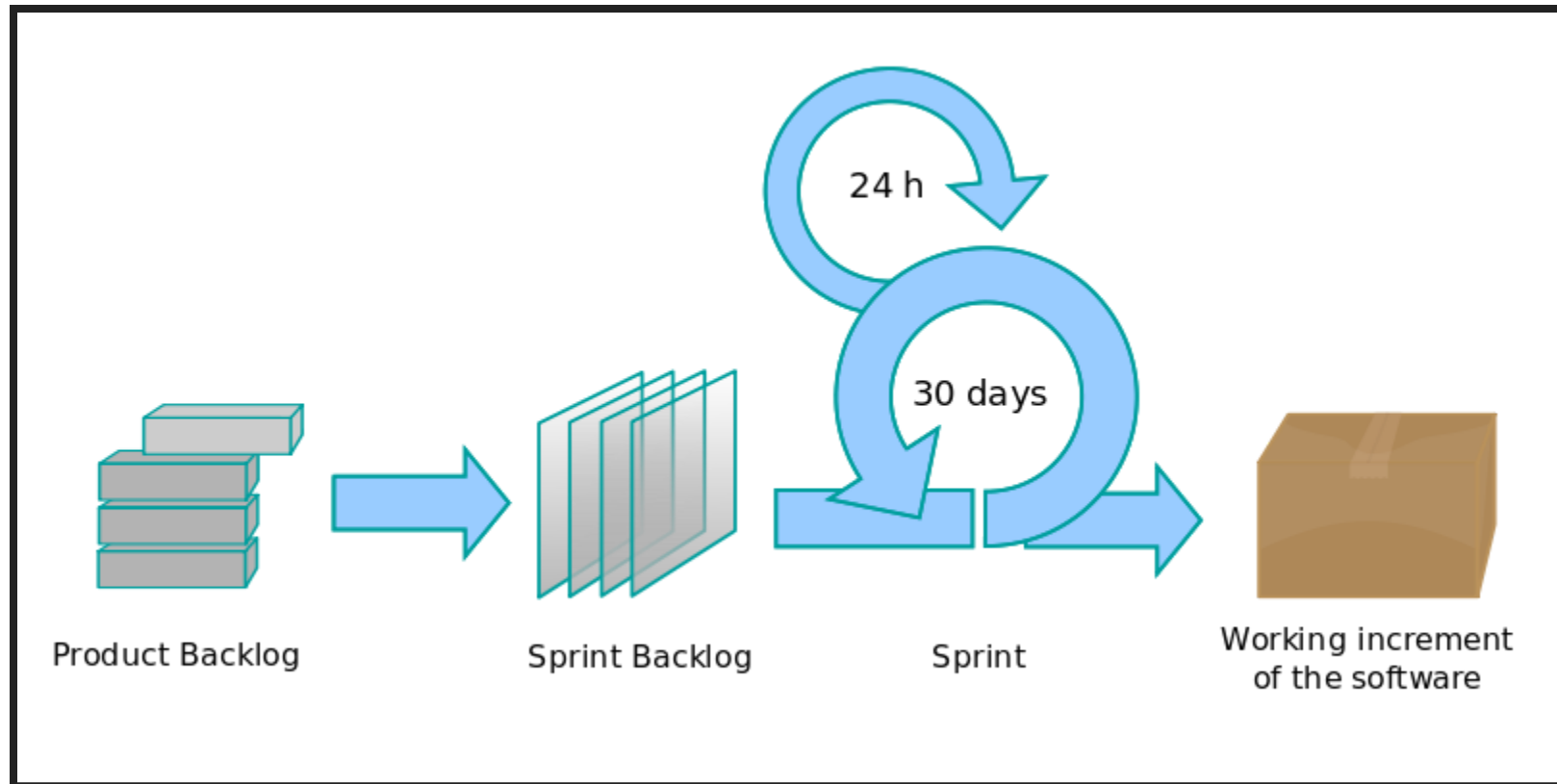


scrum



fuelle: [Wikipedia](https://en.wikipedia.org/wiki/Scrum)

