

Modelado y Programación

Proyecto 1: Chat

García Pérez Adrián

Descripción del proyecto

El proyecto consiste en la creación de un chat usando el protocolo de comunicación por red TCP, aplicando los conocimientos adquiridos sobre buenas prácticas de programación, lenguajes de programación y programación concurrente.

Dentro del chat, es posible levantar un servidor, el cuál aceptará las conexiones de los clientes que deseen participar en el mismo. Así, los clientes serán capaces de comunicarse con todos los miembros del chat, enviando mensajes públicos, privados, creando salas privadas y definiendo un estado.

Lenguaje de programación

El proyecto se desarrolló usando el lenguaje de programación Rust, un lenguaje compilado, de propósito general y multiparadigma, habitualmente usado para el desarrollo de software de sistema.

Decidí usar Rust porque me pareció un reto, ya que no tenía idea de cómo funcionaba, no conocía sus ventajas ni sus desventajas. En particular, me pareció buena idea usarlo debido a que es compilado y a que Rust es bastante estricto en ese aspecto.

Por último, pensé que era una buena opción para tener un primer acercamiento al desarrollo de software de sistema, a pesar de realizar un proyecto de software de aplicación.

Interfaz gráfica

La interfaz gráfica fue hecha utilizando la biblioteca Gtk-rs, la cual fue elegida debido a que nos permite definir la interfaz gráfica con archivos creados en Glade, un diseñador de interfaces gráficas. De esta forma, la atención se ve enfocada en definir el comportamiento de la interfaz al interactuar con el usuario.

Diseño

El proyecto se estructuró en 2 componentes principales: los archivos ejecutables y el código de biblioteca.

En pocas palabras, los ejecutables hacen uso del código de biblioteca (que es donde se realizó la mayor parte de programación), para crear una interfaz gráfica que permita usar el chat desde el lado del cliente.

En el código de biblioteca, se definieron distintas estructuras fundamentales: el servidor, los clientes, las salas y los eventos del protocolo.

Dichos componentes interactúan entre sí de forma que permiten levantar el servidor donde se aceptan las conexiones, permitiendo así la comunicación entre los clientes.

Además, el servidor cuenta con la posibilidad de crear una bitácora (*aka log*) sobre los distintos eventos ocurridos durante su ejecución, esto con el fin de observar como se comporta el servidor al interactuar con los usuarios y, si es el caso, depurar errores del mismo al ver su comportamiento en la bitácora.