

Pontificia Universidad Javeriana



Primera Entrega Proyecto
Fundamentos Ingenieria de Software

ROPA

Laura Sofia Aponte Sánchez
Juan Esteban Bello Durango
Santiago Galindo Cubillos
Juan Felipe Gutierrez Adarme

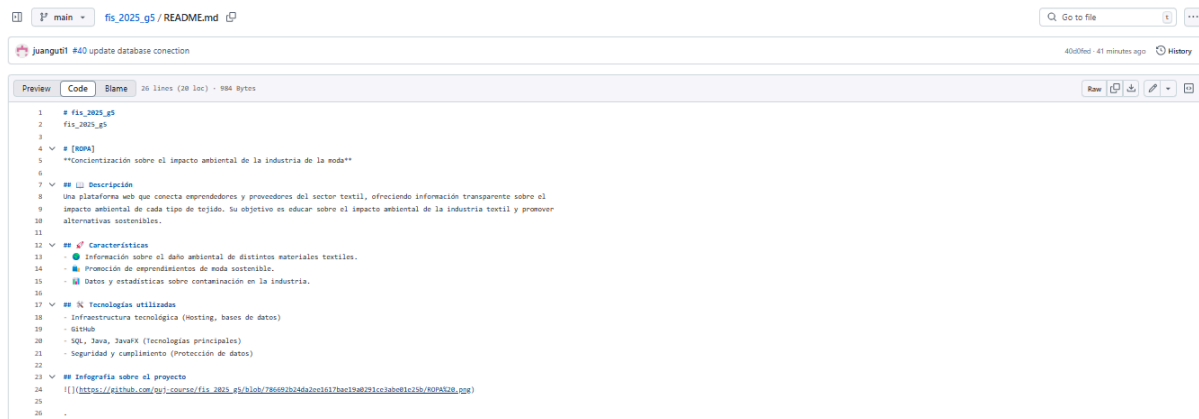
Ing. Luis Gabriel Moreno Sandoval. PhD

Bogotá

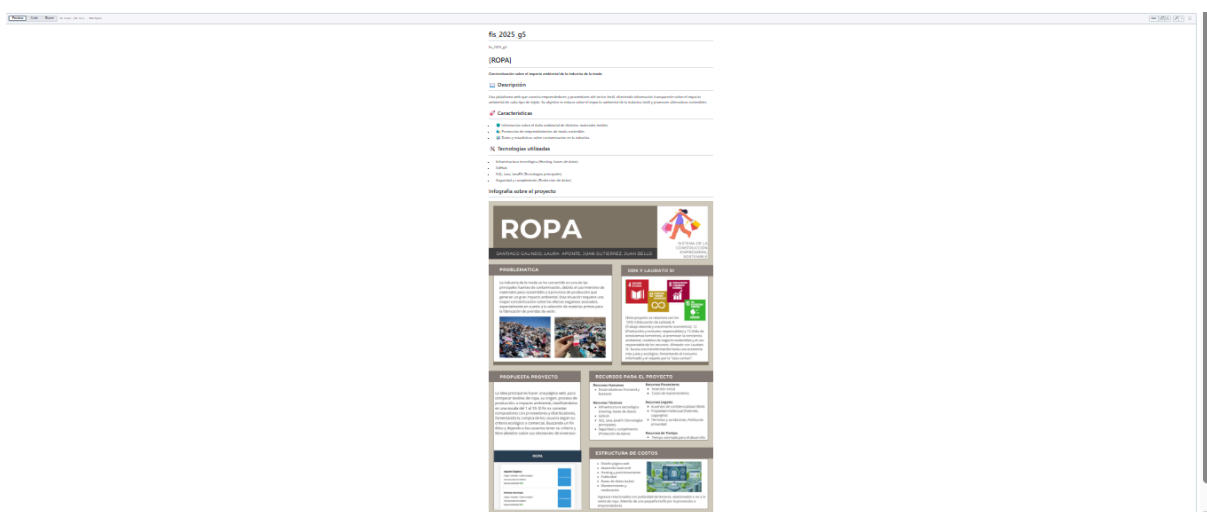
10 de Marzo de 2023

1. Estructura del proyecto

Organización del repositorio: El repositorio debe estar bien estructurado para facilitar la colaboración. Un archivo README.md es esencial para que cualquier persona que acceda al proyecto entienda rápidamente su propósito, cómo configurarlo e instalarlo. La estructura debe incluir carpetas que separen el código, los tests, y otros recursos, lo que facilita la navegación y la escalabilidad del proyecto.

