**Documentación Proyecto**

**Plataforma de Mensajería Electrónica**

**NRC:** 2092

**Equipo No.** 10

**Integrantes:**

Javier Camayo Caicedo

Juan Ríos Gómez

Jairo Rivera Zapata

José Jaramillo Castaño

Daniel Lozano Ospino

**Descripción de roles del equipo**

|  |  |  |  |
| --- | --- | --- | --- |
| Rol | Integrante | Descripción | Tareas |
| SCRUM Master | Tutor | Persona encargada de llevar el seguimiento del proyecto. | Seguimiento del correcto desarrollo de las tareas del proyecto. |
| Desarrollador Front-End | Integrante 1 | Persona encargada de realizar la parte visual de la aplicación. | Desarrollo de las vistas y estilos de la aplicación. |
| Desarrollador Back-End | Integrante 2 | Persona encargada de realizar la lógica de negocios de la aplicación. | Desarrollo de los controladores de la aplicación y diseño e integración de la base de datos. |
| Desarrollador Back-End | Integrante 3 | Persona encargada de realizar la lógica de negocios de la aplicación. | Desarrollo de los controladores de la aplicación y diseño e integración de la base de datos. |
| Desarrollador Full-Stack | Integrante 4 | Persona encargada de realizar la parte visual y la lógica de negocios de la aplicación. | Desarrollo de las vistas y estilos de la aplicación, de los controladores de la aplicación y diseño e integración de la base de datos. |

**Definición de artefactos**

**Backlog Sprint 1**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación | Responsable |
| Definición de roles | Definición de los roles de los integrantes del equipo de trabajo del proyecto. | 2 horas | Todos los integrantes |
| Definición de artefactos | Definición de los artefactos de la metodología SCRUM para el desarrollo del proyecto. | 4 horas | Todos los integrantes |
| Diseño del diagrama de clases | Diseño del diagrama de clases de la aplicación a desarrollar. | 2 horas | Todos los integrantes |
| Definición del cronograma | Definición del cronograma de actividades del proyecto. | 2 horas | Todos los integrantes |

**Backlog Sprint 2**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación | Responsable |
| Definición del mapa de navegabilidad | Definición del mapa de navegabilidad que mostrará la distribución de las vistas de la aplicación | 3 horas | Integrante 1 y 4 |
| Selección de la plantilla de estilos | Selección de la librería CSS a usar para los estilos de las vistas de la aplicación. | 1 hora | Integrante 1 y 4 |
| Diseño e implementación de las vistas | Diseño e implementación de las vistas en HTML y CSS. | 15 horas | Integrante 1 y 4 |
| Creación del proyecto en GIT | Creación del proyecto en GIT y posterior cargue a la nube en la plataforma Github. | 1 hora | Integrante 1 |

