



School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab (India)

Real-Time Chat Application

A platform of realtime chat

Pradeep Kumawat

Advance Web Development

Reg. No. - 11709697

Roll No. - 18



Faculty incharge: Miss Neha Sharma

Github Repo

<https://github.com/pradeepkumawat8600/RealTimeChatApplication/>

Overview

Real-time chat uses Web-based apps, which permit communication that is usually addressed directly but is anonymous among users in a multi-user environment.

In this project, Users register by giving a handle, which is unique to every user (a dummy name). Only the handle will be revealed to other users. So people are free to choose any handle and hence they stay anonymous.

Goals

1. An instant messaging solution to enable users to seamlessly communicate with each other.
2. Good readability with syntax highlighting.

Specifications

This project to create a chat application is made with nodejs and Express as a backend framework, handlebars as a templating engine, . Apart from these, I used a number of various js modules to make it work. Website is mobile friendly as well.



NodeJs

Node.js is primarily used for non-blocking, event-driven servers, due to its single-threaded nature. It's used for traditional web sites and back-end API services, but was designed with real-time, push-based architectures in mind.

ExpressJs

It is a Node.js web application framework. It provides a set of features to develop web and mobile applications. One can respond to HTTP requests using different middlewares and also render HTML pages.

Other Modules used

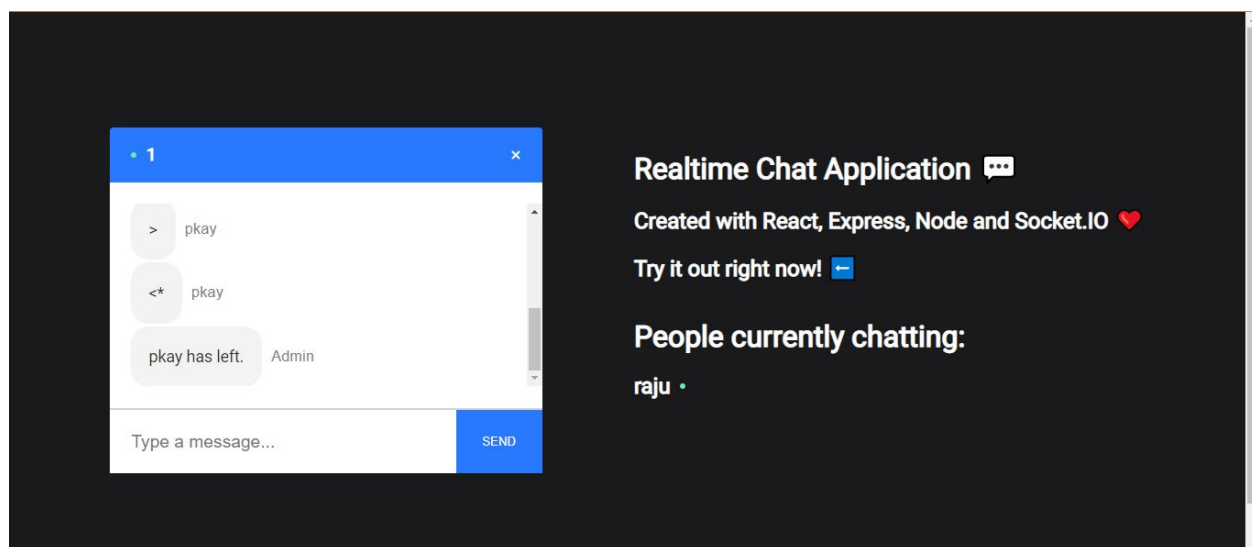
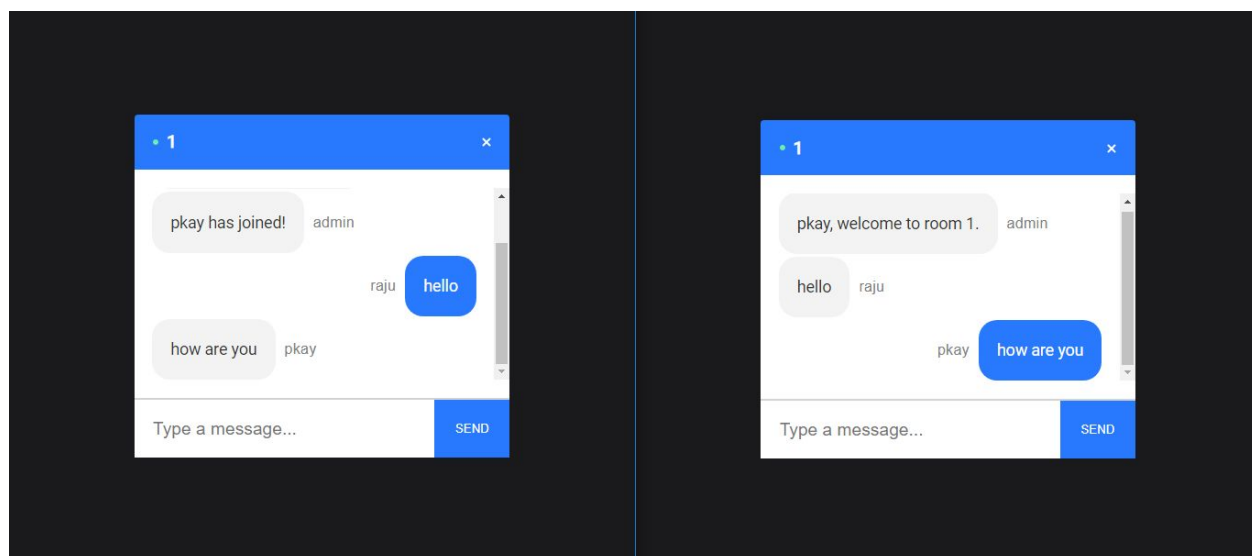
Socket.io