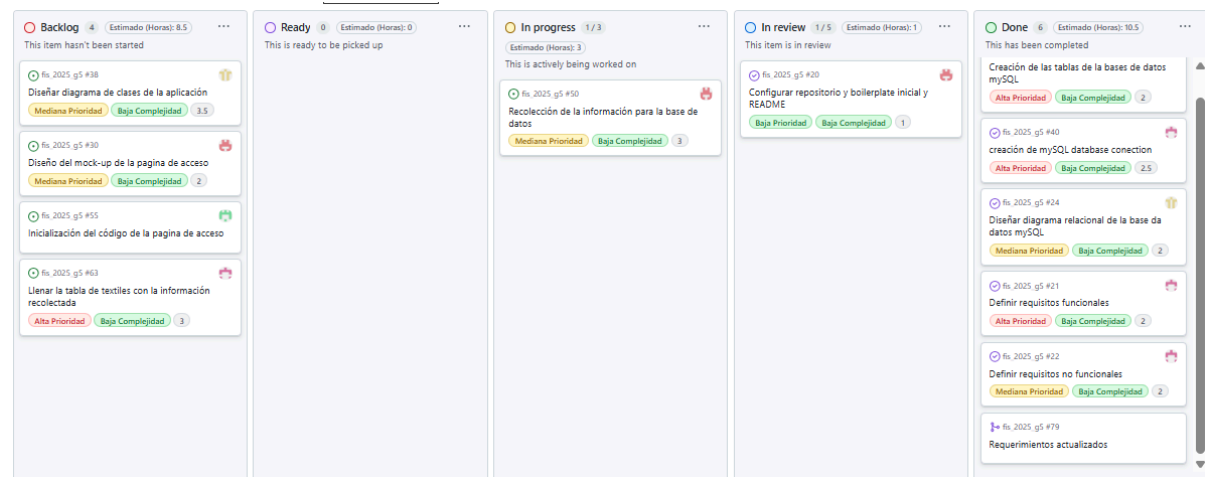


```
1 # fis_2025_g5
2 fis_2025_g5
3
4 # [ROPA]
5 ""Concientización sobre el impacto ambiental de la industria de la moda""
6
7 ## Descripción
8 Una plataforma web que conecta emprendedores y proveedores del sector textil, ofreciendo información transparente sobre el
9 impacto ambiental de cada tipo de tejido. Su objetivo es educar sobre el impacto ambiental de la industria textil y promover
10 alternativas sostenibles.
11
12 ## Características
13 - Información sobre el daño ambiental de distintos materiales textiles.
14 - Promoción de emprendimientos de moda sostenible.
15 - Datos y estadísticas sobre contaminación en la industria.
16
17 ## Tecnologías utilizadas
18 - Infraestructura tecnológica (hosting, bases de datos)
19 - GitHub
20 - SQL, Java, JavaFX (tecnologías principales)
21 - Seguridad y cumplimiento (Protección de datos)
22
23 ## Infografía sobre el proyecto
24 []
25
26
```



2. Tablero kanban

Flujo de trabajo ágil visualizado: Refleja cómo se mueve cada tarea a lo largo del proyecto. Las columnas Backlog, Ready, In Progress y Done. Además, el Roadmap debe incluir la planificación de los sprints, detallando las tareas que se deben completar en cada ciclo de trabajo, en este caso sprints de 1 semana.



3. Definición de Historias de Usuario

Las historias de usuario son la unidad básica de trabajo en metodologías ágiles. En GitHub, cada tarea o historia de usuario debe tener un título claro, una descripción detallada de lo que se espera, una prioridad (alta, media, baja), y una estimación de complejidad y esfuerzo en horas que definimos de la siguiente forma: baja(0 a 4 horas), media (4 a 8 horas), y alta(8 o más horas). Además, cada historia debe estar vinculada a un issue, y asignada a un miembro del equipo para garantizar que se complete dentro del sprint correspondiente. A continuación un ejemplo de historia de usuario en issue.

