



Universidad
Nacional
de San Martín

Proyectos de Software

Métricas

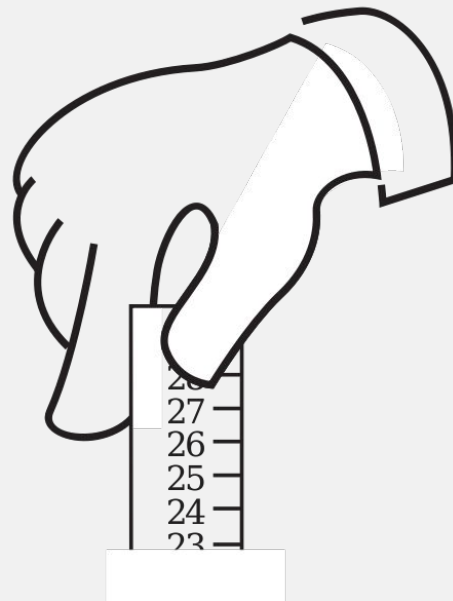
Lic. Pablo Nuñez Monzon

2024



¿Para qué me sirven?

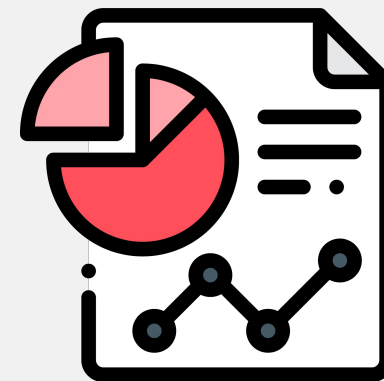
- Mejorar de la toma de decisiones.
- Optimizar el proceso de desarrollo.
- Aumentar de la transparencia y la comunicación.



¿Para qué me sirven?

Las métricas sirven para medir el avance. Proveen visibilidad, del estado de nuestro producto y del avance del trabajo realizado.

Queremos entender cuánto esfuerzo se hizo, cuánto falta para “terminar” y cuánto estamos avanzando por unidad de tiempo.

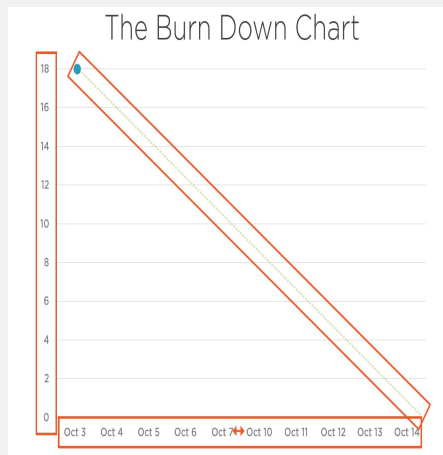


Responden a:

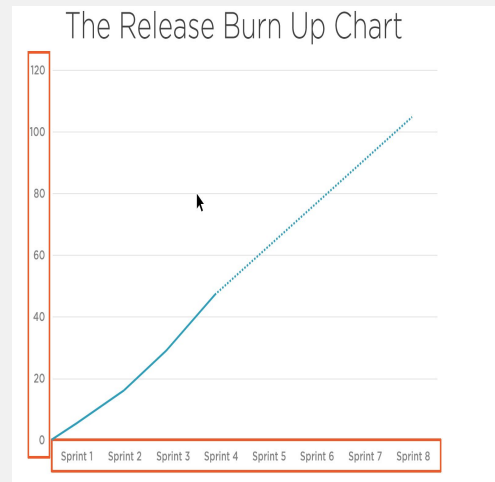
- ¿Cuál es la velocidad de desarrollo del equipo?
- ¿Cuánto trabajo avanzamos por tiempo?
- ¿Cómo planificamos el trabajo para adelante?
- ¿Cuál es nuestra capacidad productiva como equipo?
- ¿Cuánto tiempo nos lleva entregar valor al producto?
- ¿Para cuando estimamos el próximo release?

¿Cuáles son?