SCRUM

un framework de gestión de proyectos basado en las **mejores prácticas** llevadas a cabo por equipos de alto rendimiento

que se basa en **iteraciones incrementales** de producto

que se apoya en **reuniones** sistemáticas a lo largo del proyecto

que produce entregas parciales priorizadas por beneficio aportado al cliente

beneficios

- entrega periódica de resultados (verificación de expectativas)
- entregas parciales (ROI temprano)
- flexibilidad y adaptación respecto a las necesidades del cliente
- mejores estimaciones (las realizan las personas que lo van a desarrollar)

cómo funciona el mundo

¿cómo trabajamos? ¿cómo nos han enseñado a trabajar?

diagramas de Gantt

Un artefacto de la I Guerra Mundial... usado para planificar software

Muchas veces se ajusta la realidad para que se parezca al diagrama

una nueva forma de pensar

menos gente, menos tiempo, menos coste, más valor entregado

el término **Scrum** proviene del rugby: cómo trabaja un equipo en conjunto para mover el balón por el campo

- basado no en cómo se debe trabajar, sino en cómo se trabaja
- en las mejores prácticas de los mejores equipos de empresas de todo el mundo
- se pregunta por qué cuesta tanto entregar valor y estimar esfuerzo
- asume la incertidumbre y adopta la creatividad

una nueva forma de pensar

idea simple: ciclos "inspect & adapt"

- **transparencia:** todos los aspectos del proceso que afectan al resultado son visibles
- **inspección:** se debe controlar con la frecuencia suficiente los diversos aspectos del proceso para que puedan detectarse variaciones inaceptables en el mismo
- **revisión:** el producto debe estar dentro de los límites aceptables; en caso contrario, el proceso debe adaptarse.

una nueva forma de pensar

¿Cuántas veces hemos oído hablar sobre proyectos masivos con costes prohibitivos que se cancelan no sólo debido a sus sobrecostes sino porque simplemente "no funcionan"? ¿Cuánto tiempo de nuestras vidas se desperdicia en trabajo que no aporta valor?

No tiene que ser así. Sólo porque todo el mundo siempre nos haya dicho que esta es la forma en la que funciona el mundo, eso no significa que estén en lo cierto. Hay una forma diferente de hacer las cosas, una forma diferente de trabajar

y si no lo hacemos, seremos externalizados...

equipos

**en el mundo laboral, los equipos
consiguen que el trabajo salga adelante**

sin embargo, todo se centra en la acción individual...

error

equipos

estudios con grupos de alumnos de Yale sobre rendimiento de individuos...

¿Cuál puede ser el ratio de mejora entre el mejor y el peor?

1:10

fuentes: [Scrum: the art of doing twice the work in half the time](#), Jeff Sutherland (2014)

equipos

estudios con muestras de 3800 proyectos distintos de diversas áreas (financiera, tecnológica, etc.) sobre rendimiento de equipos

¿Cuál puede ser el ratio de mejora entre el mejor y el peor?

!!!1:2000!!!

entonces, ¿dónde deberíamos centrar nuestra atención para mejorar?

fuentes: [Scrum: the art of doing twice the work in half the time](#), Jeff Sutherland (2014)

características en equipos

trascendentes: sentido y propósito

autónomos: auto-gestión y auto-organización (decisiones propias)

multifuncionales: disponen de todas las habilidades necesarias para completar el proyecto

Fuente: "[The New New Product Development Game](#)", Takeuchi y Nonaka

tamaño de los equipos

*“añadir más personas a un proyecto retrasado
tan sólo lo retrasa más”*

Fuente: ["The Mythical Man-Month"](#), Fred Brooks

por otra parte, Putnam llegó a la conclusión de que cuando los equipos crecen por encima de **8 miembros**, su rendimiento cae drásticamente

Fuente: ["Familiar Metric Management - Small is Beautiful- Once Again"](#), Putnam & Myers



¿por qué?

estudio de George Miller (1956): el número máximo de elementos que la persona media puede retener en su memoria a corto plazo es 7

investigaciones posteriores demostraron que Miller estaba equivocado: ese número no es 7...

es 4

Fuente: "The Magical Number 4 in Short-Term Memory: A Reconsideration of Mental Storage Capacity", Corwan, Nelson

explicación

la afirmación de Brooks se explica con dos razones:

- el tiempo que lleva hacer que la nueva persona "coja ritmo"
- el número de canales de comunicación crece drásticamente con el número de personas, y sencillamente nuestro cerebro no puede con ello

$$\text{Nº Canales Comunicación} = n * (n-1) / 2$$

no podemos saber qué están haciendo los demás

auto-organizado

dentro de un contexto (reglas) y unas limitaciones (externas)
es necesario un grado de compromiso, motivación y educación
durante el período de adaptación y maduración, es necesario *delegar*
el manager puede fijar las reglas del juego e ir relajándolas
progresivamente

multifuncional

los proyectos se asignan a equipos

(no a gente, ni gente a proyectos)

- es necesaria cierta estabilidad en el equipo
- ¿como escalar? aumentando el número de equipos
- multifuncional: reúne todas las habilidades necesarias para construir un incremento

especialistas

- su contribución es indispensable, pero no se mantendrá lo suficientemente ocupado
- tratarlo como proveedor externo lo convertiría en cuello de botella

mejor solución: eliminar la dependencia del especialista

- formar a otros miembros del equipo
- que el especialista se especialice en otras cosas

“La especialización es para insectos –Robert A. Heinlein, Time Enough for Love”

¿y el soporte?

equipos de soporte

- problemas de motivación
- menor aprendizaje (sólo aprenden de lo que se rompe)
- falta de comunicación con equipos de desarrollo puede llevar a repetir los mismos errores
- problemas de calidad: equipos de desarrollo saben que tienen respaldo

¿y el soporte?

equipos de desarrollo

- soportan su propio código: aprenden de sus errores
- los hace más lentos

alternativa: dedicar un porcentaje de tiempo planificado a soporte

¿y el soporte?

equipos de despliegue, sistemas...

- soportan al resto de equipos en algún aspecto técnico concreto del proceso de desarrollo
- podrían llevar al anti-patrón organizativo ágil típico:
organización basada en departamentos de silos de conocimiento

scrum master

es necesario que alguien se asegure de que el proceso sea efectivo

no es un jefe de proyecto

- facilita las reuniones
- ayuda al equipo a descubrir qué se interpone en su camino
- ayuda a entender scrum
- se pregunta ¿cómo podemos hacerlo mejor?

no hay gente mala, hay malos procesos

product owner

el cliente también forma parte del equipo

- participa en todas las reuniones
- define qué va y qué no va (funcionalidad) y en qué orden (prioridades)
- señala qué aporta valor
- decide las tareas a realizar en el siguiente sprint

¿quién es el cliente en nuestra organización?

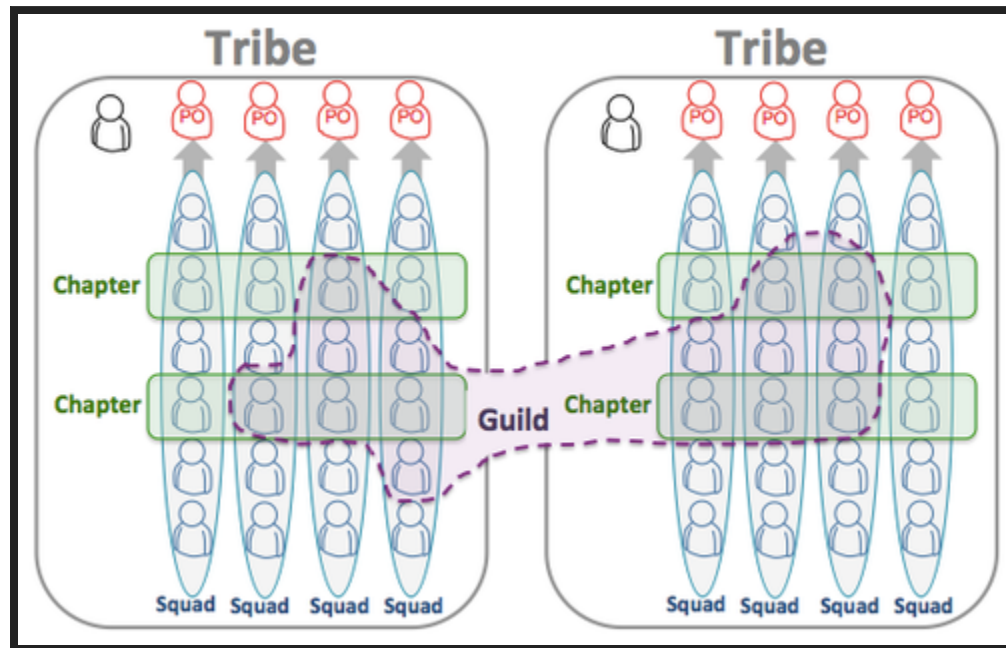
product owner

características necesarias:

- conocer el dominio y entender el proceso de negocio (qué se puede hacer y qué no)
- tener autoridad para tomar decisiones
- estar disponible para el equipo a diario
- ser responsable del valor entregado

muchas cosas... **puede haber un equipo de POs**

caso real



fuelle: [Spotify](#)

¿cómo somos los humanos?

- nos desconcentramos con facilidad
- pasamos más horas en la oficina de las que deberíamos
- somos horribles estimando cuánto nos llevará hacer algo

tenemos un problema con el tiempo

el sprint: origen

práctica copiada del MIT Media Lab en los 90:

cada 3 semanas, una demostración...

y si no es lo suficientemente buena, se cierra el proyecto

el sprint: claves

- cuanto antes entreguemos producto a los clientes, antes tendremos feedback
- trabajar durante un ciclo corto y comprobar dónde estamos
- una funcionalidad no está terminada si no puede usarse por el cliente
- una vez que el equipo se compromete con unas tareas, nadie puede añadir más

product backlog

lista ordenada de ideas para el producto realizada por distintos participantes

típicamente comienza siendo vago y pequeño y acaba siendo definido y extenso

sus tareas deben refinarse

está en **continua evolución**

debe contener todo aquello susceptible de ser incluido en el producto

product backlog

ordenar maximizando el valor entregado de las características:

- tienen el mayor impacto de negocio
- son más importantes para el cliente
- pueden producir el mayor beneficio
- son las más fáciles de implementar

“el 80% del valor está en el 20% de las características”

product backlog

características deseables

- con el detalle adecuado: dependiente del valor y la prioridad
- estimado
- emergente: sus contenidos cambian frecuentemente en función de las necesidades
- priorizado: elementos más importantes y urgentes aparecen en la parte superior de la lista

no es necesario priorizar todos los elementos en un primer momento

fuelle: [Scrum: the art of doing twice the work in half the time](#), Jeff Sutherland (2014)

sprint backlog

lista de tareas del product backlog refinadas a desarrollar en la próxima iteración

- determina el siguiente incremento de producto
- puede contener desglose de tareas
- en principio la lista de tareas se mantendrá inamovible durante toda la iteración

incremento de producto

algo que funciona

que tenga suficiente calidad

que sea aceptable para el cliente

que esté **terminado** :

funcional, documentado y con sus tests funcionando

no implica paso a producción

la stand-up diaria

reunión de pie de 10-15 min para preguntarse:

¿qué hice ayer para ayudar a conseguir los objetivos del equipo?

¿qué voy a hacer hoy para ayudar a conseguir los objetivos del equipo?

¿con qué dificultades me he encontrado?



origen: Borland

Quattro Pro for Windows:

- 1M líneas de código
- 31 meses
- 8 personas

¡¡mil líneas de código a la semana!!

¿cómo?

Fuente: [Borland Software Craftsmanship A New Look at Process, Quality and Productivity](#), Jim Coplien (AT&T)

saturación de comunicación

todos saben lo que se está haciendo

¿qué perjudica a la saturación de comunicación?

la especialización

roles, puestos, títulos... nos aferramos a sus funciones y a su nicho de conocimiento

reuniones diarias

todos los miembros del equipo se reunían a diario

- se discute qué tal se está haciendo
- se ofrece ayuda ante las dificultades
- se pone en común para conseguir cumplir los objetivos



¡ojo!

no lo convirtamos en un *reporting* individual...

