UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Universidad Técnica Federico Santa María

Tarea 1: Mypass

https://github.com/tareas-pruebas-sw/tarea-1

Autores:
Ignacio Alvarado
Ian Cooper

Abril 2024

1 Validación

En esta sección se muestran las preguntas que permiten especificar de mejor forma el requerimiento.

1.1 Almacenamiento y gestión de contraseñas

1. ¿El programa estará en linea o funcionará de manera local en el computador del/los usuario/s?

   - El programa se ejecutará de manera local mediante la consola.

2. ¿Comó será el almacenamiento de las contraseñas? (local o en línea, Base de datos, archivo texto, etc)

   - Las contraseñas se deben almacenar en Mongodb Atlas (BD en la nube).

3. ¿Cómo se gestionará el acceso de distintos usuarios al programa?

   - El programa debe tener un sistema de inicio de sesión, dónde los usuarios podrán autenticarse y además se podrán crear usuarios con un nombre de usuario único y su contraseña, para luego acceder al programa. Las contraseñas del usuario se deben guardar hasheadas en SHA256.

4. ¿Qué medidas de seguridad se implementarán para garantizar que las contraseñas almacenadas estén protegidas?

   - Se debe utilizar un algoritmo de encriptación simétrica para almacenar las contraseñas y luego poder recuperarlas.

5. ¿Qué información adicional entregará el usuario al momento de agregar una contraseña?

   - El usuario entregará una palabra clave que será única para cada contraseña y esta contendrá letras (minúscula y/o mayúscula) y números.

6. ¿Cuál será la longitud máxima y mínima que tendrá la palabra clave?

   - Longitud mínima 4 y máxima 25 caracteres.

1.2 Generador de contraseñas

7. En el modulo de generación de contraseñas, ¿cuál será la longitud máxima y mínima que se permita especificar al usuario?

   - La longitud mínima será 8 y máxima 32 caracteres.

8. ¿Cuales serán los carácteres que se permitan elegir a los usuarios, al momento de generar una contraseña?

    - Todas las letras mayúsculas del alfabeto español. [A-Z]
    - Todas las letras minúsculas del alfabeto español. [a-z]
    - Números del 0 al 9. [0-9]
    - Carácteres especiales, se debe permitir elegir entre los siguientes: ! " # $ % & ' ( ) * + , - . / : ; ¡ = ¿ ? @ [ ] ^ _ ‘ { | } ~

2 Verificación

Para el proceso de verificación, generamos un conjunto de posibles entradas al programa, las cuales se encuentran en el excel de pruebas. Cabe mencionar que es imposible probar exhaustivamente cada posible entrada, entonces el conjunto de pruebas se hizo en base a las especificaciones del requerimiento, lo que nos permite asegurar que el programa cumple el requerimiento.

3 Organización

- Somos un equipo de 2 integrantes.
- Usamos slack como medio de comunicación.
- Se usó gitflow como flujo de trabajo para el repositorio de la tarea en github.
- La tarea de especificar mejor el requerimiento lo realizamos en conjunto en una reunión.
- En cuanto al desasarrollo del programa lo dividimos en 3 funcionalidades principales y las asignamos de la siguiente forma:
    - Funcionalidad 1 (Ignacio Alvarado): Almacenar y gestionar contraseñas, lo que incluye crear, recuperar, actualizar y eliminar contraseñas.
    - Funcionalidad 2 (Ian Cooper): Generar contraseñas seguras.
    - Funcionalidad 3 (Ian Cooper): Administración de usuarios, lo que incluye creación de cuentas de usuario e inicio de sesión de usuarios.
- Se definió la siguiente estrategia de pruebas:
    - Las pruebas al programa se realizarán de forma manual mediante la consola del programa.
    - Ignacio Alvarado realizó pruebas a las funcionalidades implementadas por Ian Cooper y viceversa.

- Primero realizamos pruebas de manera individual.
- Luego se arregló el código en base a las pruebas fallidas.
- Luego de realizar las pruebas de manera individual, se consolidaron las pruebas realizadas por cada uno en un conjunto único y se realizaron en conjunto en una reunión, dónde también se discutió si faltaba alguna prueba.

4 Flujo de trabajo y configuraciones

4.1 Slack

Se creó un espacio de trabajo en Slack, esto se puede observar en la Figura 1.