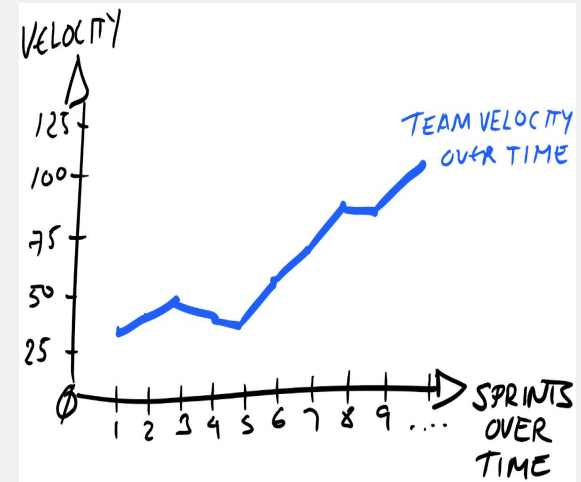
1. Burn down



2. Burn up

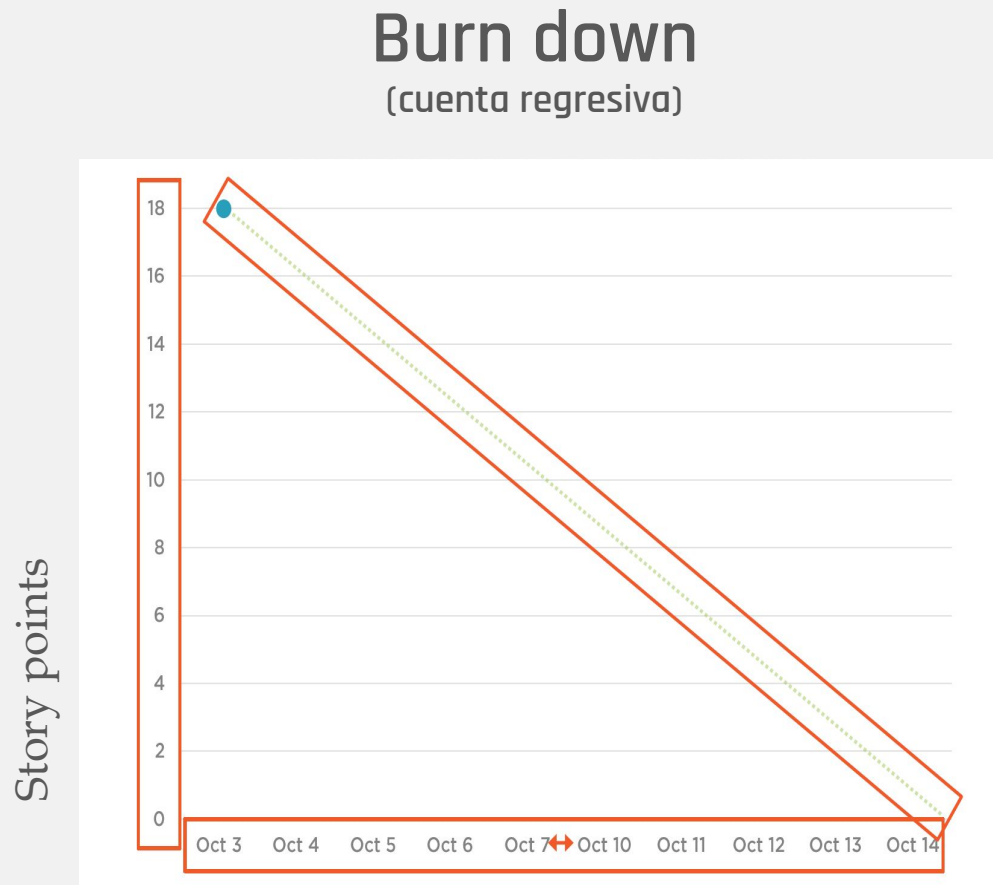


3. Velocity



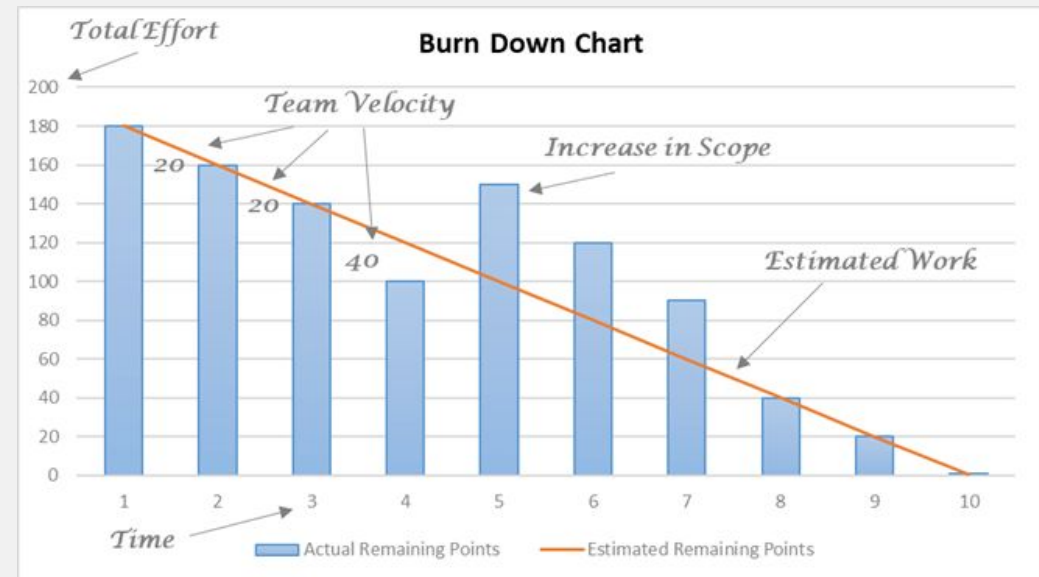
1. BurnDown chart

- Muestra los user story points (sp) pendientes por realizar en función de los días del sprint.
- Permite ver que % del trabajo planificado que realizamos y cuánto nos queda.
- Brinda gráficamente la comparación entre lo planificado y lo real, para detectar desvíos y retrasos.