creación de mysql database connection #40

🔒 Closed Task | Parent: pordefinir | puj-course/fis_2025_g5 Public

GalindoS420 opened last week · edited by juanguti1

crear la conexión de la base de datos a partir de las credenciales

Create sub-issue

GalindoS420 added Backend Frontend last week

GalindoS420 added this to the Revisión Entrega 2 milestone last week

GalindoS420 self-assigned this last week

GalindoS420 added the Task issue type last week

GalindoS420 added a parent issue last week

pordefinir #37

GalindoS420 added this to fis_2025_g5_ROPA last week

GalindoS420 moved this to Backlog in fis_2025_g5_ROPA last week

GalindoS420 assigned bello-juan 5 days ago

juanguti1 removed this from the Revisión Entrega 2 milestone yesterday

juanguti1 changed the title Añadir lógica para mostrar datos desde el backend pordefinir yesterday

github-project-automation added this to KAMBAN_FIS_2025_G5 yesterday

juanguti1 changed the title pordefinir creacion de mysql database yesterday

Assignees: juanguti1

Labels: Backend Frontend

Type: Task

Projects: KAMBAN_FIS_2025_G5

Status: Done

Priority: Alta Prioridad

Complejidad: Baja Complejidad

Estimado (Horas): 2.5

Sprint: Sprint 3 · Mar 4 · Mar 10

1 closed project

Milestone: Base de Datos

Due by March 16, 2025, 60% complete

Relationships: Parent issue

pordefinir puj-course/fis_2025_g5#37 2/2

4. Milestones (Épicas)

Las épicas son grandes tareas o entregables que se dividen en tareas más pequeñas (issues) a lo largo de varios sprints. Esto permite tener una visión general de las entregas clave, como en este caso la finalización de un módulo de la página, por ejemplo la base de datos, la página principal, la página de acceso etc.

The screenshot shows the 'Milestones' view in Jira. At the top, there are tabs for 'Labels' and 'Milestones', and a 'New milestone' button. Below the tabs, a summary bar indicates '5 Open' and '0 Closed' items. The main content area lists five milestones, each with a title, due date, last updated time, progress bar, and status (complete, open, closed). Each item also has 'Edit', 'Close', and 'Delete' links.

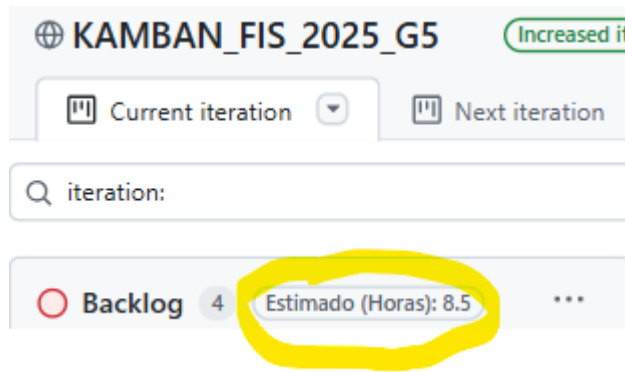
Milestone	Due by	Last updated	Progress	Status
Base de Datos	March 17, 2025	5 minutes ago	60% complete	2 open, 3 closed
Pagina de Emprendimientos	April 28, 2025	3 hours ago	0% complete	0 open, 0 closed
Pagina de Proveedores	April 14, 2025	3 hours ago	0% complete	0 open, 0 closed
Página de acceso	April 07, 2025	3 hours ago	42% complete	4 open, 3 closed
Pagina de inicio	March 31, 2025	3 hours ago	0% complete	0 open, 0 closed

5. Estimación de Esfuerzo:

En un proyecto ágil, la estimación de esfuerzo es clave para planificar y distribuir las tareas de manera realista. En este caso se utilizó la herramienta Poker Scrum (donde el equipo vota el esfuerzo de cada tarea y se comparan los resultados) para hacer la estimación de esfuerzo de cada tarea y esto se refleja en la asignación de tiempo a los issues, asegurando que todo el equipo tenga claridad sobre lo que se espera para completar una tarea. Esto se refleja en los issues y en las columnas del kamban.

The screenshot shows a Jira Kanban board. On the left, a list of task history items is visible, including 'Diseñar diagrama de clases de la aplicación #38'. On the right, a detailed view of a task is shown. The task is titled 'Diseñar diagrama de clases de la aplicación #38' and is currently in the 'Backlog' status. The task's priority is 'Mediana Prioridad', complexity is 'Baja Complejidad', and the estimated effort is '3.5' hours. The task is assigned to 'GalindoS420' and is currently in the 'Sprint 4' column, which runs from March 11 to March 17.

