

MyTask: Requerimientos, Organización y Flujo de Trabajo

Diego Moyano, Sebastian Castro
MyTask/GitHub

September 9, 2024

Contents

1	Especificación de Requerimientos (Validación)	3
1.1	Validación	3
2	Verificación: Asegurando que el Programa Cumpla los Requerimientos	3
3	Organización y Flujo de Trabajo	3
3.1	Organización del Proyecto	3
3.2	Flujo de Trabajo	4
3.3	Evidencia de Flujo de Trabajo y Configuraciones	4
4	Problemas Encontrados y Soluciones	6
5	Conclusión	6

1 Especificación de Requerimientos (Validación)

1.1 Validación

Para asegurar que los requerimientos se especifiquen de la mejor manera, se pueden adoptar las siguientes medidas:

- Análisis de necesidades: Revisar en conjunto del usuario los objetivos del sistema, interiorizándose además con el funcionamiento de la aplicación.
- Diseño de pruebas unitarias basadas en los casos de uso más comunes, donde cada desarrollador testeará sus módulos construidos.

2 Verificación: Asegurando que el Programa Cumpla los Requerimientos

Para verificar que el programa cumple con los requerimientos especificados, se llevaron a cabo las siguientes acciones:

- Pruebas funcionales: Evaluación de cada una de las funcionalidades del sistema de gestión de tareas.
- Pruebas de integración: Asegurar que el manejo de archivos JSON interactúa correctamente con el sistema.
- Pruebas de regresión: Revisar que las nuevas características no afecten las funcionalidades existentes.
- Pruebas manuales: Validación a través de la ejecución manual de casos de uso.

3 Organización y Flujo de Trabajo

3.1 Organización del Proyecto

Para la organización del proyecto se utilizarán ramas de git, haciendo uso de gitflow como modelo de creación de ramas. Cada desarrollador hizo uso de su propia rama, bajo el nombre de usuario de git, de esta manera se trabajará de manera independiente y se irán subiendo los cambios a una única rama llamada develop, la que para este caso actuará como la rama principal de la tarea.

3.2 Flujo de Trabajo

El flujo de trabajo sigue una estructura ágil en la que se planifican tareas de acuerdo al tablero en Jira, avanzando metódica y constantemente, revisando el trabajo del par por medio de las pull request de cada uno, de esta manera ambos desarrolladores serán capaces de revisar qué es lo que se sube a la rama principal y con qué tendrán que trabajar posteriormente.

- Desarrollo de funciones clave: Tareas, almacenamiento, y pruebas.
- Protección de ramas por medio de la utilización de ramas propias diferenciadas por los nombres de usuario.
- Revisión de código: Mediante pull requests en GitHub -¡ Pruebas cruzadas.
- Pruebas unitarias, cada desarrollador testea su código con el fin de poder subir código de calidad a la rama principal.

3.3 Evidencia de Flujo de Trabajo y Configuraciones

En esta sección se incluyen capturas de pantalla que muestran el progreso y las configuraciones, además de la configuración de slack con Jira y Git.

```
daleo@DESKTOP-SIIR4UV MINGW64 ~/OneDrive/Escritorio/Pruebas de SW/Tarea/Main (develop)
$ git branch
Seed4407
* develop
dmoyan0
master
```

Figure 1: Configuración ramas de git

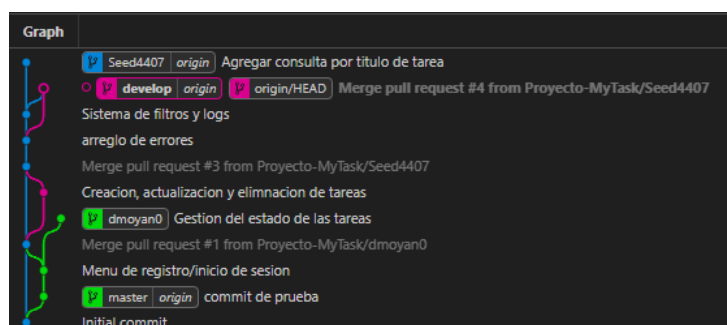


Figure 2: Gráfico de las ramas de GitFlow

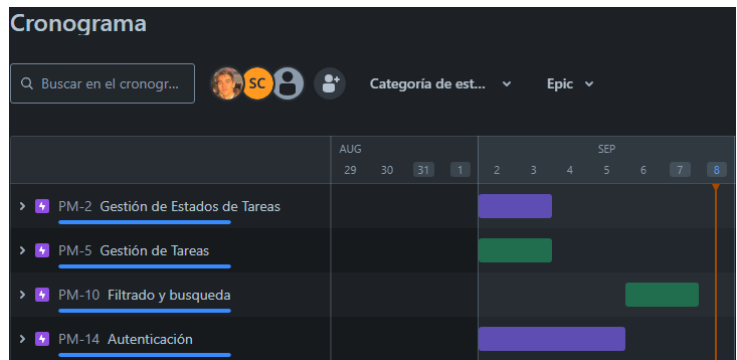


Figure 3: Cronograma de Jira



Figure 4: Mensajes de Git en Slack

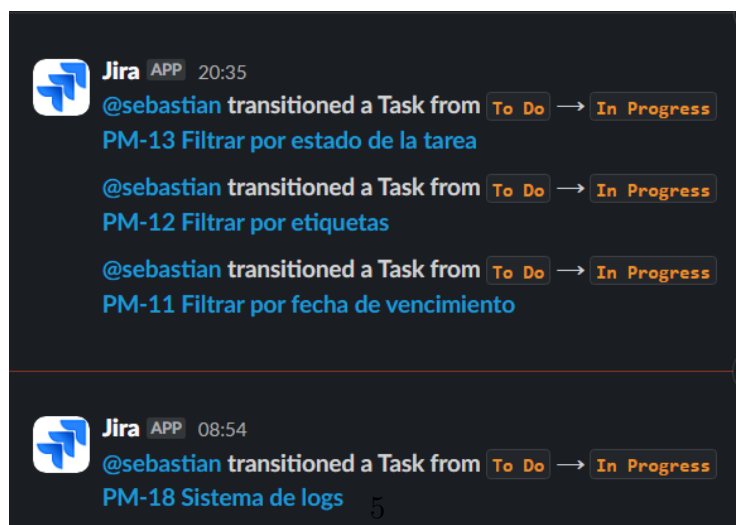


Figure 5: Mensajes de Jira en Slack

4 Problemas Encontrados y Soluciones

Durante el desarrollo del proyecto, se enfrentaron los siguientes problemas:

- **Problema 1:** Conflictos en PullRequests, se pierde parte del avance al aceptar una pullrequest antes de aceptar la anterior.
 - **Solución:** Se pone en práctica revisar las pullrequest de manera más rápida, de manera de no perder información, además de mantener una copia del avance en local.
- **Problema 2:** Fallos en el formato de JSON, ambos desarrolladores ocupamos formatos distintos para el mismo problema, por lo que dificultó la integración de los módulos.
 - **Solución:** Se mejora la comunicación entre los miembros del equipo, de manera de reducir este tipo de errores.
- **Problema 3:** Desconocimiento de GitFlow
 - **Solución:** Se estudia y por medio de la práctica se aprende del uso de este método.

5 Conclusión

Este proyecto ha permitido el desarrollo de un sistema de gestión de tareas que cumple con los requerimientos especificados. A través de un flujo de trabajo metodico y un control riguroso de ramas, se ha logrado una solución eficiente y se ha aprendido a trabajar con métodos distintos de desarrollo.