¿Cómo se hace?

- Cada día, se actualiza en el gráfico cuántos SP del sprint quedan por realizar.
- Es importante actualizar el estado de las tareas.
- Muestra el avance de trabajo del sprint (sprint backlog) en función de los story points y el tiempo (días).
- Muestra el trabajo pendiente dentro del sprint.
- Refleja la realidad y el estado del sprint del proyecto.



Eje X: Días del sprint.

Eje Y: Esfuerzo pendiente (sp).

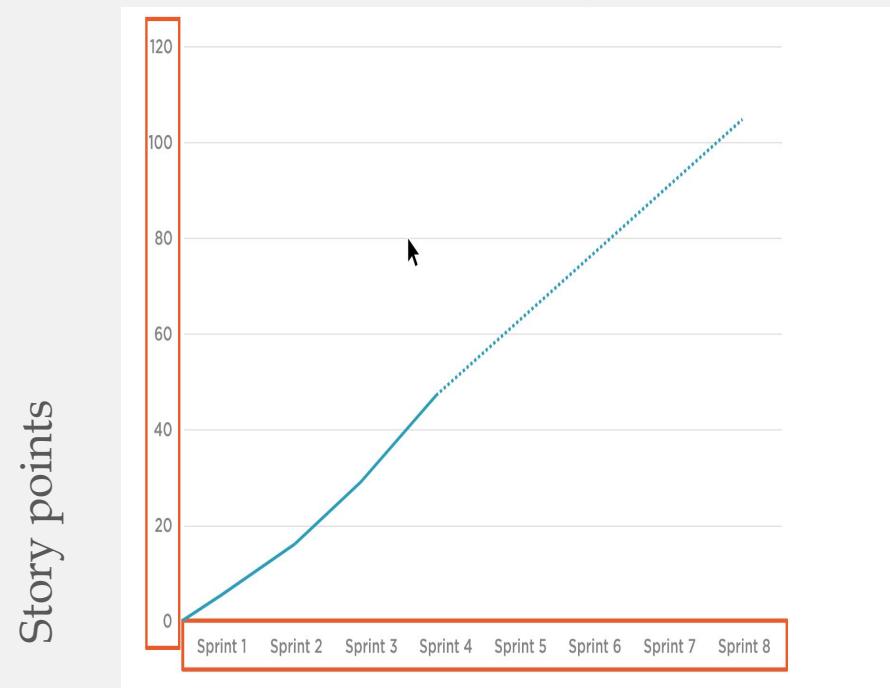
Línea 1: Progreso ideal (sp remanentes).

Línea 2 (barras): Progreso real (sp remanentes).

2. BurnUp chart

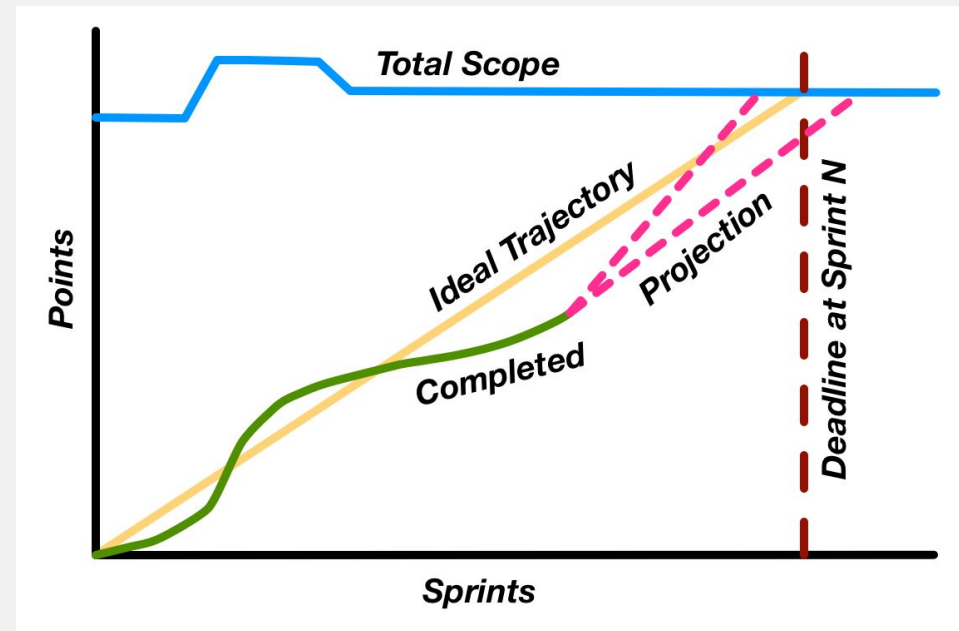
- El burn-up chart (literalmente «gráfico de quemado hacia arriba») muestra el trabajo aportado por un equipo.
- Permite tomar decisiones realistas acerca del trabajo pendiente y las expectativas alrededor del mismo.
- Usamos el burn-up chart para ayudar al equipo a encontrar su ritmo sostenible.