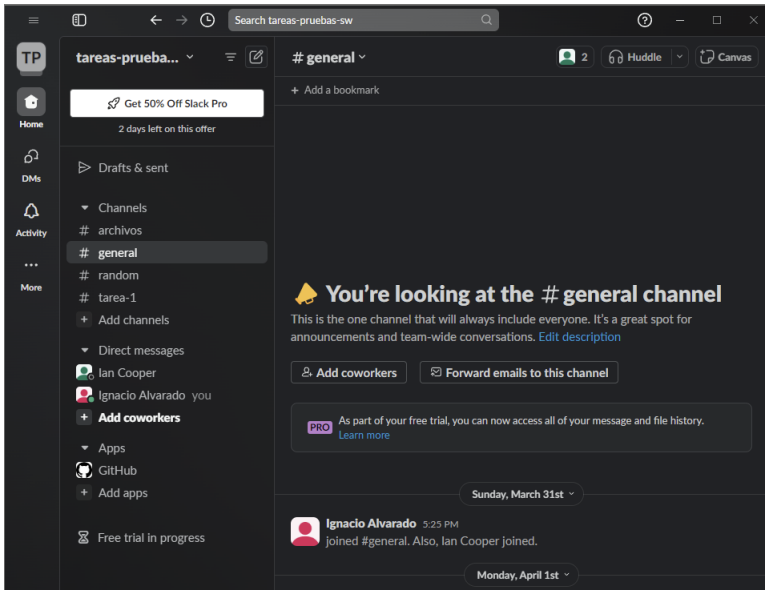

Figure 1: Slack Workspace

4.2 Slack + Github

Se integró el espacio de trabajo de slack con el repositorio de github. Integramos el canal tarea-1 con github y luego nos suscribimos al repositorio. Esto se puede observar en la Figura 2.

4.3 Gitflow

Se utilizó el flujo de trabajo Gitflow para trabajar en el repositorio. En la Figura 3 se puede observar las ramas main, develop y 3 ramas feature. La

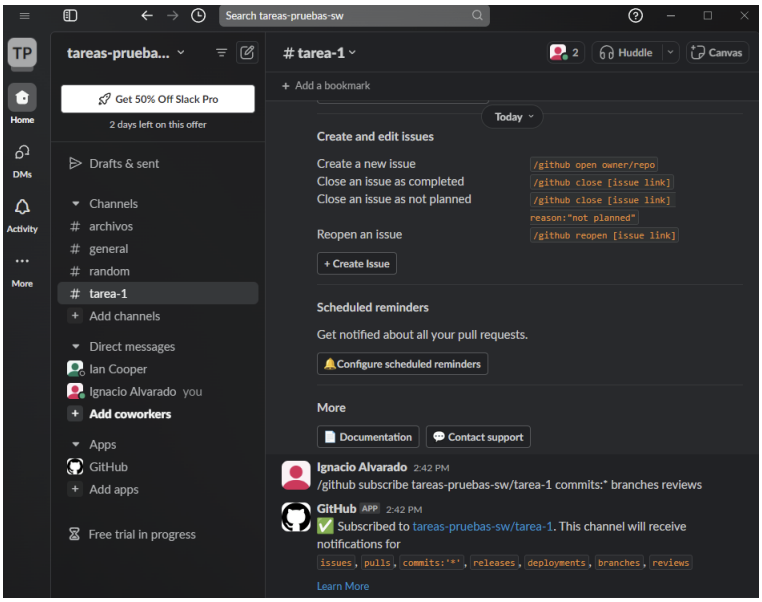

Figure 2: Slack + Github

rama feature/password-manager fue usada para crear la funcionalidad de agregar, actualizar, recuperar y eliminar contraseñas. La rama feature/password-generator fue usada para crear la funcionalidad de generar contraseñas. La rama feature/password fue usada para crear la funcionalidad de crear usuarios y autenticar usuarios. Cabe recalcar que también hicimos uso de ramas bugfix para arreglar bugs, todo esto se puede comprobar en los PR del repositorio.

4.4 Protección de ramas

Se protegió las ramas develop y main, para que ambas requieran un pull request antes de hacer merge con estas. En las Figuras 4 y 5.

4.5 Revisión/Aprobadores

Se estableció que si un integrante hace un pull request a develop para fusionar una feature a develop, este debe ser revisado y aprobado por el otro integrante del equipo.

5 Problemas

- **Gitflow Feature:** La rama develop tenia una regla de protección que imponia hacer un pull request antes de hacer un merge. Al momento de terminar una de las features, utilizamos el comando “git flow finish feature

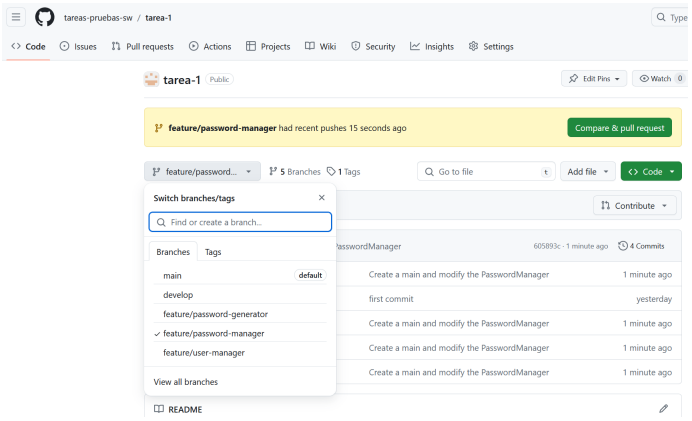

Figure 3: Gitflow

branch" y al momento de hacer un push a develop aparecia un error, ya que la rama develop estaba protegida. Para solucionar el problema se eliminó la regla de protección y se realizó el push. Consideramos que una mejor forma de trabajar es solicitar un pull request al querer hacer un merge a develop para que el otro integrante del equipo revise el código antes de subirlo a develop, entonces luego volvimos a añadir la regla de protección para que en cambios futuros se haga un pull request.

