

Project Plan

Guevara Augusto y Perez Aballay Luciana

21 de Agosto de 2023

Gráfico Burndown

Un gráfico burndown es una herramienta visual utilizada en la gestión de proyectos, especialmente en metodologías ágiles como Scrum, para rastrear y mostrar el progreso del trabajo a lo largo del tiempo. Su objetivo principal es proporcionar una representación clara y actualizada de cómo el equipo está avanzando en la realización de tareas o historias de usuario planificadas para un sprint o período de tiempo determinado.

El gráfico burndown muestra dos ejes principales:

Eje horizontal (eje X): Representa el tiempo, generalmente en días o iteraciones.

Eje vertical (eje Y): Representa el trabajo pendiente o la cantidad de trabajo restante en el proyecto.

El término “burndown” se deriva del hecho de que a medida que el equipo completa tareas, el trabajo pendiente “quema” o disminuye gradualmente, mostrando un patrón descendente en el gráfico.

En un gráfico burndown, hay dos líneas principales que se representan:

Línea de trabajo planificado: Esta línea muestra la cantidad total de trabajo estimado para el proyecto. En cada día o iteración, esta línea generalmente disminuye en función del trabajo que se ha planificado completar.

Línea de trabajo real: Esta línea representa la cantidad real de trabajo completado por el equipo a medida que avanza el proyecto. Idealmente, esta línea también debería disminuir a medida que el equipo completa tareas.

El objetivo es que la línea de trabajo real siga de cerca la línea de trabajo planificado. Si hay desviaciones, el equipo puede identificarlas rápidamente y tomar medidas para ajustar la planificación o abordar cualquier problema que pueda surgir.

El gráfico burndown es una herramienta útil para que los equipos y los stakeholders puedan evaluar visualmente cómo está progresando el trabajo en comparación con las expectativas. También ayuda a identificar posibles retrasos o aceleraciones en el proyecto, lo que permite tomar decisiones informadas para garantizar el éxito del proyecto en general.

Planificación inicial: Al comienzo del proyecto o sprint, el equipo de desarrollo y el Product Owner identifican las tareas o historias de usuario que deben completarse. Estiman el esfuerzo necesario para cada tarea en unidades como puntos de historia o días de trabajo.

Creación del gráfico: En un eje, generalmente horizontal (X), coloca el tiempo en unidades de tiempo (días, semanas, iteraciones) a lo largo del proyecto o sprint. En el otro eje, generalmente vertical (Y), coloca la cantidad total de trabajo pendiente o los puntos estimados.

Línea de trabajo planificado: Dibuja una línea que representa la cantidad total de trabajo estimado en el proyecto. Esta línea debe disminuir gradualmente a medida que se completan las tareas.

Registro diario o periódico: Al final de cada día o período (como una iteración), registra la cantidad real de trabajo completado por el equipo. Actualiza el gráfico dibujando una línea que representa la cantidad de trabajo pendiente en ese momento.

Comparación y análisis: A medida que avanzas en el proyecto, compara la línea de trabajo planificado con la línea de trabajo real. Si la línea de trabajo real está por encima de la línea de trabajo planificado, puede indicar un retraso en el progreso. Si está por debajo, puede indicar que el equipo está avanzando más rápido de lo esperado.

Toma de decisiones: Utiliza el gráfico burndown como una herramienta para la toma de decisiones. Si hay desviaciones significativas, el equipo y el Product Owner pueden reunirse para discutir las razones detrás de esos cambios y tomar medidas correctivas.

Ajustes y adaptaciones: Si hay desviaciones entre las líneas planificadas y reales, el equipo puede adaptar su enfoque. Esto podría implicar asignar más recursos a ciertas tareas, reevaluar la prioridad de las historias de usuario o tomar otras medidas para garantizar que el proyecto se mantenga encaminado.

Seguimiento continuo: Mantén el gráfico burndown actualizado a lo largo del proyecto para tener una imagen clara del progreso y cómo se están cumpliendo las estimaciones. En resumen, el gráfico burndown es una herramienta poderosa para rastrear el progreso en proyectos ágiles. Proporciona visibilidad en tiempo real sobre el estado del trabajo y permite a los equipos y a los stakeholders tomar decisiones informadas para garantizar el éxito del proyecto.

Para generar este tipo de gráficos primero cargamos la herramienta “*plan*”, una vez instalado el paquete insertamos el código:

```
library("plan")
data(burndown)
plot(burndown)
```

