

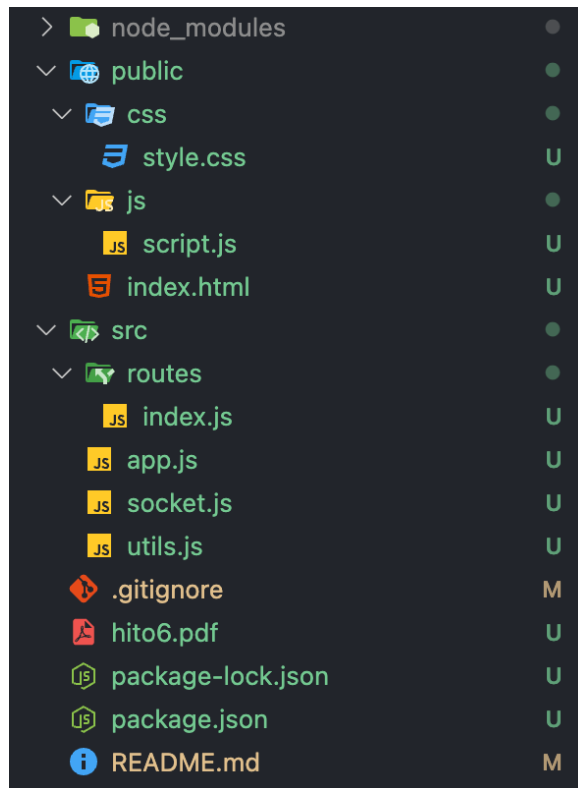
Evidencias Hito 6

Alumno: Nicolás Chávez Garzón

Grupo: 1

Hito: implementar un HTML para hacer funcionar el sistema de mensajería. Esto requiere implementar una ruta en el servidor para poder acceder al HTML.

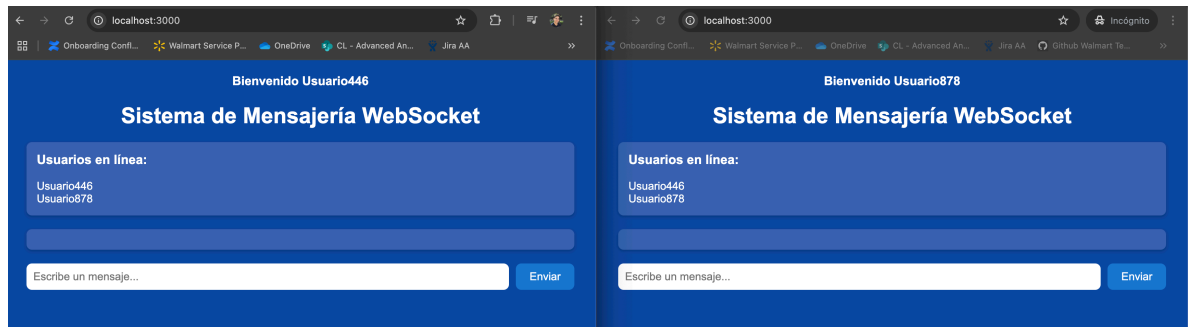
Se crea un proyecto con la siguiente estructura:



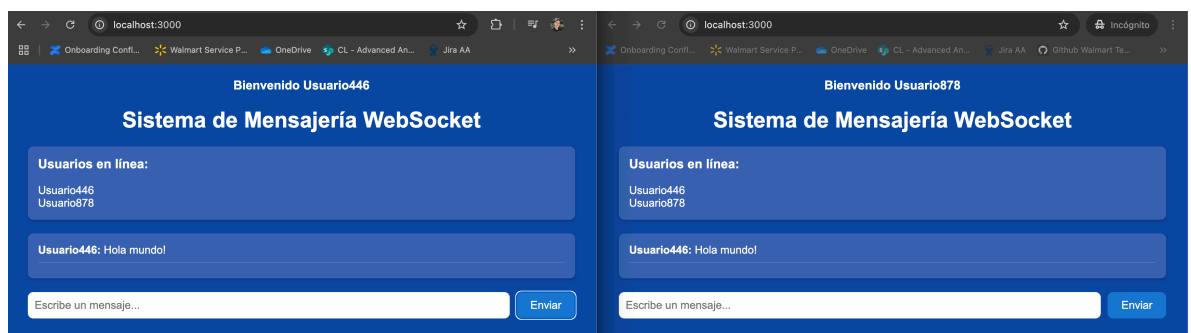
En “public” tenemos los estilos CSS utilizados para el front del sistema de mensajería, junto con el script y el HTML respectivo. Por otro lado, en “src” separamos las funcionalidades del sistema de mensajería en distintos scripts para que sean independientes al momento de realizar cambios. En `socket.js` definimos todos los eventos WebSocket solicitados para el hito, mientras que en `utils.js` creamos las funciones para manejar las desconexiones y los errores. Finalmente, `app.js` es nuestro archivo principal que inicializa el servidor.

Así, podemos inicializar el servidor de manera local y servir los componentes de WebSocket en el HTML para interactuar con el sistema de mensajería. A continuación, se presentan ejemplos:

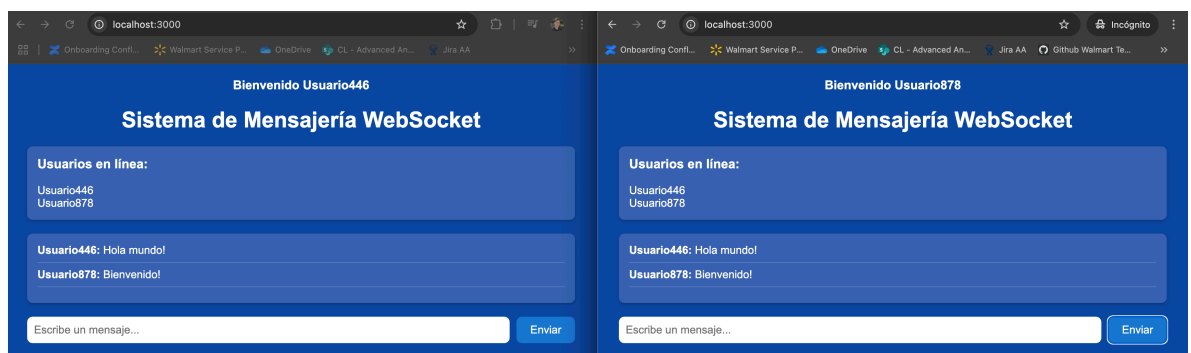
1. **Se inicializan dos sesiones en distintos navegadores, en donde podemos notar que existe un mensaje de bienvenida en cada sesión que identifica al usuario con un respectivo username (login).**



2. **Si enviamos un mensaje con la primera sesión, vemos que en la segunda sesión se recibe el mensaje de manera automática, identificando de manera correcta al primer usuario.**



3. **Asimismo, si respondemos con la segunda sesión, la primera recibirá el mensaje de manera correcta.**



4. Por último, si agregamos una tercera sesión, podemos ver que se identifica con un nuevo username y los otros usuarios pueden recibir sus mensajes e interactuar con él.