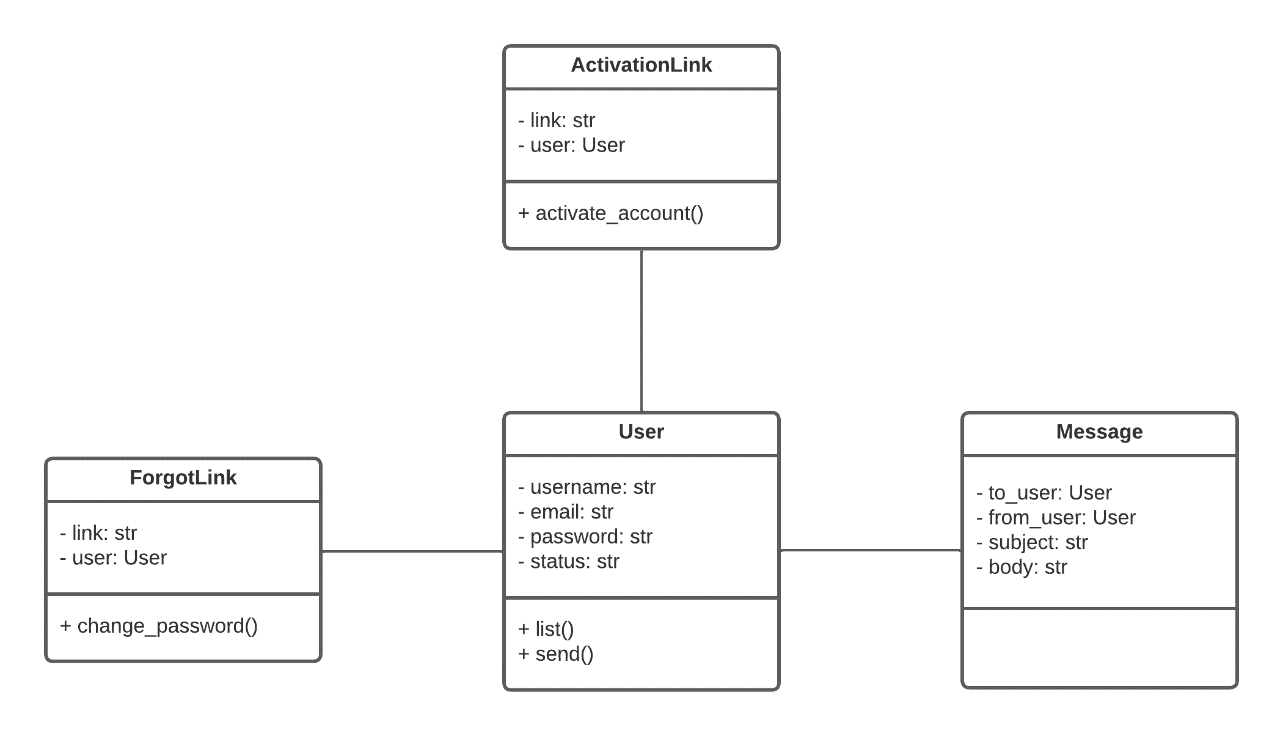
**Backlog Sprint 3**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación | Responsable |
| Diseño e implementación de los controladores para formularios y otras funcionalidades | Diseño e implementación de los controladores que representan la lógica de negocios de la aplicación. | 40 horas | Integrante 2 y 3 |
| Diseño e implementación de base de datos | Diseño de la base de datos relacional que alojará los datos de la aplicación e implementación de la misma usando el motor SQLite. | 10 horas | Integrante 4 |
| Desarrollo de integración de controladores y bases de datos | Integración de la base de datos con los controladores para la búsqueda y almacenamiento de información persistente de manera segura. | 30 horas | Integrante 2 y 3 |
| Diseño e implementación de portal de acceso usando método de autenticación basado en usuario y contraseña | Diseño e implementación del método de autenticación de los usuarios y manejo de sesiones. | 20 horas | Integrante 4 |

**Backlog Sprint 4**

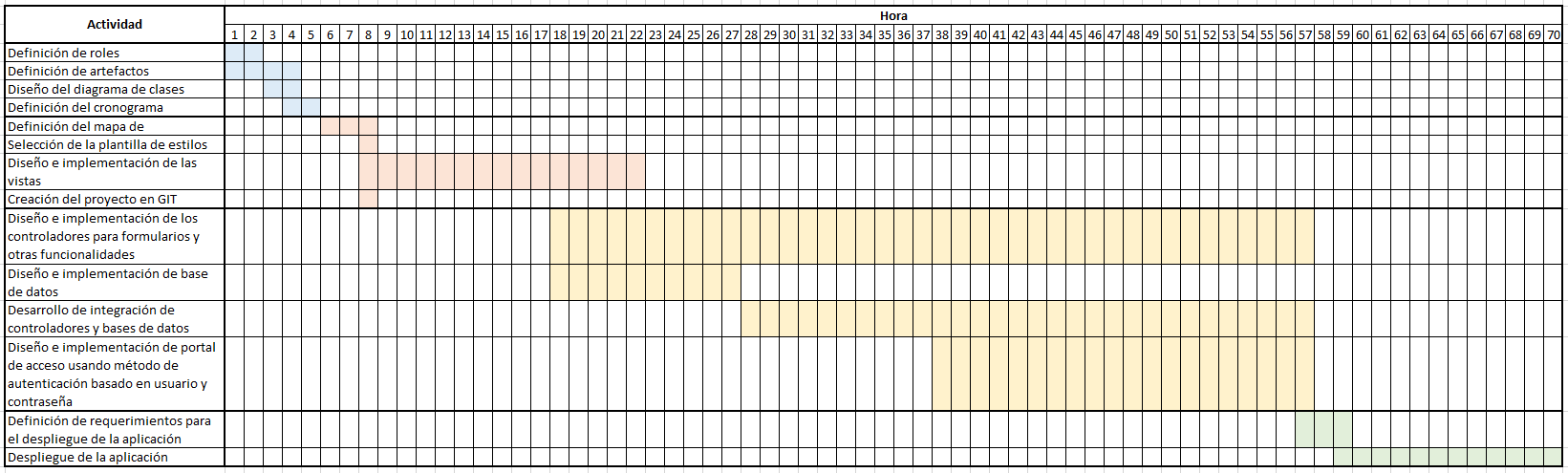
|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación | Responsable |
| Definición de requerimientos para el despliegue de la aplicación | Definición de los requerimientos necesarios para realizar el despliegue de la aplicación en la plataforma PythonAnywhere | 3 horas | Todos los integrantes |
| Despliegue de la aplicación | Configuración, despliegue y verificación del funcionamiento de la aplicación en la plataforma PythonAnywhere | 12 horas | Todos los integrantes |