Task	Status	Priority	Complejidad	Estimado (Horas)	Sprint
Diseñar diagrama de clases de la aplicación #38	Backlog	Mediana Prioridad	Baja Complejidad	3.5	Sprint 4 (Mar 11 - Mar 17)



Asimismo, se utilizó el Poker Scrum para definir el tiempo estimado de cada issue:

Issue	Laura Aponte	Juan Bello	Santiago Galindo	Juan Gutierrez	Conclusión (horas)
#23	3	3	1	1	2
#40	3	2	2	4	2.5
#21	1	2	2	2	2

#23 Creación de tablas de la base de datos

#40 Conexión de base de datos y java

#21 Requisitos funcionales

En el primer y tercer caso se promediaron los valores pues no se alejaban mucho entre sí, mientras que en el segundo caso como los valores era un poco más alejados entre sí se discute entre los integrantes el porque consideraban ese valor y después de escuchar todos los puntos se determinó una duración de 2.5

6. Uso de Metodologías Ágil

La metodología ágil requiere una organización en el control de versiones para mantener un flujo de trabajo eficiente. En GitHub:

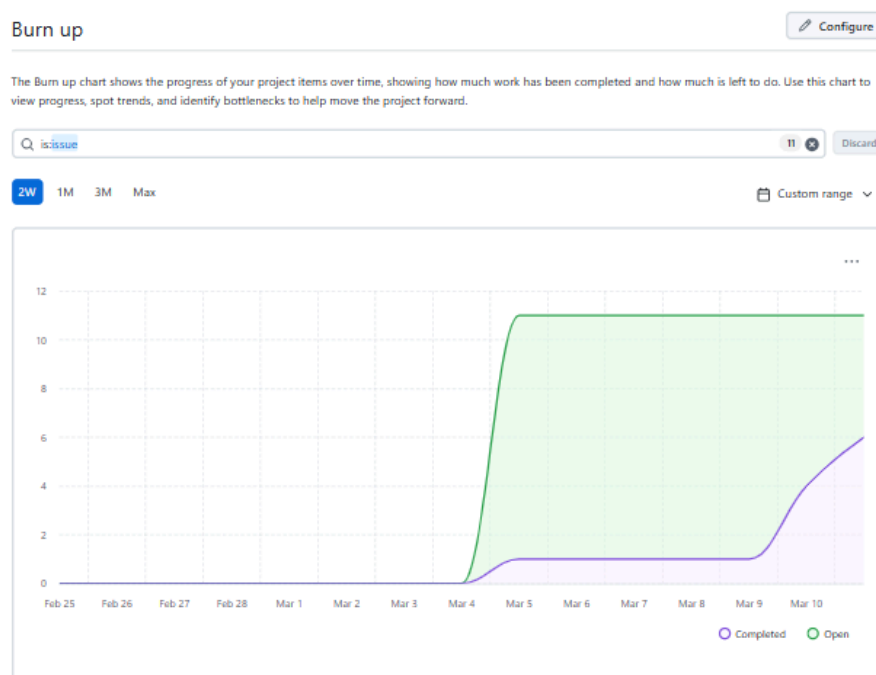
- Rama Develop: Los cambios diarios deben realizarse en la rama develop, lo que asegura que el código siempre esté en un estado funcional.
- Rama Main: La rama main solo debe contener las versiones estables y liberadas del proyecto, por lo que no debe recibir cambios constantes, solo en momentos clave (como la entrega de un sprint).
- Ramas personales: Cada miembro del equipo debe crear una rama para trabajar en sus tareas asignadas, lo que evita conflictos y permite un trabajo más organizado.

- Merge: Una vez que las tareas están completas, las ramas personales deben fusionarse de nuevo en develop o, en algunos casos, directamente en main para entregas más grandes. Esto asegura que todos los miembros contribuyan al proyecto de forma efectiva.
- La metodología ágil definida es poker scrum.


7. Gráficos de Seguimiento


Los gráficos de seguimiento son herramientas visuales que ayudan a monitorear el progreso del equipo a lo largo del ciclo de vida del proyecto. Estos gráficos permiten a los miembros del equipo, gerentes y stakeholders tener una visión clara y actualizada sobre el estado del proyecto, facilitando la toma de decisiones informadas.

- Gráfico de Burn Up: A diferencia del Burn Down, el gráfico de Burn Up muestra el progreso hacia un objetivo total, en lugar de mostrar el trabajo pendiente. Este gráfico refleja el trabajo completado (en el eje Y) frente al tiempo (en el eje X). Además, es más útil para ver cambios en el alcance del proyecto, ya que puedes ver cómo el trabajo pendiente (el total de tareas) cambia a lo largo del tiempo.