“he hecho esto y voy a hacer esto otro”

¡el equipo debe demandar grandeza!

planificación del sprint

preparar el backlog: la pila de tareas que se va a vaciar

seleccionar, entender, estimar, refinar, priorizar y comprometer

en ciclos: todo está en movimiento

no más de 2 horas

revisión del sprint

revisión del incremento de producto

no más de 1 hora

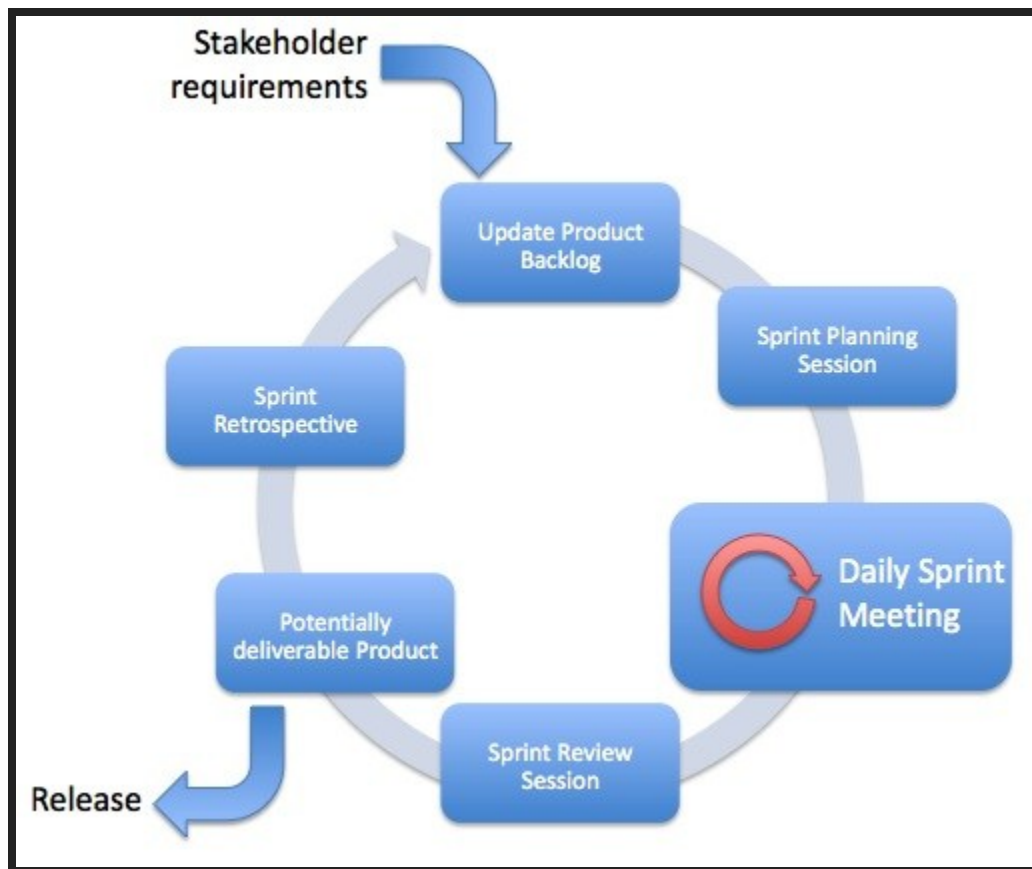
- ¿hace lo que debe hacer?
- ¿tiene la calidad suficiente?
- ¿cumple con los criterios de aceptación?

retrospectiva del sprint

puede hacerse junto con la revisión del sprint

- ¿cómo ha ido el proceso? relaciones personales, herramientas..
- identificación de posibles mejoras
- propuestas para el siguiente sprint

resumen



hablemos de desperdicio

¿todo lo que hacemos es aprovechable? ¿alguna vez hemos trabajado para nada?

¿qué porcentaje de nuestro tiempo de trabajo es realmente productivo?

una cosa a la vez

no somos multitarea... ejemplo: conducir y usar el móvil

¿entonces, por qué lo hacemos?