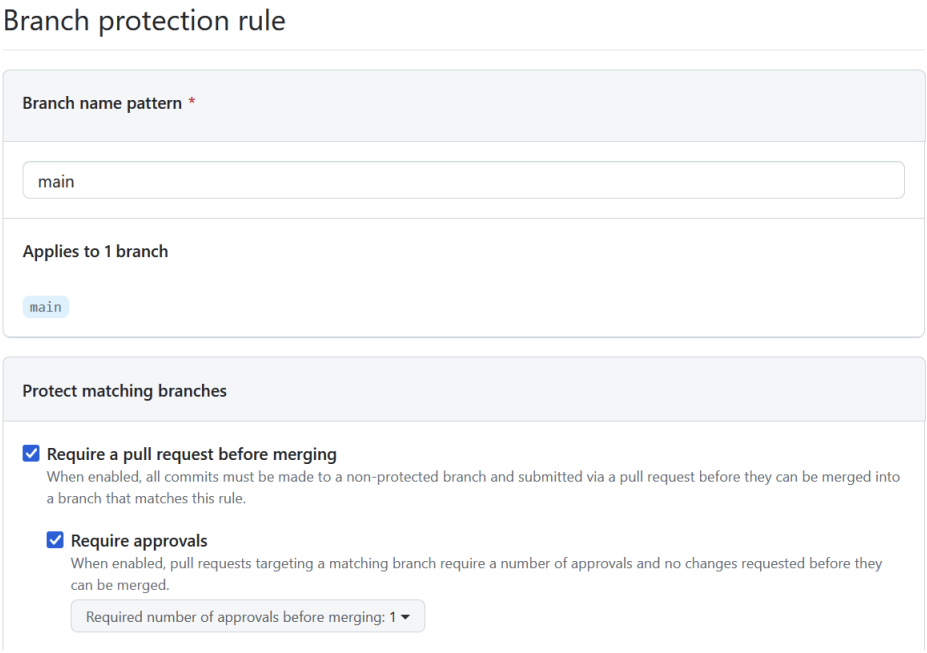

Figure 4: Protección rama main.

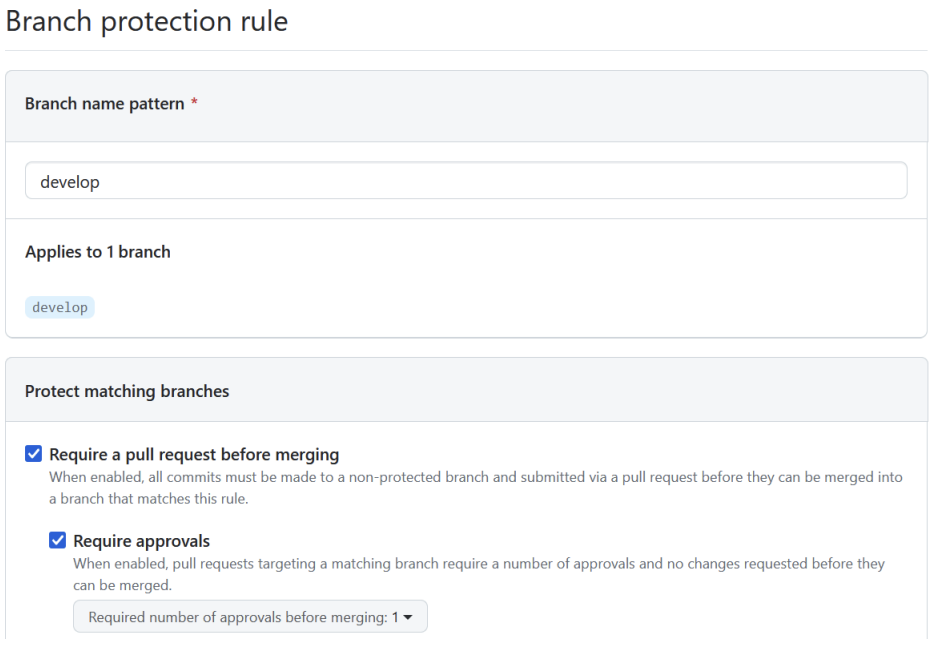

Figure 5: Protección rama develop.