-
- Gráfico Status por Asignado: Este gráfico muestra la distribución de los issues abiertos y cerrados en función de los miembros del equipo a los que están asignados. Permite evaluar la distribución equitativa del trabajo dentro del equipo y detectar si algún miembro tiene demasiados issues asignados en comparación con otros. Está representado como una gráfica de barras

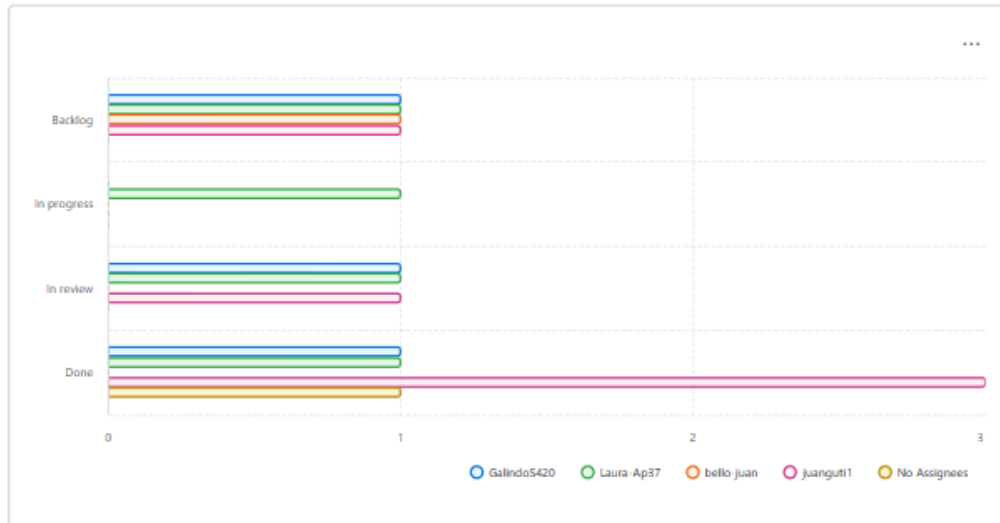
Chart 1 - Status por Asignado 

 Configure


Filter by keyword or by field


Discard

Save



- Gráfico Milestone por Prioridad: Este gráfico clasifica los issues según su prioridad (alta, media, baja) y los agrupa según el milestone al que pertenecen. Ayuda al equipo y a los stakeholders a identificar si los milestones están correctamente balanceados en términos de prioridad de los issues. Se representa en una grafica de columnas.