The Release Burn Up Chart



¿Cómo se hace?

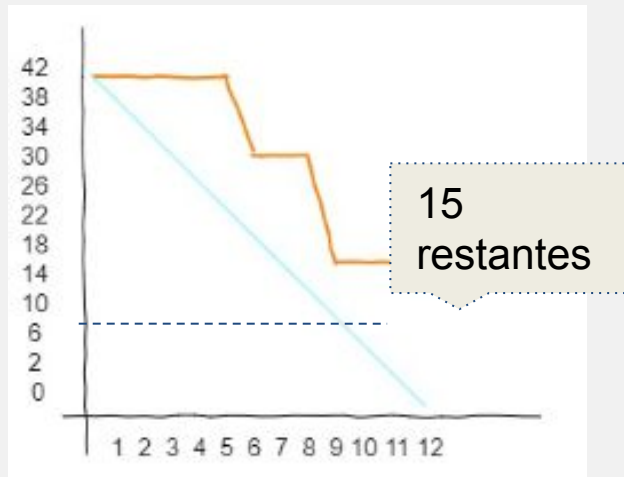
- Mostramos en el eje horizontal la duración del proyecto, medido en sprints (todos de la misma duración), y los puntos en los que hemos estimado las historias de nuestro product backlog en el eje vertical.
- Vamos trazando, iteración tras iteración, una línea que irá uniando los puntos entregados.
- Las 2 líneas del gráfico mas importantes representan: esfuerzo del equipo o línea de trabajo completado (verde) y línea ideal de trayectoria de avance (amarillo).



Tiempo

3. Velocity

Mide la velocidad del equipo. Es la cantidad de trabajo que un equipo puede satisfactoriamente completar en un sprint.