**Diagrama de clases**

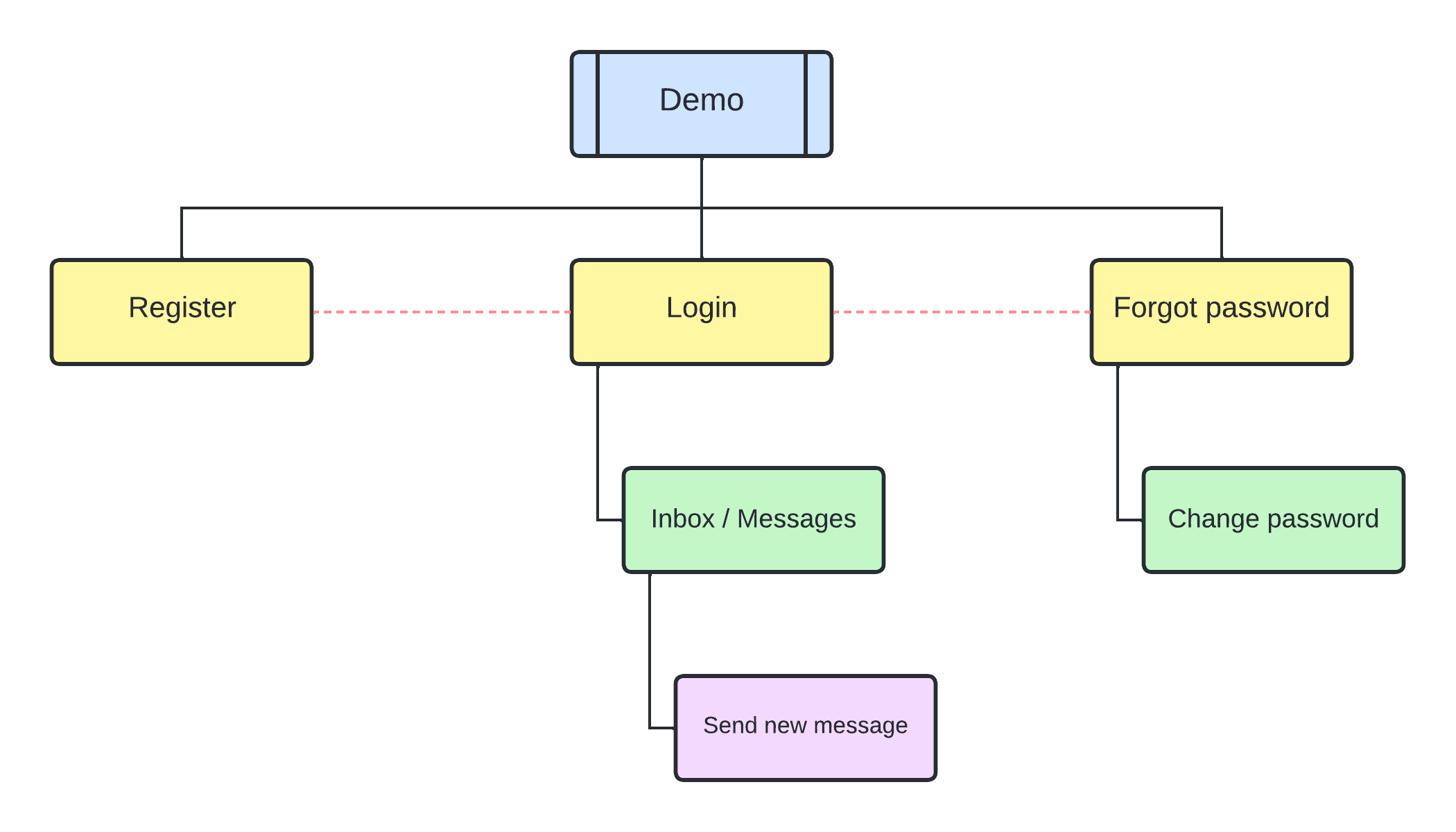


La clase usuario tendrá la información referente al usuario, es decir, su nombre de usuario, su correo, su contraseña y el estado de su cuenta. La clase mensaje tendrá la información de los mensajes de la plataforma, es decir, el usuario que envía el mensaje, el usuario que lo recibe, el asunto y el cuerpo del mensaje.