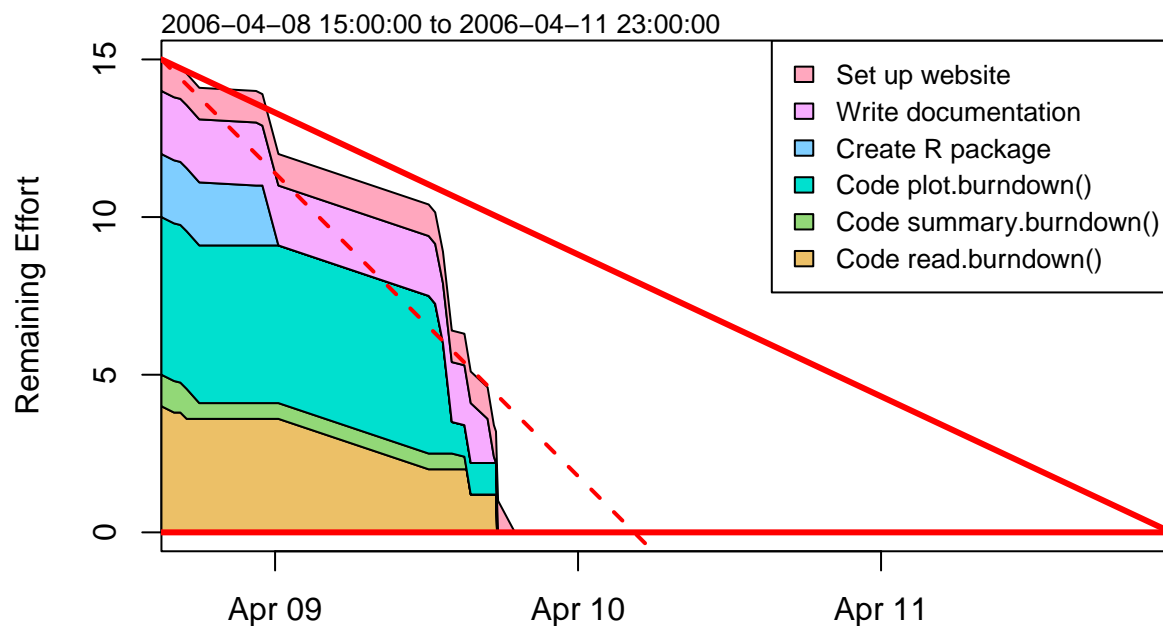


Diagrama de Gantt

Un diagrama de Gantt es una herramienta visual utilizada en la gestión de proyectos para planificar, programar y monitorear el progreso de las tareas a lo largo del tiempo.

Cada tarea se representa como una barra horizontal en el gráfico, que comienza en la fecha de inicio prevista de la tarea y termina en la fecha de finalización prevista. La longitud de la barra representa la duración estimada de la tarea.

Elementos clave de un diagrama de Gantt:

Tareas: Cada tarea del proyecto se representa mediante una barra en el gráfico. La posición y longitud de la barra indican cuándo comienza y termina la tarea.

Fechas: El eje de tiempo muestra las fechas en las que se llevarán a cabo las tareas. Puede ser un calendario diario, semanal o mensual, según la escala del proyecto.

Dependencias: Las dependencias entre tareas se pueden representar en el diagrama de Gantt. Esto muestra qué tareas deben completarse antes de que otras puedan comenzar.

Hitos: Los hitos son eventos o puntos de referencia importantes en el proyecto que no tienen duración. Se representan como puntos o diamantes en el gráfico y ayudan a resaltar momentos críticos.

Línea de tiempo: En algunos casos, se puede incluir una línea vertical que indica la fecha actual. Esto permite al equipo ver fácilmente si están en camino, adelantados o retrasados en relación con la planificación.

Los diagramas de Gantt son valiosos porque ofrecen una representación visual clara del proyecto en su conjunto y ayudan a los equipos a comprender cómo las tareas se superponen, cuándo se alcanzarán los hitos y si hay algún retraso en el programa. Esto permite a los gestores y equipos tomar decisiones informadas, asignar recursos adecuadamente y ajustar la planificación según sea necesario.

Hoy en día, muchas herramientas de software permiten crear y gestionar diagramas de Gantt de manera fácil y eficiente, lo que ha contribuido a su popularidad continua en la gestión de proyectos.

Para realizar este tipo de gráficos tenemos que, una vez cargada la herramienta “*plan*”, insertar el siguiente código:

```
library("plan")
g <- new("gantt")
g <- ganttAddTask(g, "Módulo TyHM II") # no times, so a heading
g <- ganttAddTask(g, "Physical Oceanography", "2023-09-03", "2023-12-05", done=100)
g <- ganttAddTask(g, "Chemistry Oceanography", "2023-09-03", "2023-12-05", done=100)
g <- ganttAddTask(g, "Fluid Dynamics", "2023-09-03", "2023-12-05", done=100)
g <- ganttAddTask(g, "Biological Oceanography", "2024-01-03", "2024-04-05")
g <- ganttAddTask(g, "Geological Oceanography", "2024-01-03", "2024-04-05")
g <- ganttAddTask(g, "Time-series Analysis", "2024-01-03", "2024-04-05")
g <- ganttAddTask(g, "Redacción Tesina") # no times, so a heading
g <- ganttAddTask(g, "Literature review", "2023-09-03", "2024-02-01", done=20)
g <- ganttAddTask(g, "Develop analysis skills", "2023-09-03", "2024-08-01", done=30)
g <- ganttAddTask(g, "Thesis work", "2023-10-01", "2024-09-01")
g <- ganttAddTask(g, "Defend thesis proposal", "2024-05-01", "2024-06-01")
g <- ganttAddTask(g, "Write papers & thesis", "2024-03-01", "2024-04-01")
g <- ganttAddTask(g, "Defend thesis", "2024-09-01", "2024-10-15")
font <- ifelse(is.na(g[["start"]]), 2, 1)
plot(g, ylabel=list(font=font),
event.time="2024-01-01", event.label="Report Date")
par(lend="square") # default is round
legend("topright", pch=22, pt.cex=2, pt.bg=gray(c(0.3, 0.9)),
border="black", xpd=NA,
legend=c("Completado", "Por realizar"), title="Diploma Ingeniero", bg="white")
```

