

INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO



Implementación de mensajería en gRPC

Sistemas Distribuidos

SANTIAGO HERNÁNDEZ 180151

Introducción:

El presente informe describe el proyecto TurboMessage, un sistema de mensajería desarrollado para facilitar la comunicación entre usuarios. Este sistema permite el envío, recepción, lectura y eliminación de correos electrónicos. El objetivo principal de TurboMessage es proporcionar una plataforma rápida y eficiente para la gestión de correos electrónicos entre usuarios registrados.

Arquitectura del Proyecto

El proyecto TurboMessage sigue una arquitectura cliente-servidor. A continuación, se describirán los componentes y la interacción entre ellos.

1. Cliente: El cliente de TurboMessage es la interfaz con la que los usuarios interactúan para utilizar el sistema de mensajería. Está implementado en Python y ofrece una interfaz de línea de comandos (CLI) para facilitar la interacción del usuario. El cliente se conecta al servidor utilizando gRPC y realiza solicitudes al servidor para llevar a cabo operaciones como el envío, lectura y eliminación de correos electrónicos. El cliente también maneja las respuestas del servidor y proporciona retroalimentación al usuario.
2. Servidor: El servidor de TurboMessage es el núcleo del sistema y se encarga de procesar las solicitudes del cliente. Está implementado en Python y utiliza gRPC para recibir las solicitudes y enviar las respuestas correspondientes. El servidor gestiona una base de datos de usuarios y correos electrónicos. Al recibir una solicitud de inicio de sesión, el servidor verifica las credenciales del usuario y autentica al cliente. Una vez autenticado, el servidor permite al cliente realizar operaciones como enviar, recibir, leer y eliminar correos electrónicos. El servidor actualiza la base de datos de acuerdo con las operaciones solicitadas.
3. Protocolo de Comunicación: El protocolo de comunicación entre el cliente y el servidor se define en el archivo `turbomessage.proto`. Este archivo especifica los mensajes y servicios necesarios para la comunicación y utiliza la estructura de datos de protocolo buffers para serializar los mensajes. Los mensajes definidos en el archivo `turbomessage.proto` incluyen información como el nombre de usuario, contraseña, identificador de correo, remitente, destinatario, asunto y cuerpo. Los servicios definidos en el archivo `turbomessage.proto` representan las operaciones que el cliente puede realizar en el servidor, como inicio de sesión, envío de correos electrónicos, lectura de correos electrónicos y eliminación de correos electrónicos.

Parte 3: Desafíos y Mejoras Potenciales

1. Manejo de errores: Es importante tener un manejo adecuado de errores en el sistema para garantizar una experiencia de usuario fluida. En la implementación actual, los errores se informan al cliente mediante mensajes de respuesta. Sin embargo, se puede

mejorar el manejo de errores al proporcionar códigos de error más descriptivos y claros para que el usuario comprenda rápidamente la causa del problema y tome las medidas necesarias.

2. Funcionalidades adicionales: El proyecto TurboMessage actualmente admite operaciones básicas de envío, recepción, lectura y eliminación de correos electrónicos. Sin embargo, se pueden agregar funcionalidades adicionales para mejorar la experiencia del usuario. Algunas mejoras potenciales incluyen la capacidad de marcar correos electrónicos como favoritos, organizarlos en carpetas personalizadas, buscar correos electrónicos por palabra clave y establecer notificaciones por correo electrónico.
3. Interfaz de usuario mejorada: Aunque el cliente actual utiliza una interfaz de línea de comandos (CLI) simple, se pueden explorar opciones para desarrollar una interfaz de usuario más amigable y atractiva. Esto podría incluir el desarrollo de una aplicación de escritorio con una interfaz gráfica de usuario (GUI) intuitiva y fácil de usar.

En resumen, el proyecto TurboMessage ha logrado implementar un sistema de mensajería funcional utilizando una arquitectura cliente-servidor basada en gRPC. Aunque el sistema actual cumple con los requisitos básicos de envío, recepción, lectura y eliminación de correos electrónicos, hay varias áreas de mejora que se pueden explorar.

```
welcome santiago
Inbox:
Sent Mail:
Actions to Perform:
1. Compose a new email
2. Read an email
3. Delete an email
4. Refresh inbox
5. Exit TurboMessage
Enter the number of the action you want to perform ->
```