**Cronograma de tareas**



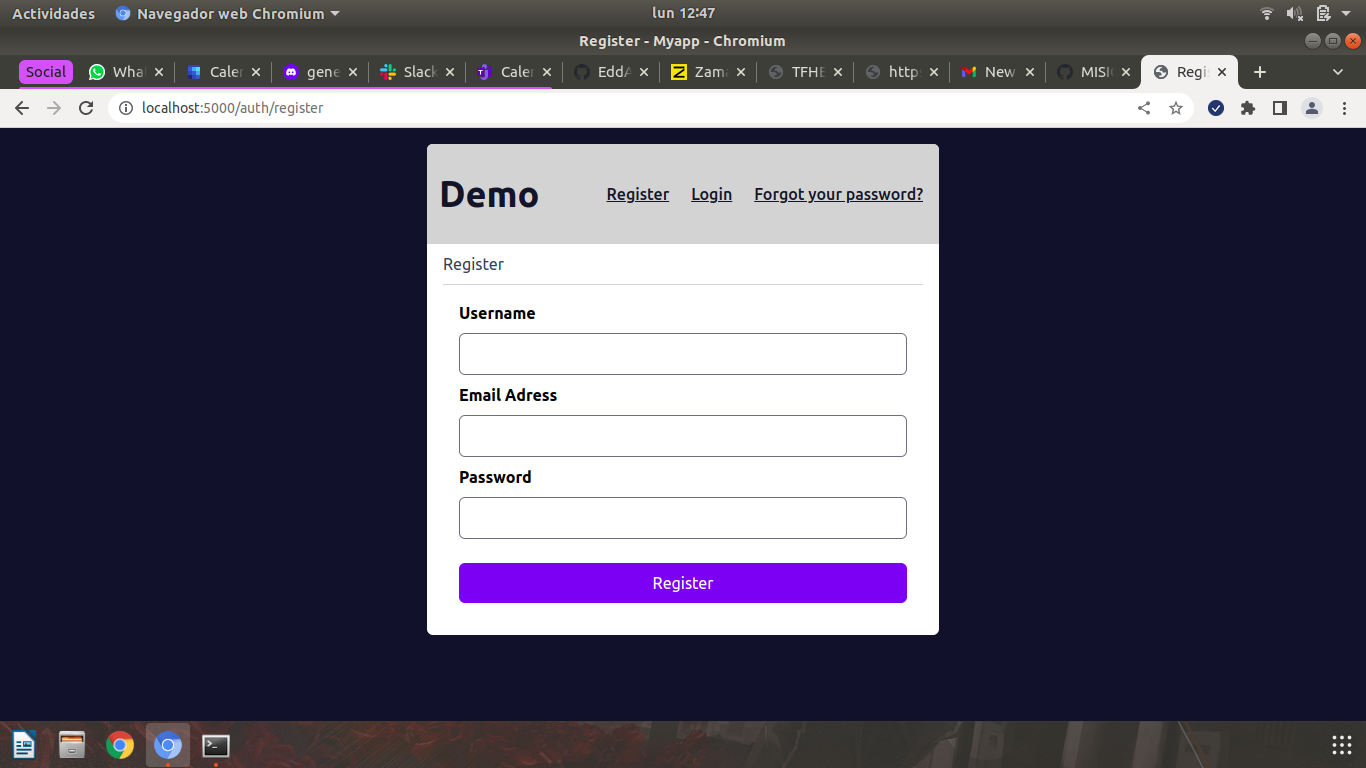
**Mapa de navegabilidad**



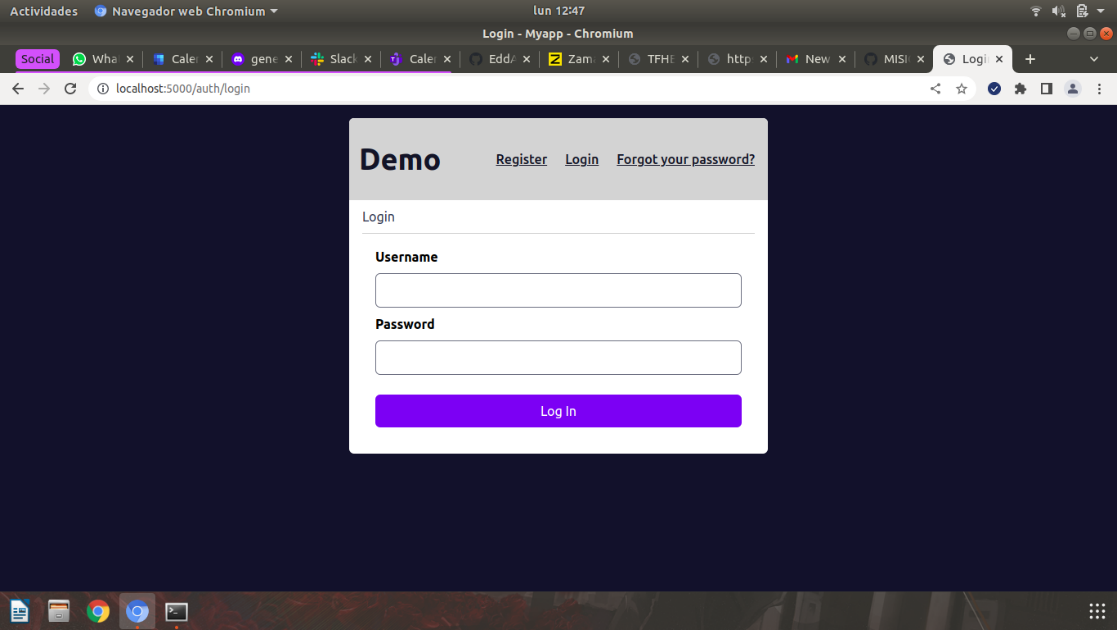
La aplicación en su pantalla principal tiene 3 opciones, estas son el registro, el inicio de sesión y la recuperación de contraseña, y se puede desplazar entre ellas. La pantalla de recuperación de contraseña, dará paso a la pantalla de cambio de contraseña al acceder a ella mediante el enlace enviado. La pantalla de inicio de sesión da paso a la de visualización de mensajes y esta a su vez da paso a la pantalla de envío de nuevo mensaje.

