

Manual de usuario

Aplicación con login para mandar incidencias con interfaz de usuario.

A pesar de que comencé a preparar una base de datos para almacenar la información, al final me decanté por guardar solo un hashmap con los usuarios y un índice para las incidencias quedando estas en el Servidor e inicializandose cada vez que se ejecuta la parte del servidor.

Para registrarse hay que llenar todos los campos del formulario, donde el usuario tendrá un máximo de 20 caracteres, el nombre de 50 y el apellido de 80. La edad está delimitada en el spinner siendo el rango de valores posibles [16, 130] años. Tanto el email como la contraseña están controladas con patrones (obtenidos en la página de regex 101). El email debe tener un @ y un . y entre ellos debe haber caracteres. La contraseña debe ser de 6 a 12 caracteres y tener minúsculas, mayúsculas, números y caracteres especiales.

Al intentar registrarte si no pasas la validación, te muestra un mensaje con los requisitos.

Se usa el puerto 6001 para la comunicación y sockets SSL, habiendo almacenado el certificado para esto en la carpeta certificados.

Al establecer la conexión el servidor y el cliente generan e intercambian la parte pública de las claves asimétricas y el servidor le devuelve la clave simétrica cifrada con la clave asimétrica. Y a partir de aquí toda comunicación se cifra con la clave simétrica.

Al final tuve que usar la clave simétrica porque la asimétrica tenía más limitaciones en cuanto a la longitud de los mensajes.

Para la comunicación entre el servidor y el cliente he creado una clase para poder mandar la información de una sola vez. Con el tipo de mensaje que serían las acciones que puede hacer el servidor. Para poder cifrar/descifrar el objeto necesitaba poder pasarlo a byte[], para eso he usado una librería (<https://www.baeldung.com/java-commons-lang-3>) que me permite serializar y deserializar objetos.

Me dio problemas cómo enviaba y recibía información con ObjectOutputStream y ObjectInputStream, mayormente porque cerraba sin querer uno de los sockets o por que se corrompía el stream y no era capaz de deserializar el objeto.

Aún, si hay algún error inesperado durante la ejecución es posible que la ventana del cliente se quede bloqueada y que haya que interrumpir la ejecución de manera manual.