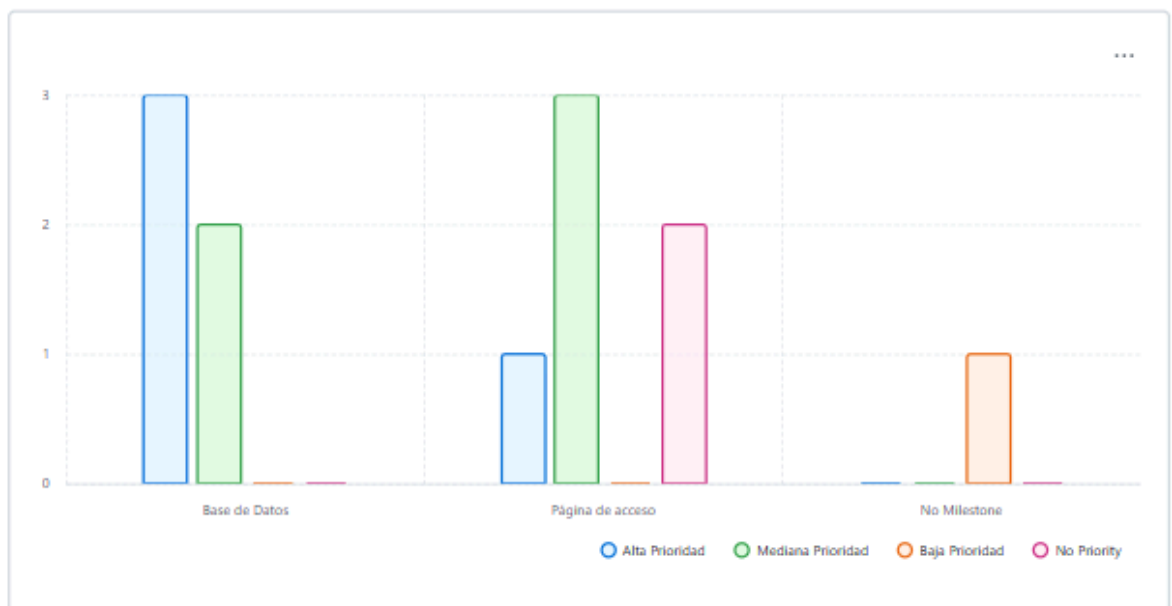
Chart 2 - Milestone por Prioridad 

 Configure

Filter by keyword or by field

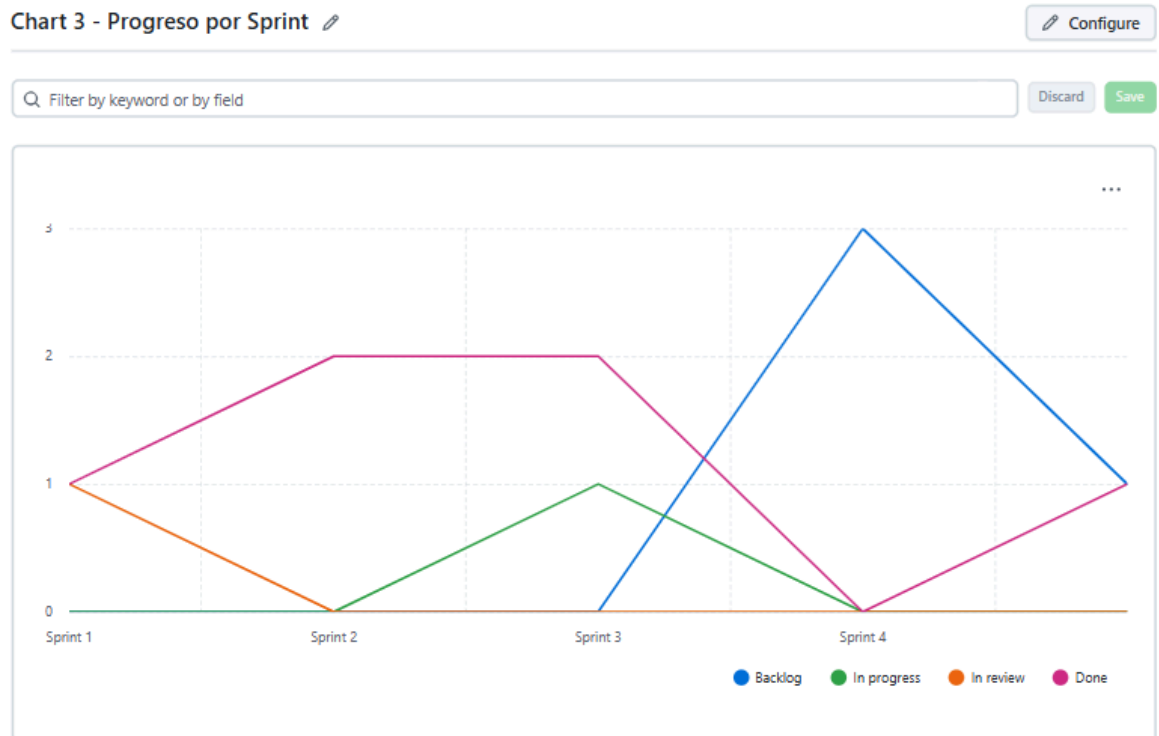
Discard

Save




- Gráfico de Progreso por Sprint: Este gráfico de líneas muestra la evolución del estado de los issues a lo largo de los sprints. En el eje X se representan los sprints, y en el eje Y, la cantidad de issues en cada estado (Backlog, In Progress, Done,

etc.). A medida que avanza el sprint, las líneas reflejan cómo los issues cambian de estado, mostrando el flujo de trabajo del equipo.



-
- Gráfico de Complejidad por Sprint: es útil para evaluar cómo varía la dificultad de las tareas en un proyecto ágil y para ajustar la planificación, las estimaciones y la capacidad del equipo a medida que avanzan los sprints. Facilita la toma de decisiones sobre el manejo de la carga de trabajo y mejora la precisión de las estimaciones en futuros sprints.

Chart 4 -Complejidad por Sprint 

 Configure

 Filter by keyword or by field

Discard

Save

