

Proyecto N°1: Sockets

Profesor: Cesar Azurdia
Auxiliares: Giovanni Castiglioni, Camilo Fredes

Ayudantes: Vicente Aitken, Martín Moreno, Ariel Núñez, Simón Repolt

Quizás la aplicación más conocida de los sockets TCP/IP es el chat. En esta tarea se tiene por objetivo crear un chat de servicio al cliente de la página de un banco. La idea es que cada usuario posea una cuenta personal en el banco asociada a un identificador único y una cantidad de dinero con el que realizar transacciones. El chat posee tres actores: los clientes, los ejecutivos y el servidor.

Terminal del cliente

1. Al ingresar al chat, el cliente debe autenticarse mediante un RUT y una contraseña. El cliente debe ingresar sus datos y el servidor lo identificará (en caso contrario, le pedirá que reingrese un RUT válido). Además, cada cliente debe poseer un saldo inicial en su cuenta bancaria.
2. El cliente deberá escoger entre distintos servicios posibles que ofrece el banco. Por ejemplo:

¡Bienvenido a la plataforma de servicio al cliente del Banco Zembe!
Para autenticarse debe ingresar su RUT y contraseña.

usuario: 11.420.420-k
contraseña: *****

Asistente: ¡Bienvenido Ariel! ¿En qué te podemos ayudar?

- [1] Cambio de contraseña.
- [2] Realizar transferencia.
- [3] Consulta de saldo.
- [4] Historial de operaciones.
- [5] Contactarse con ejecutivo.
- [6] Salir.

Ariel: 3

Asistente: Su saldo actual es de 10 Zembe Coins.

Asistente: ¿Desea realizar otra operación?

Ariel: 1

Asistente: Ingrese su nueva contraseña.

Ariel: we_ballin

Asistente: Ingrese su nueva contraseña nuevamente.

Ariel: we_ballin

Asistente: Su clave ha sido actualizada exitosamente.

Asistente: ¿Desea realizar otra operación?

...

3. El cliente puede realizar una transferencia a cualquiera de los otros clientes existentes en la base de datos del banco, independientemente de si estos están conectados o no. Para transferir deben confirmar ingresando su contraseña.
4. Si el cliente desea contactarse con un ejecutivo, quedará en una cola de espera hasta que un ejecutivo pueda atenderle y se le indicará el tiempo estimado de espera.

Terminal del servidor

1. El servidor debe ser capaz de aceptar al menos siete clientes y tres ejecutivos distintos al mismo tiempo. Es decir, un total de diez usuarios simultáneos.
2. El servidor debe poseer una base de datos con la información de los clientes y los ejecutivos. Para los clientes se deben considerar al menos el RUT, la contraseña, los usuarios a quienes ha enviado dinero, sus operaciones y el saldo de su cuenta. Para los ejecutivos sólo se necesitan su RUT y contraseña.
3. Se debe mostrar un mensaje en el terminal del servidor cada vez que un cliente se conecta/-desconecta, cuando un cliente es redirigido a un ejecutivo o cuando el usuario realiza alguna acción. Por ejemplo:

...

[SERVIDOR] Cliente Foto Metro conectado.

[SERVIDOR] Cliente Nomón desconectado.

[SERVIDOR] Ejecutivo Métanse a TICs conectado.

[SERVIDOR] Cliente Banda Angosta redirigido a ejecutivo Luz Visible.

[SERVIDOR] Cambio Clave Cliente Radio Frecuencia.

...

4. El servidor debe crear un registro de todas las solicitudes realizadas por los clientes.
5. Si es la primera vez que un cliente realiza una transferencia a otro, esta no puede exceder un determinado monto.
6. Si un cliente pide ver su historial de operaciones, el servidor mostrará las últimas cinco operaciones. Sin embargo, si el cliente se contacta con el ejecutivo, este podrá mostrarle la totalidad de sus operaciones pasadas.

Terminal del Ejecutivo

1. Al igual que los clientes, los ejecutivos deben autenticarse con un RUT y contraseña. Al autenticarse se les asigna un perfil de atención donde se les muestra la cantidad de clientes conectados.
2. Los ejecutivos deben poseer al menos los siguientes comandos para interactuar con el servidor y los clientes:
 - **:status:** Muestra la cantidad de clientes conectados y solicitudes de conexión con ejecutivos.
 - **:details:** Muestra los clientes conectados y su última acción.
 - **:history:** Muestra el historial del cliente que está atendiendo: cambios de contraseña, transferencias realizadas, consultas de saldo, solicitudes de su historial de operaciones y derivaciones a ejecutivos.
 - **:operations:** Le muestra al cliente TODO su historial de operaciones (movimientos realizados).
 - **:connect:** Se conecta con el primer cliente que se encuentre en la cola.
 - **:disconnect:** Termina la conexión con el cliente.
 - **:exit:** El ejecutivo se desconecta del servidor.
3. La terminal del ejecutivo podría verse de la siguiente manera:

¡Bienvenido a la plataforma de servicio al cliente del Banco Zembe!
Para autenticarse debe ingresar su RUT y contraseña.

usuario: 13.131.313-1
contraseña: *****

Asistente: Hola Giovanni, en este momento hay 4 clientes conectados.

Giovanni: :details

Asistente: Clientes conectados:

10.000.000-0 Foto Metro
13.456.123-7 Banda Angosta
11.420.420-k Radio Frecuencia
45.234.678-1 Nomón

El cliente Nomón se quiere conectar.

Giovanni: :connect

...

Una vez implementado el chat de servicio al cliente, se debe entregar un reporte que contenga:

1. Introducción.
2. Marco teórico que al menos debe responder:
 - ¿Qué es un socket?
 - ¿Cómo funciona un socket?
 - ¿Qué tipo de socket se utilizó en la implementación y cuáles son sus características?
3. Descripción de la implementación del servidor. Pueden explicar las partes más relevantes del código mostrando partes del mismo.
4. Discusión y conclusiones.

Entregables: Códigos funcionales de cliente, servidor y ejecutivo, reporte, y un breve video de ejemplo mostrando la funcionalidad del código.

BONUS: Podrán optar a 5 décimas extra quienes implementen un servidor capaz de crear una cuenta nueva para usuarios no identificados en la base de datos del banco.