

***Equipo 6***

# MyTask

Sistema para gestionar tareas

Diego Morales Arellano  
Rodrigo Vera Velásquez

---

[Enlace Repositorio](#)

## Índice

Introducción.....	2
¿Cómo especificarías mejor el requerimiento? .....	2
¿Cómo asegurarías que el programa cumpla el requerimiento?.....	2
Organización y el flujo de trabajo.....	3
Problemas encontrados .....	6

## Introducción

En la actualidad, la gestión eficiente de tareas habituales es una necesidad muy importante en entornos personales como laborales. Un cliente ha solicitado el desarrollo de un sistema que permita organizar sus tareas de manera sencilla y práctica. En respuesta a esta solicitud, nace **MyTask**, una aplicación diseñada para gestionar tareas de forma rápida y efectiva desde la línea de comandos. **MyTask** permite a los usuarios crear, consultar, actualizar y eliminar tareas, asegurando que cada una esté categorizada según su estado y etiqueta. El sistema está estructurado para facilitar la gestión de tareas activas y completadas, permitiendo una transición fluida entre diferentes estados (pendiente, en progreso, completada) y archivando aquellas completadas para diferenciarlas. Además, el sistema incluye un sistema de autenticación que garantiza la seguridad del acceso diferenciando entre usuarios y tareas, así como funcionalidades avanzadas de búsqueda y filtrado que ayudan a los usuarios a organizar sus actividades de manera eficiente.

## ¿Cómo especificarías mejor el requerimiento?

Mediante el contexto del requerimiento del cliente, como equipo nos centraremos en asegurar que el sistema resuelva el problema correcto, esto quiere decir, un sistema para gestionar sus tareas de forma eficiente, para ello debemos tomar en consideración las necesidades reales del cliente, asegurando plena funcionalidad de cada requerimiento para que estas estén acorde a sus expectativas y propósitos, en otras palabras, lo justo y necesario para que lo sienta suficiente a lo que requiere. Por ende, por cada funcionalidad tendremos:

- Gestor de tareas: Asegurarse de que el sistema permita a los usuarios gestionar sus tareas de manera sencilla y directa
- Filtrado y Búsqueda de tareas: Validar que los filtrados y búsquedas funcionen de tal manera que el cliente pueda encontrar la información rápidamente, cubriendo cualquier necesidad.
- Gestión de Estados de Tareas: Confirmar que el cliente pueda tener control de los estados de sus tareas, para archivar aquellas que sean completadas y hacer diferencia entre las tareas activas.
- Autenticación: Mantener la seguridad del sistema con un sistema de autenticación seguro para cada usuario.