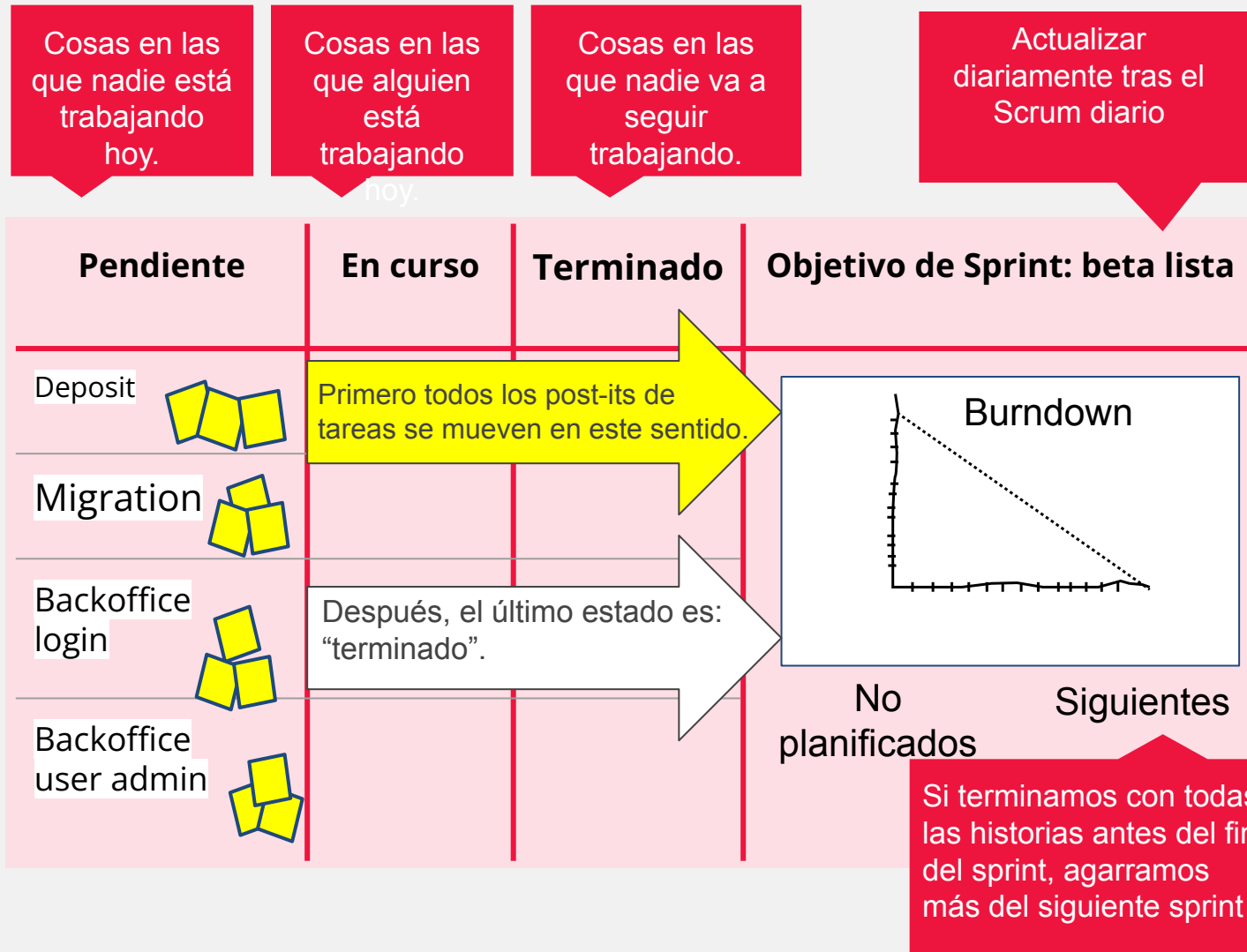
$$\text{Velocity} = 42 - 15 = 37$$

- Basado en datos empíricos sobre lo que el equipo completó en sprints anteriores.
- Provee una base para estimar la cantidad de trabajo que se podrá completar en los próximos sprints (base y promedio).
- Se puede ajustar según las variables que afectan al equipo en cada sprint (bajas de equipo, vacaciones, feriados).

ejemplo práctico!