*porque nos es difícil inhibir el impulso de realizar
otra actividad*

hagamos un ejercicio

fuelle: [if you think you're good at multitasking, you probably aren't](#), David Sanbonmatsu (2013)

muy bonito, pero yo tengo que llevar 5 proyectos simultáneamente

proyectos simultáneos	tiempo disponible por proyecto	tiempo desperdiciado
1	100%	0%
2	40%	20%
3	20%	40%
4	10%	60%
5	5%	75%

fuentes: [Quality Software Management](#), Gerald M. Weinberg (1991)

esto tiene un nombre: [dual-task interference](#)

**debemos ser conscientes del coste del
cambio de contexto y minimizarlo junto
con las interrupciones**

técnicas: pomodoro, tiempo de soledad

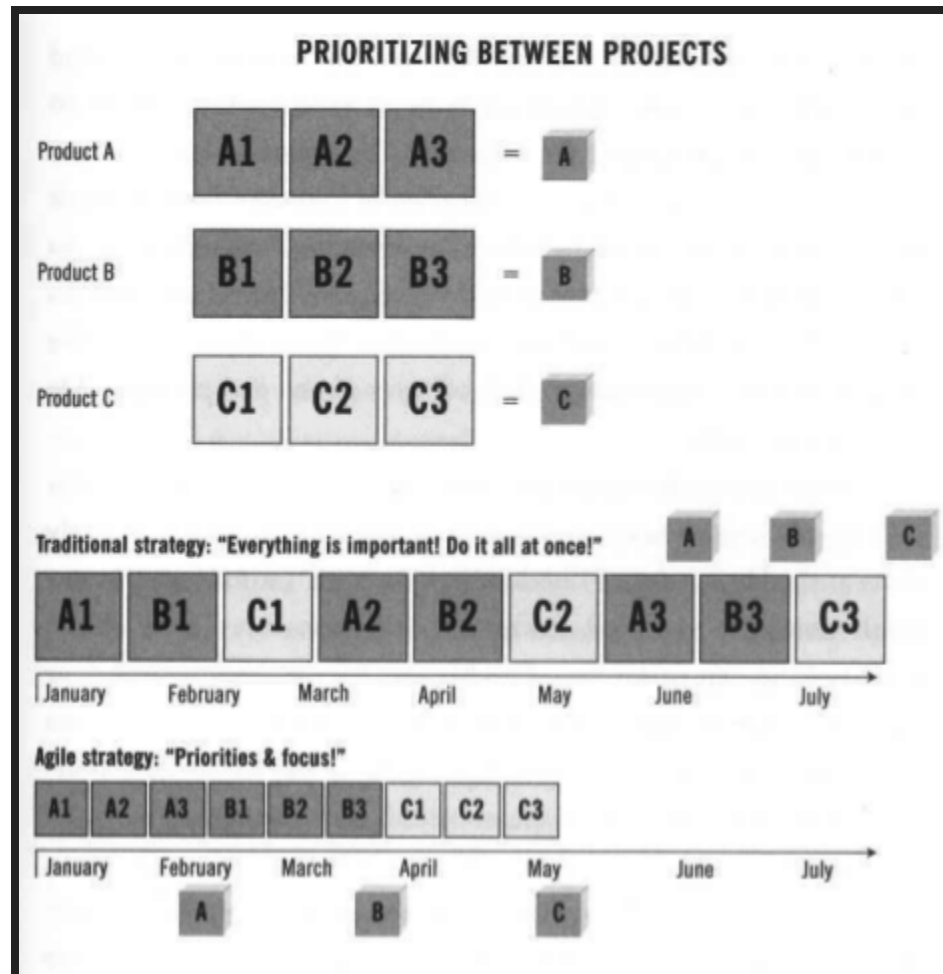
fuentes: [Rework](#): *interruption is the enemy of productivity*, David Heinemeier Hanson, Jason Fried



técnica pomodoro

1. decidir tarea a realizar
2. poner pomodoro a 25 minutos
3. trabajar intensivamente y anotar una X cuando el reloj suene
4. hacer pausa de 5 minutos
5. cada cuatro pomodoros, hacer pausa más larga (15 minutos)

pair programming con pomodoros...



fuentes: [Scrum: the art of doing twice the work in half the time](#), Jeff Sutherland (2014)

otras prácticas

¿cuándo debemos corregir un bug?

ejemplo: bugs en Palm (PDA)

- al crearlo: una hora
- tres semanas después: 24 horas

hagámoslo bien a la primera

otras prácticas


trabajar más hace producir menos

más horas -> fatiga -> errores -> corrección de errores
y otras: testarudez, falta de creatividad, menos moral, mayor
irritabilidad

¡hay que dormir!