## ¿Cómo asegurarías que el programa cumpla el requerimiento?

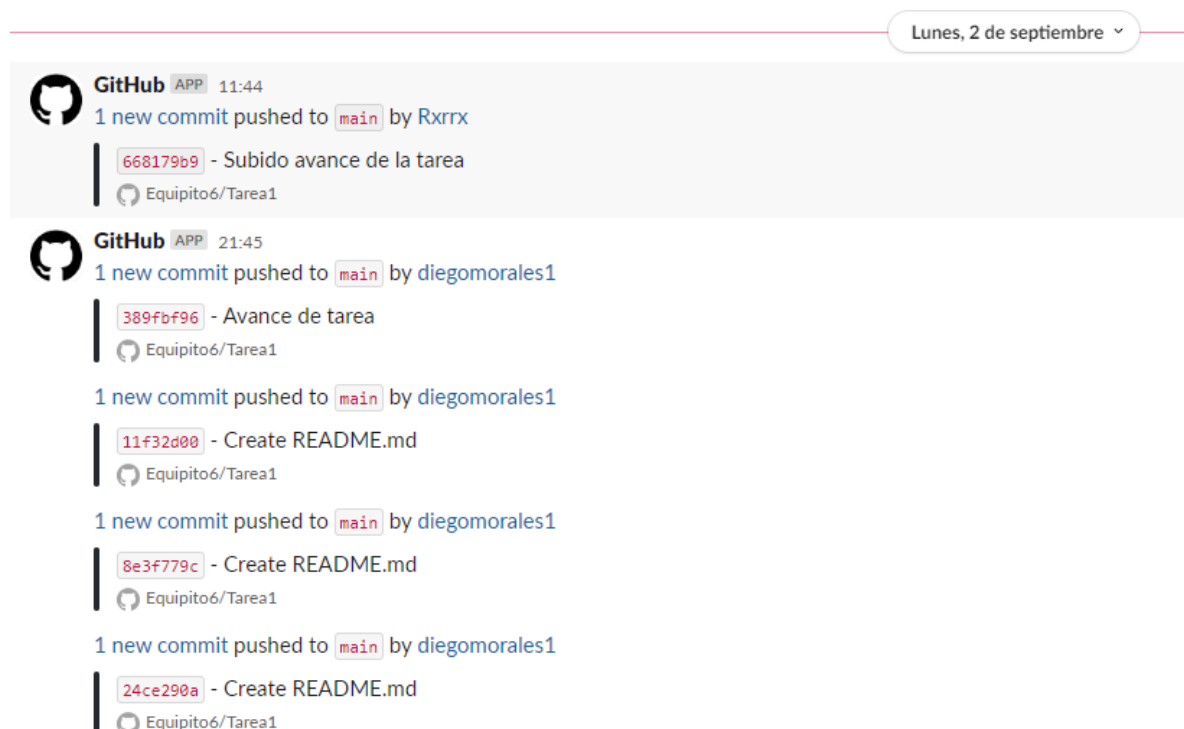
Para asegurar que el sistema cumpla con el requerimiento, se debe de confirmar que el sistema esté implementado de acuerdo con las especificaciones dadas por el cliente, además de que las funcionalidades requeridas funcionen tal cual como se describieron, para ello se deben de tener en consideración todos los escenarios posibles donde se puedan presentar fallos, tener un control de auditoria y presentar logs en casos necesarios. Todo lo ultimo mencionado nos permite tener un control sobre lo que esta pasando para acercarnos a lo que requiere el cliente, además de recrear pruebas funcionales y unitarias para cada funcionalidad implementada.

## Organización y el flujo de trabajo

En primera instancia conversamos como equipo para distribuir tareas y empezar a trabajar en cada parte de los requerimientos, para esto se implementó un código base, un código esqueleto para darle vida de apoco, integrando cada vez nuevas funcionalidades. Aquel código base fue la primera versión de este, por lo tanto, a medida que pasaba el tiempo fuimos implementando las funcionalidades subiéndolas al Github del equipo con un README especificando que es lo nuevo que contiene aquella versión, que cambios se realizaron y que es lo nuevo implementado.






Como ultimo podremos mencionar las pruebas unitarias realizadas para verificar la funcionalidad de cada uno, así poder acercarnos cada vez mas a las expectativas del cliente. Utilizamos el método de pruebas cruzadas, esto quiere decir, tomar el código hecho por el otro integrante y hacer pruebas que tal vez pasaron desapercibidas para el desarrollador original, así poder integrar nuevas ideas y conversar en aquellas instancias para solucionar los problemas que se presentaban.

Dentro de las versiones del sistema y aportes al flujo de trabajo podremos observar las siguientes imágenes:




**Imagen 1:** Primera versión del sistema para gestionar tareas subido al Github

Martes, 3 de septiembre ▾

-  **GitHub** APP 13:16  
1 new commit pushed to `main` by Rxrrx
- `14353c8f` - probando
-  Equipito6/Tarea1
-  **GitHub** APP 13:19  
Pull request opened by Rxrrx
- #1 Rama1**  
Se eliminaron archivos innecesarios.
-  Equipito6/Tarea1 | 3 sep.
- [Comment](#)
-  1 respuesta hace 6 días
-  **GitHub** APP 13:20  
3 new commits pushed to `main` by Rxrrx
- `0651a7f4` - Delete Operaciones.log
  - `40c23449` - Delete interfazMDB.py
  - `f62866a3` - Merge pull request #1 from Equipito6/Rama1
-  Equipito6/Tarea1
-  **GitHub** APP 13:20  
ha respondido [a un hilo](#)  
Pull request merged by Rxrrx
- #1 Rama1**  
 Equipito6/Tarea1 | 3 sep.