Día 0 : Comienzo

Sprint: 15 días
Día 1 a día 15 de agosto
(solo 10 días hábiles)
Story points: 70



User stories:
5 x 13p
1 x 5p

HISTORIA DE USUARIO
13p

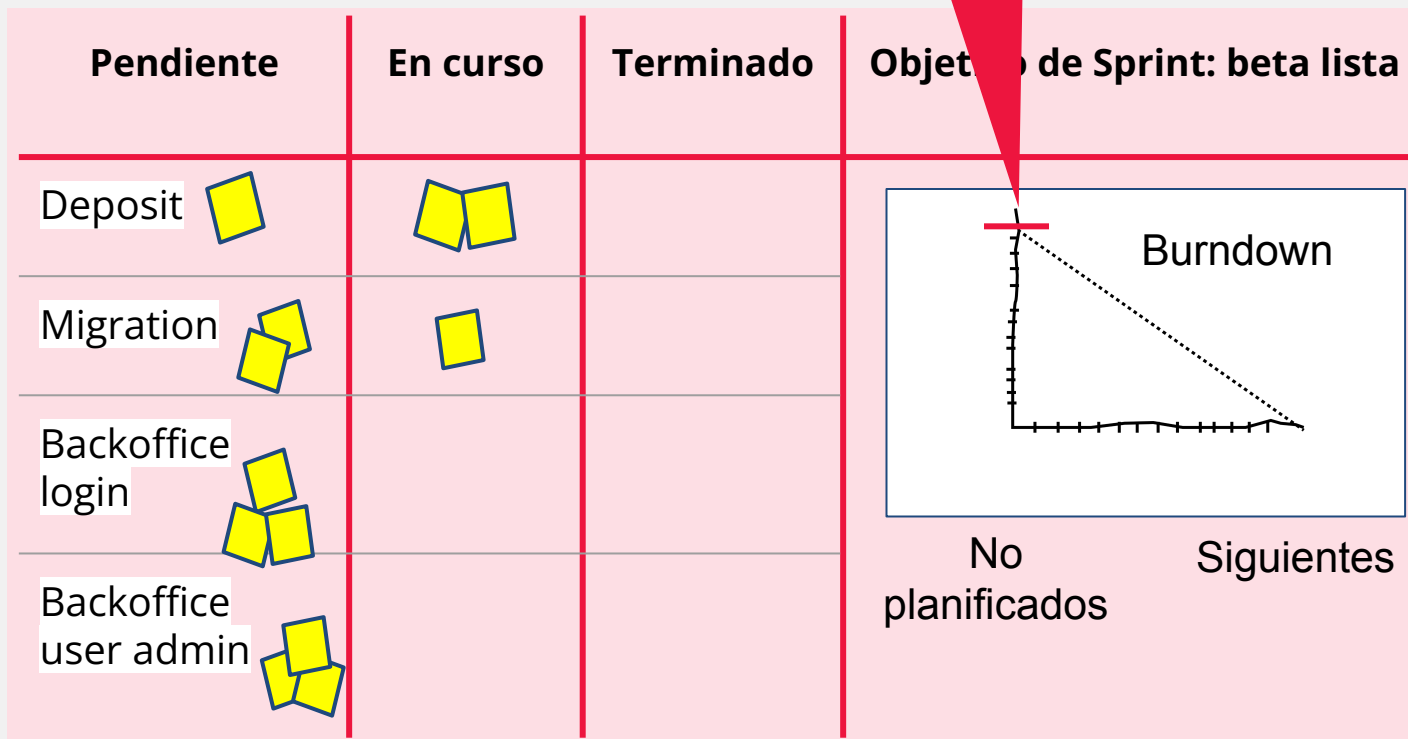
Tarea 8h

Tarea 8h

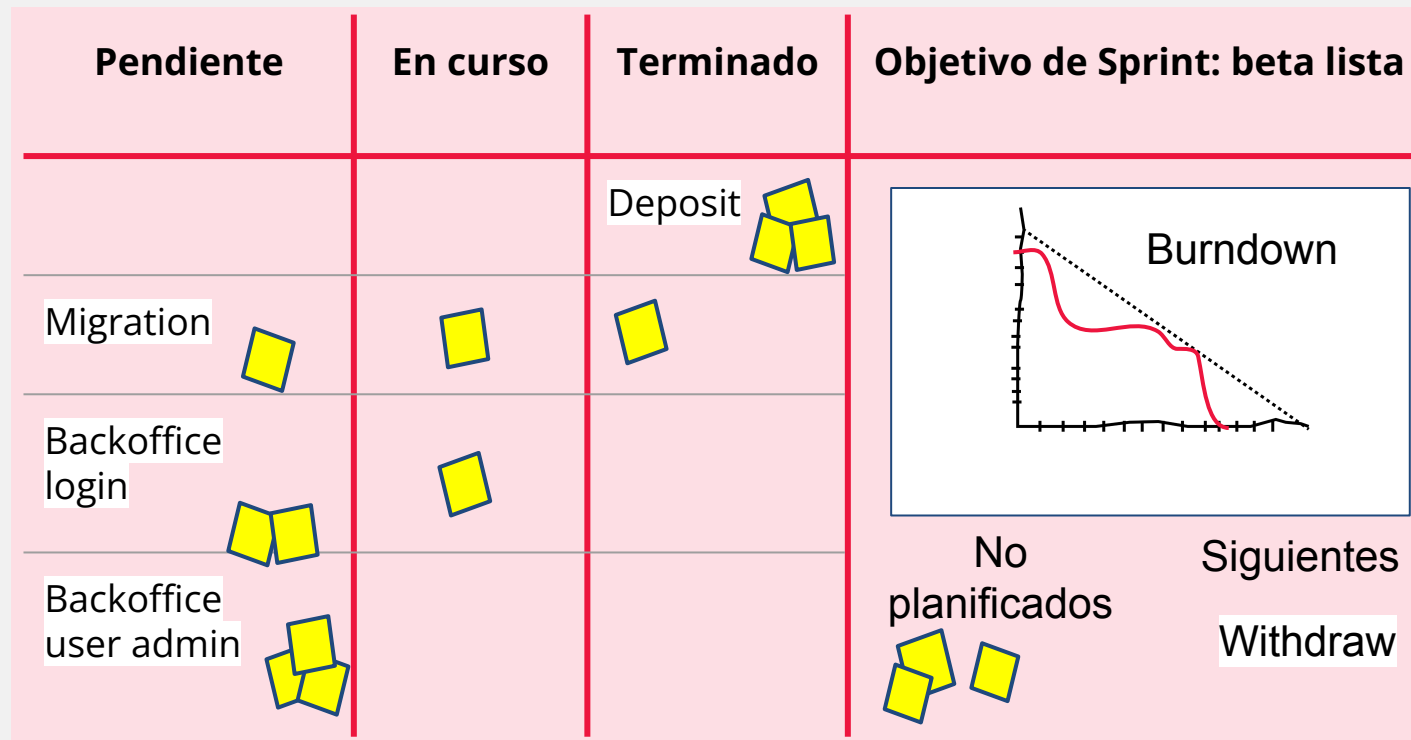
Tarea 8h

Día 1

Aún sin historias terminadas



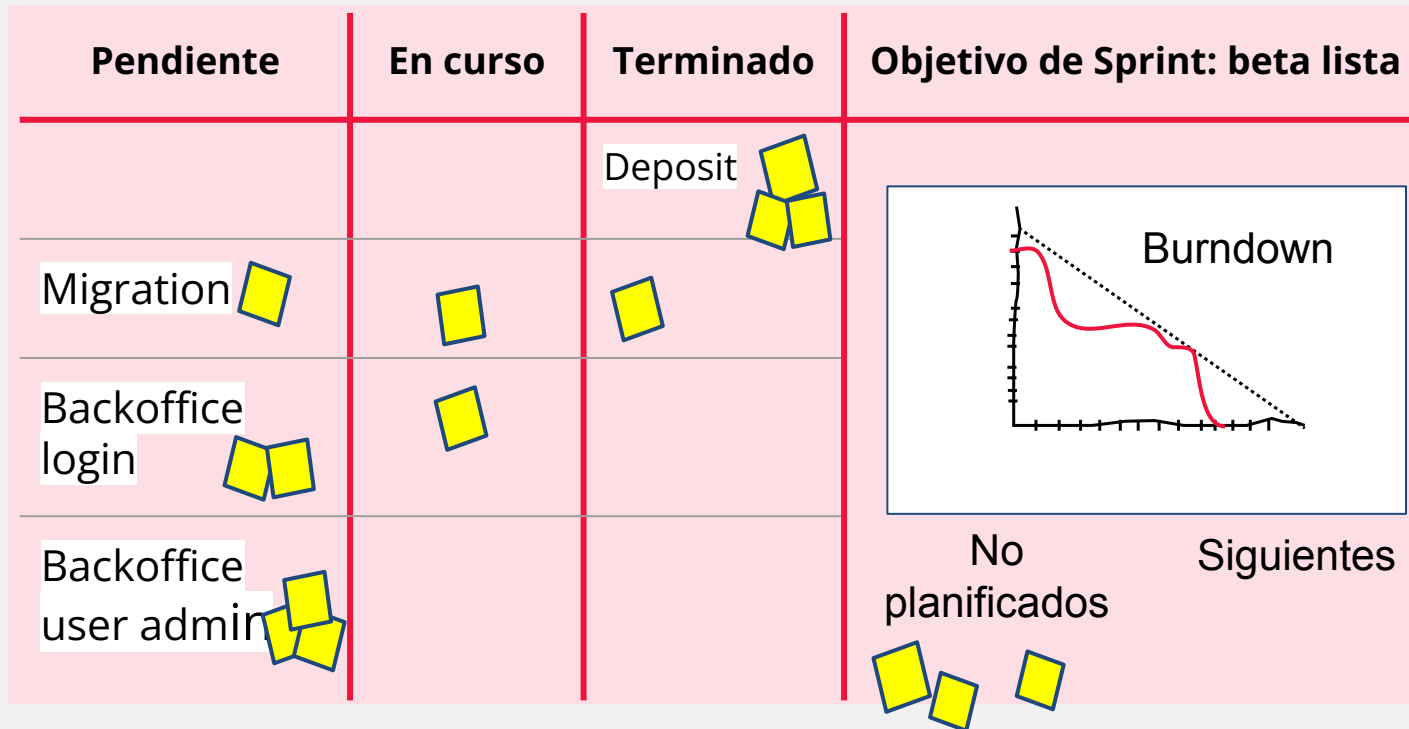
A medida que van pasando los días...