[Rework: go to sleep](#), David Heinemeier Hanson, Jason Fried

[how naps affect your brain and why you should have one every day](#), Belle Beth Cooper (buffer social)



otras prácticas no seamos héroes

los esfuerzos heroicos son fracasos de planificación

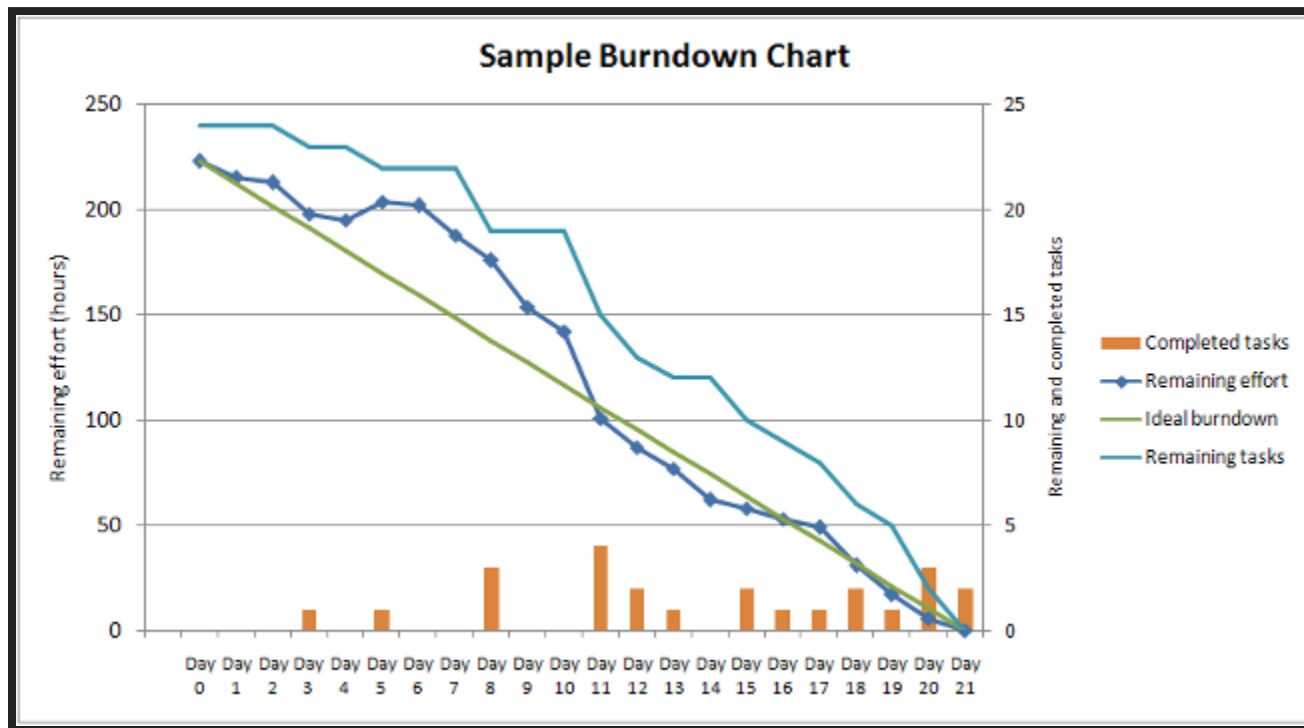
lo acabo por mis %\$%"·\$%&

descartar una tarea no implica fracaso

[Rework](#): *don't be a hero*, David Heinemeier Hanson, Jason Fried

métricas

gráfico burndown



fuelle: [Wikipedia](#)

velocidad de desarrollo

¿qué nos impide ir más rápido?

males endémicos de una organización:

proceso, personalidad y procedimiento

- personas sin la capacidad de decisión suficiente
- gente que no aparece en las reuniones
- personas que forman equipos sin estar en la misma ubicación

por algún motivo, la gente ve el cambio cultural a nivel de una organización como imposible. Yo solía estar de acuerdo con ellos, especialmente cuando se trataba de grandes compañías con una cultura y políticas osificadas -Jeff Sutherland

nada está escrito en piedra
cuestionémoslo todo

fuelle: [Scrum: the art of doing twice the work in half the time](#), Jeff Sutherland (2014)



[Acceso al repositorio del curso](#)

[Miguel Expósito Martín](#)