**Vistas de la aplicación**

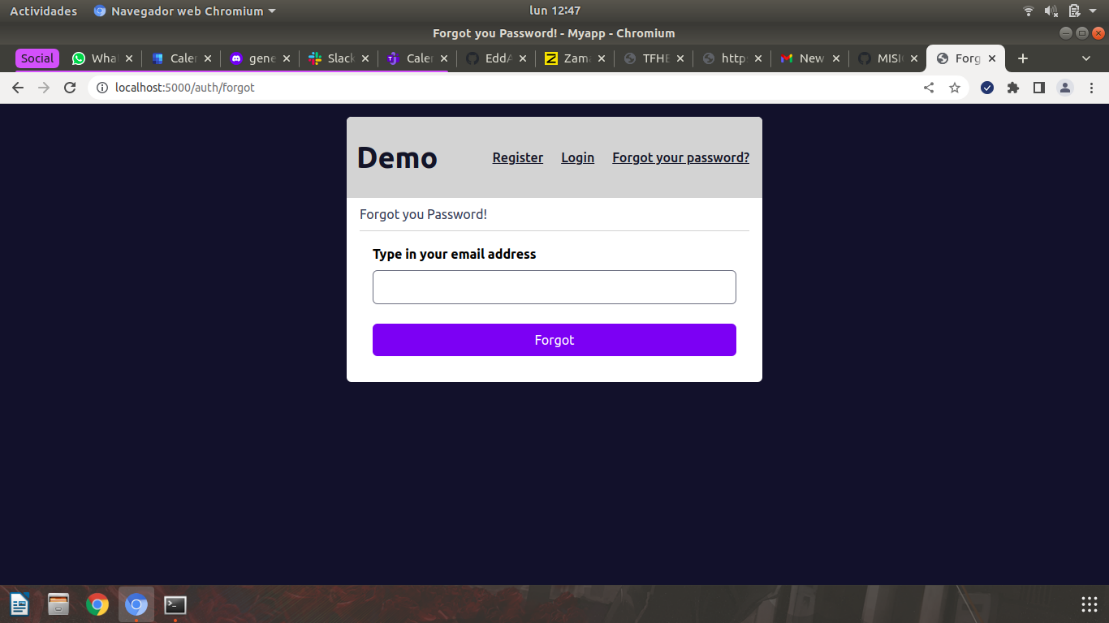
* **Vista de registro:**



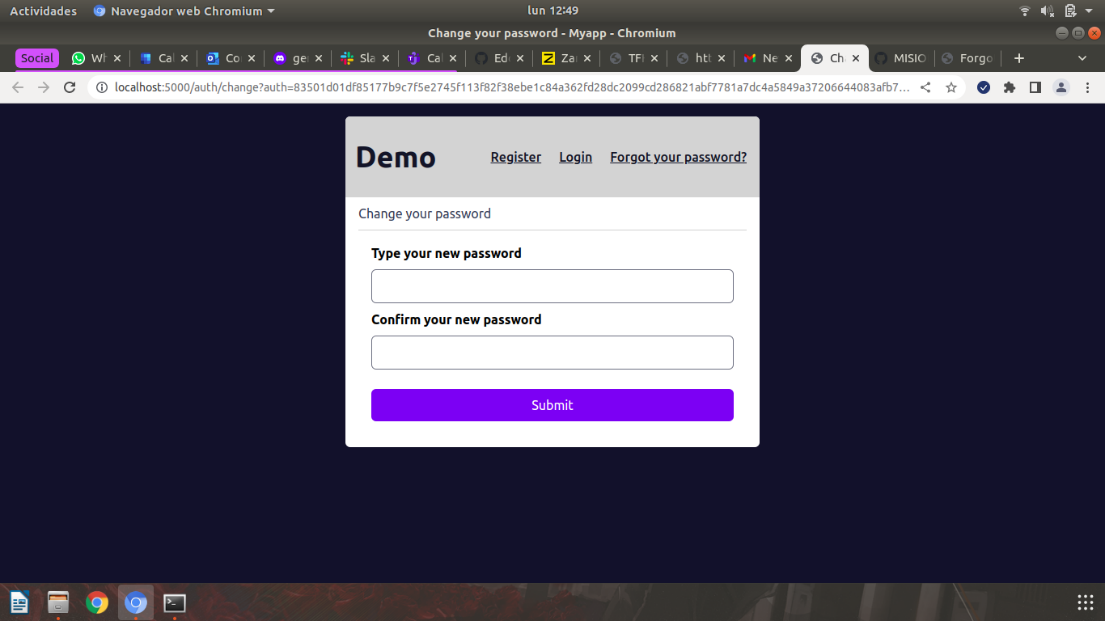