En la daily, se pueden validar qué tareas u historias están como terminadas y se marcan los puntos restantes en el gráfico, según la fecha.

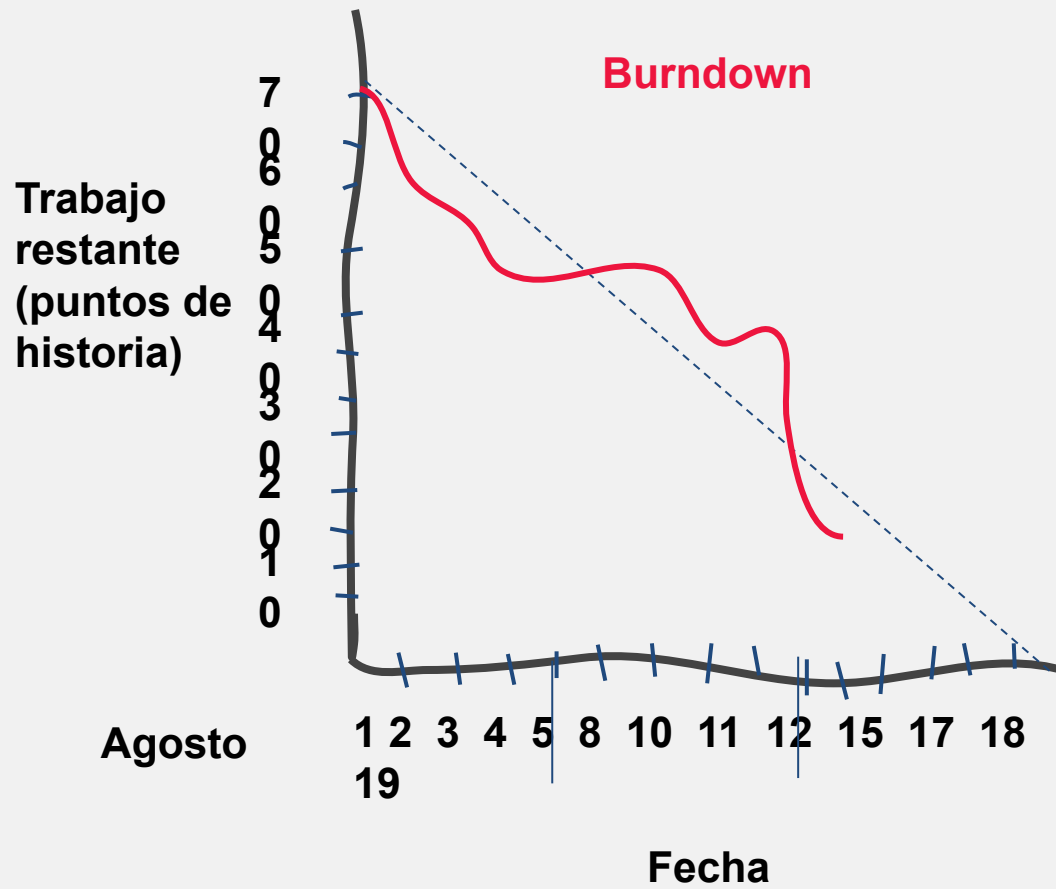
Para cerrar un ticket, verificar si se cumple el “Criterio de aceptación”

Día 12: Fin del sprint



Se completaron:

- 3 user stories de 13p, probadas y funcionando.