**Imagen 2:** Organización y creación de Branch para subir las nuevas versiones dentro del Github



GitHub

APP

22:20

Jueves, 5 de septiembre ▾


2 new commits pushed to `main` by Rxrrx


5734f835

 - Versión 2 de la tarea

c485f358

 - Merge pull request #6 from Equipito6/Rama1

 Equipito6/Tarea1



GitHub


APP

22:20


ha respondido [a un hilo](#)

Pull request merged by Rxrrx

#6 Versión 2 de la tarea

 Equipito6/Tarea1 | 5 sep.

**Imagen 3:** Sigüientes versiones de MyTask



GitHub


APP

12:02


Viernes, 6 de septiembre ▾

Pull request opened by diegomorales1


#7 Tercera version

 Equipito6/Tarea1 | 6 sep.

Comment

 1 respuesta

 hace 3 días



GitHub

APP

12:02


2 new commits pushed to `main` by diegomorales1


f3a5ff4d

 - Add files via upload

b9f0ffc3

 - Merge pull request #7 from Equipito6/Rama2

 Equipito6/Tarea1



GitHub

APP


12:07

Sábado, 7 de septiembre ▾


Pull request opened by Rxrrx

#8 Se sube la versión final


Se sube la versión final de la tarea.

 Equipito6/Tarea1 | 7 sep.

Comment

 1 respuesta

 hace 1 día



GitHub

APP

12:07


2 new commits pushed to `main` by Rxrrx


88163fe4

 - Se sube la versión final

f434112e

 - Merge pull request #8 from Equipito6/Rama1

 Equipito6/Tarea1



GitHub


APP

12:07

ha respondido [a un hilo](#)

Pull request merged by Rxrrx

#8 Se sube la versión final

 Equipito6/Tarea1 | 7 sep.

**Imagen 3:** Versión final de MyTask

## Problemas encontrados

Como podrán notar, las distintas versiones son la integración de nuevas funcionalidades, pero además pueden contar con que las pruebas unitarias que se realizaron son también soluciones implementadas y subidas en aquellas versiones, por ejemplo en las pruebas realizadas a las primeras versiones nos encontramos como con temas de que el filtro no contenía todos los casos de filtros que el cliente podría requerir, por lo tanto como solución fue implementar todos los casos que se requieran dentro de aquella funcionalidad. Otros problemas que nos encontramos y que dentro del equipo pasamos por alto fue que cada usuario fuera independiente para el sistema de gestionar las tareas, en otras palabras, si tenemos a Pepito y a Juanito, Pepito no debería de poder ver y gestionar las tareas de Juanito y viceversa, ya que, el código en sus inicios no hacia aquella diferencia, presentando todas las tareas de manera global independiente del usuario, claramente en la última versión se aplico una nueva variable para obtener el usuario y hacer diferencia entre las condiciones del sistema para verificar si el usuario es Juanito, mostrar aquellas tareas específicas de él. Y así sucesivamente teníamos problemas ligeros que nos pasaban por alto, pero dentro de las nuevas versiones siempre se podían resolver de manera óptima y rápida dentro del equipo. Dentro de las pruebas unitarias y el suite de casos se podrán observar aquellas instancias que tuvimos ciertos errores en los casos de prueba, para proponer soluciones y hacer cada vez un sistema mucho más completo y robusto para el cliente, siempre teniendo en mente el cumplimiento y funcionalidad para cada expectativa de este último, siendo practico y funcional en su totalidad.