* **Vista de inicio de sesión:**



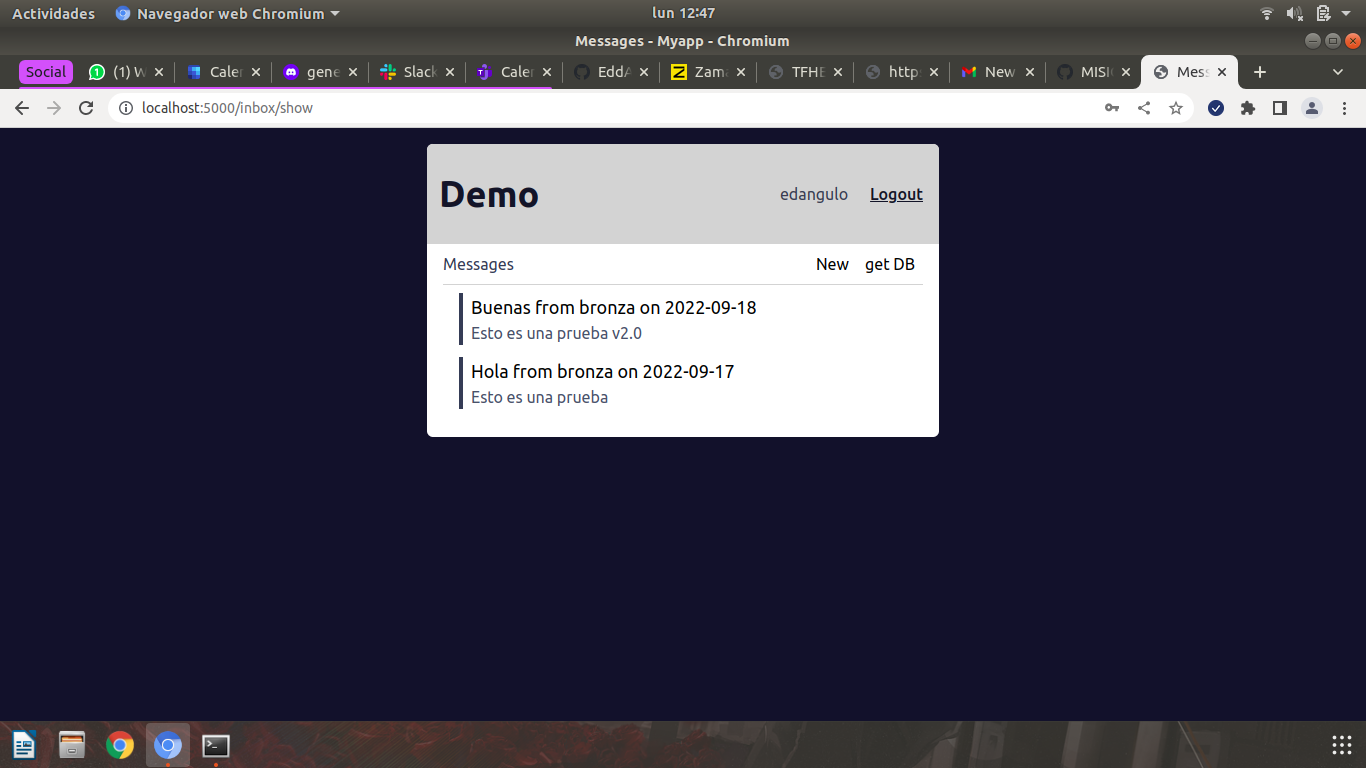
* **Vista de solicitud de recuperación de contraseña:**



