar los procesos de la metodología de la mejor manera posible. Se encarga de las labores de mentoring, coaching y formación, y de facilitar reuniones y eventos si fuera necesario, y traslada la visión general del proyecto al equipo. Pero no solo esto, sino que también trabaja junto al Product Owner en la parte de negocio que busca maximizar el ROI. En este caso las funciones del Scrum Master se podrían resumir en dos grandes tareas:

1. **Gestionar el proceso Scrum:** es el responsable de asegurar que la metodología se utiliza correctamente, y debe facilitar la ejecución de todo el proceso. Es la persona que debe velar para que la metodología sea una fuente de generación de valor para la organización.
2. **Eliminar impedimentos:** el Scrum Master también debe ayudar a eliminar los impedimentos que pueden surgir en una organización. El Scrum Master debe transmitir los beneficios de Scrum a la compañía, para que así se facilite su implementación.

Un mismo Scrum Master puede compartirse entre varios equipos, pero es importante tener en cuenta que su disponibilidad puede afectar al resultado final del proceso Scrum.

ROLES – SCRUM TEAM

El Scrum Team es el grupo de personas que tienen los conocimientos técnicos necesarios para desarrollar un proyecto de forma conjunta. Su trabajo se realiza a base de sprints, que son acciones concretas a realizar en periodos de tiempo pequeño, de una a cuatro semanas aproximadamente, con preferencia por los intervalos más cortos posibles. En cada uno de esos sprints el equipo construye y entrega un incremento o una mejora del producto.

Los equipos suelen estar formados por entre 3 y 9 personas que se autoorganizan para desarrollar y mejorar un producto. Para obtener los mejores resultados es importante que todos los miembros del Scrum Team conozcan sus roles internos. Pero para que la metodología Scrum tenga éxito es imprescindible que el equipo de desarrollo tenga la capacidad de autogestionarse a nivel interno, rindiendo cuentas como un solo equipo y sin que haya intervención en sus dinámicas.

STAKEHOLDERS

Los stakeholders son todas las personas u organizaciones que tienen una relación directa o indirecta con el proyecto, y que pueden verse beneficiados o perjudicados en cualquier momento. Los podríamos separar en dos grandes grupos, stakeholders primarios y secundarios:

Los stakeholders primarios son aquellos que tienen una relación directa con la empresa y afectan a sus procesos. Pueden ser clientes, proveedores, accionistas, socios corporativos o los mismos empleados.

Los stakeholders secundarios tienen una vinculación indirecta con la empresa y no afectan a sus procesos, pero pueden verse igualmente afectados por las acciones que tome la compañía. En este caso hablamos de medios de comunicación, empresas de la competencia, gobiernos, instituciones financieras o la comunidad, entre otros.

Todos los roles de Scrum son imprescindibles para crear el producto del proyecto, y son los responsables del éxito de cada sprint y del proyecto en su totalidad. Por eso el primer paso tras asignar las funciones de cada persona es definir qué debe hacer cada integrante del equipo. Que todos los involucrados entiendan cómo el alcance de su rol ayudará a que se cumplan eficientemente los objetivos marcados.