It is a javascript library for real-time web applications. It enables real-time, bi-directional communication between web clients and servers.

React.js

React is a library for building composable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time.

Screenshots



The screenshot shows the Visual Studio Code editor with the file explorer on the left. The project structure includes a 'client' directory with 'node_modules', 'public', and 'src' subdirectories. The 'Messages.js' file is open in the editor, showing the following code:

```

1  import React from 'react';
2
3  import ScrollToBottom from 'react-scroll-to-bottom';
4
5  import Message from './Message/Message';
6
7  import './Messages.css';
8
9  const Messages = ({ messages, name }) => (
10    <ScrollToBottom className="messages">
11      {messages.map((message, i) => <div key={i}><Message message={message} name={name}/></div>)}
12    </ScrollToBottom>
13  );
14
15  export default Messages;

```

The status bar at the bottom indicates the file is at line 1, column 1, with 2 spaces, in UTF-8 encoding, using LF line endings, and is a JavaScript file.

The screenshot shows the Visual Studio Code editor with the file explorer on the left. The project structure includes a 'server' directory with 'users.js'. The 'users.js' file is open in the editor, showing the following code:

```

1  const users = [];
2
3  const addUser = ({ id, name, room }) => {
4    name = name.trim().toLowerCase();
5    room = room.trim().toLowerCase();
6
7    const existingUser = users.find((user) => user.room === room && user.name === name);
8
9    if(!name || !room) return { error: 'Username and room are required.' };
10   if(existingUser) return { error: 'Username is taken.' };
11
12   const user = { id, name, room };
13
14   users.push(user);
15
16   return { user };
17 }
18
19 const removeUser = (id) => {
20   const index = users.findIndex((user) => user.id === id);
21
22   if(index !== -1) return users.splice(index, 1)[0];
23 }
24
25 const getUser = (id) => users.find((user) => user.id === id);
26
27 const getUsersInRoom = (room) => users.filter((user) => user.room === room);
28
29 module.exports = { addUser, removeUser, getUser, getUsersInRoom };

```

The status bar at the bottom indicates the file is at line 6, column 1, with 2 spaces, in UTF-8 encoding, using LF line endings, and is a JavaScript file.