Calcular:

¿CUAL ES LA VELOCITY DEL EQUIPO?

¿Qué hacemos con la información?



¡Ooops! Hay que quitar algunos elementos de la pila de sprint.

Puede ser que hayamos estimado menos esfuerzo del que realmente necesitamos.

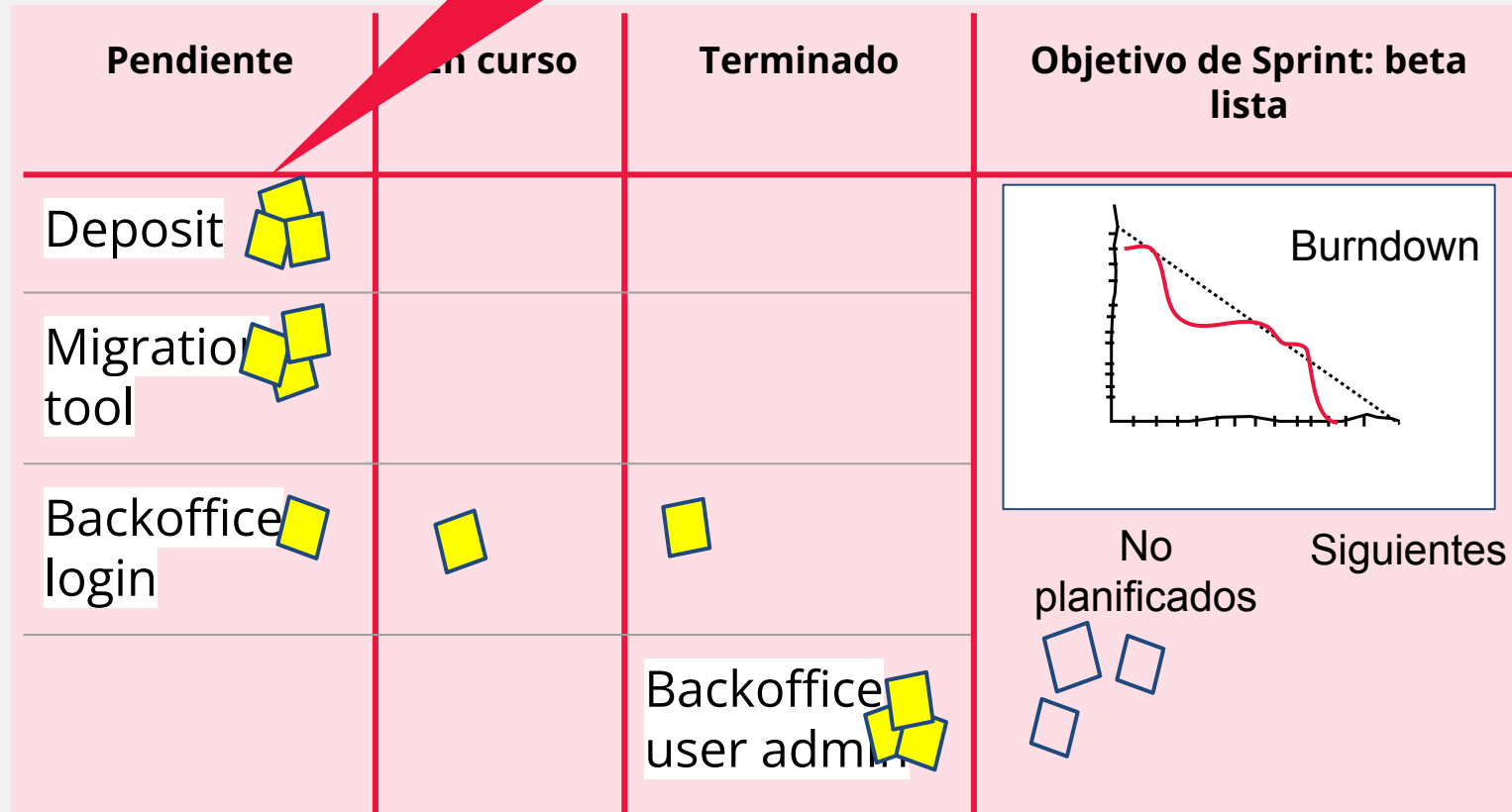


Hay que añadir algunas historias al sprint.

Puede ser que hayamos pensado que era más difícil de lo que realmente fue.

Prioridad

¡Ooops! El equipo está desarrollando historias de baja prioridad e ignorando las de alta.



El orden de las historias refleja su prioridad.

Glosario

- **Historias de usuario** = User stories = Product backlog item
- **Story Points** = Puntos de historia = Medida de complejidad de una tarea o historia
- **Release** = despliegue o lanzamiento de nueva versión del producto
- **Feature** = funcionalidad

Conclusión

- Crear historias de usuario eficientes es un arte.
- El Product Owner juega un rol vital al desarrollar y mantener las historias de usuario claras y gestionables.
- Una historia de usuario raramente es efectiva sin un Product Owner dedicado y hábil.