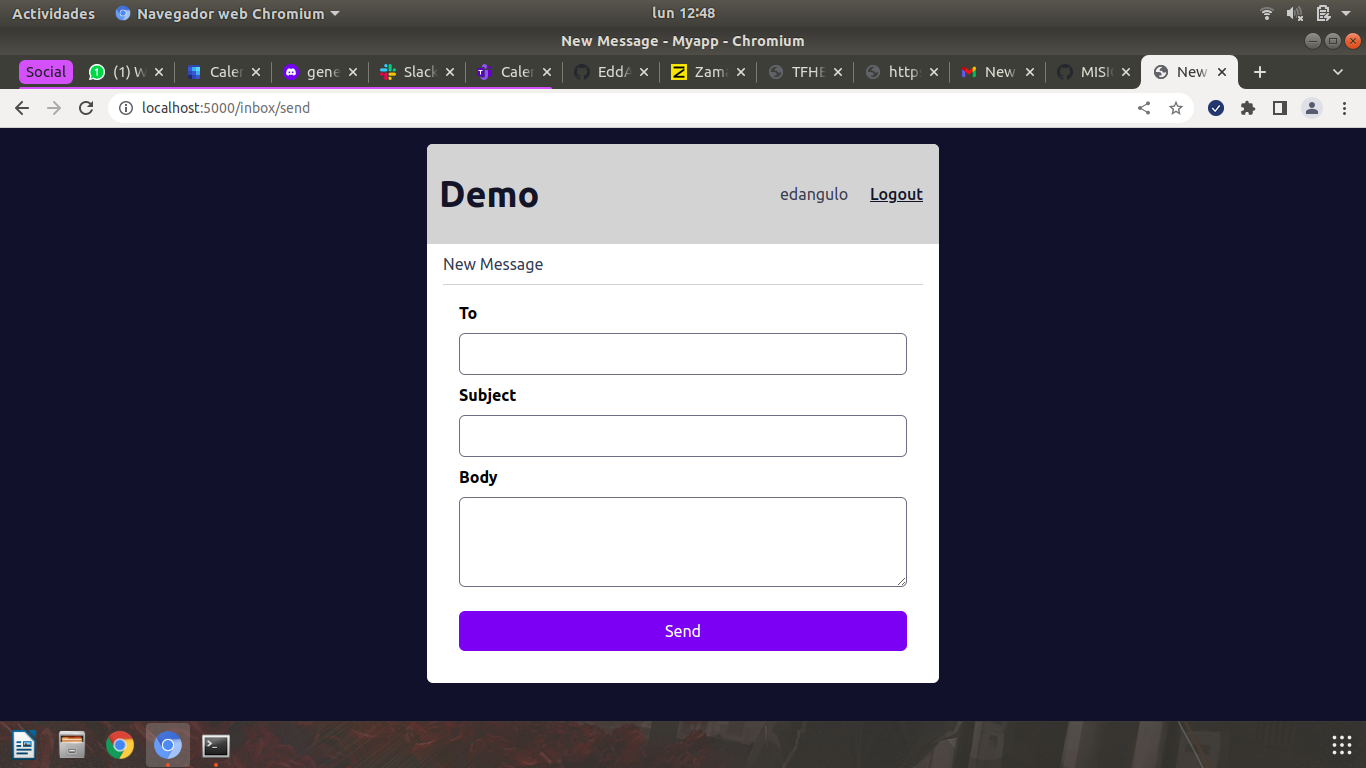
* **Vista de cambio/recuperación de contraseña:**



* **Vista de visualización de mensajes:**



* **Vista de envío de un nuevo mensaje:**



**Implementación Métodos Controladores**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Especificación Ruta | Métodos HTTP | Lógica Algorítmica |
| activate | /auth/actívate  (/auth/login) | GET, POST | Es donde se Activa la Cuenta del Usuario para ingresar a la plataforma |
| register | /auth/register  (/auth/login) | GET, POST | Genera el enlace de activación en el correo personal de usuario para pueda activar su cuenta |
| confirm | /auth/confirm  (/auth/forgot) | GET, POST | Permite el cambio de Contraseña del usuario |
| change | /auth/change  (/auth/forgot) | GET, POST | Controla la Vista para el cambio de Contraseña |
| forgot | /auth/forgot | GET, POST | Genera el enlace para realizar el cambio de Contraseña del usuario |
| login | /auth/login | GET, POST | Se realiza el Inicio de sesion del Usuario |
| show | /inbox/show | GET, POST | Muestra los mensajes que pertenecen al usuario |
| send | /inbox/send | GET, POST | Permite enviar mensajes a otros usuarios |

**Diagrama Relacional Base de Datos**

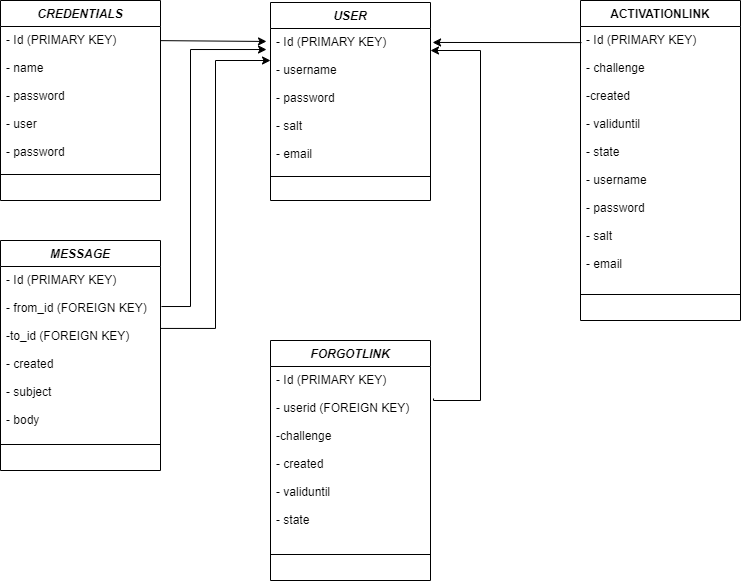


Tabla user: Tendrá los datos más importantes del usuario: nombre, email y las contraseñas de seguridad.

Tabla forgotlink: Tendrá los datos necesarios para la creación de los enlaces de activación de recuperación contraseña.

Tabla activationlink: Tendrá los datos necesarios para la creación de los enlaces de activación de las cuentas.

Tabla message: Tendrá los datos necesarios para la consulta de los mensajes que se envían entre usuarios.

Tabla credentials: Tendrá las credenciales del correo administrador desde donde se enviarán los enlaces de activación y recuperación de contraseña.

**Prácticas de Programación Segura**

Cualquier página web es susceptible ser atacada, por este motivo es importante que las páginas no sean únicamente visualmente atractivas, sino que también estén desarrolladas y diseñas de forma segura. Igual sucede con las Bases de datos.

En este caso se utilizaron las funciones hash y salt para mantener la seguridad en la base de datos, evitando que se puedan descifrar, inclusive si alguien lograra acceder a la base de datos.

**Direccion URL GitHub del Proyecto**

[**https://github.com/Utherloki/Mensajeria-Stark**](https://github.com/Utherloki/Mensajeria-Stark)