React.js Project

Name- Shivam Nath

Reg. No. – 12018682

Title: Chatting Application

Introduction:

Real-time chat applications have become a ubiquitous part of modern communication. With the rise of smartphones and the internet, people have increasingly turned to these apps to stay connected with friends, family, and colleagues. In this report, we will analyze a real-time chat application and evaluate its features, performance, and usability. We will examine the application's design, functionality, and user experience, as well as its security and privacy measures. By doing so, I hope to provide a comprehensive overview of the Technologies and Modules of this real-time chat application.

Technologies Used:

The website is built using the MERN stack, which consists of the following technologies:

MongoDB: MongoDB is a document-based NoSQL database used to store and manage data. It is used in the backend of the website to store user information and chat data.

Express: Express is a web application framework used to build the backend of the website. It provides a robust set of features for creating server-side applications using Node.js.

React: The core technology used for building the user interface of the typing speed test application. ReactJS provides a component-based approach for building reusable UI components and managing the application state, allowing for efficient rendering and updating of the user interface.

HTML (Hypertext Markup Language): The standard markup language used for creating the structure and content of web pages. HTML is used to define the structure and layout of the typing speed test application, including elements such as input fields, buttons, and paragraphs.

Node.js: Node.js is a runtime environment that allows developers to run JavaScript on the server-side. It is used to run the backend server of the website.

Socket.IO: Socket.IO is a JavaScript library that enables real-time, bidirectional, and event-based communication between the server and the client using WebSockets. It is used to implement real-time messaging on the website.

CSS (Cascading Style Sheets): The style sheet language used for describing the presentation and layout of web pages. CSS is used to define the visual appearance and styling of the typing speed test application, including fonts, colors, spacing, and positioning of UI components.

Jest: A JavaScript testing framework used for writing unit tests for React components in the typing speed test application. Jest allows for automated testing of React components, ensuring the reliability and correctness of the application.

Axios: A popular HTTP client library used for making API requests in the typing speed test application. Axios is used to communicate with a server for fetching data, such as the text to be typed, from an external source for the typing test.

React Router: A routing library for handling client-side routing in a single-page application. React Router is used to manage navigation between different views or pages within the typing speed test application, allowing for a smooth user experience.

Prop-types: A library for adding type checking to React components. Prop-types is used in the typing speed test application for adding type checking to the props passed to React components, ensuring better code quality and early error detection.

These are the main technologies used in the Chat Application project, providing the necessary tools and libraries for building a responsive, interactive, and efficient web application for measuring typing speed.

Modules:

The website includes the following modules:

User Authentication: The user authentication module is responsible for managing user accounts. Users can register and log in to their accounts to access the chat application.

Real-Time Messaging: The real-time messaging module is responsible for enabling users to communicate with each other in real-time through text messages. The module is implemented using Socket.IO library, which provides a simple and reliable way to implement real-time messaging.

Chat Room Creation: The chat room creation module is responsible for enabling users to create chat rooms and invite other users to join. This module is implemented using MongoDB, which stores the chat room information.

React-DOM: This is the official DOM rendering library for React. It is used to render React components into the HTML DOM, allowing for the creation of interactive user interfaces in the typing speed test application.

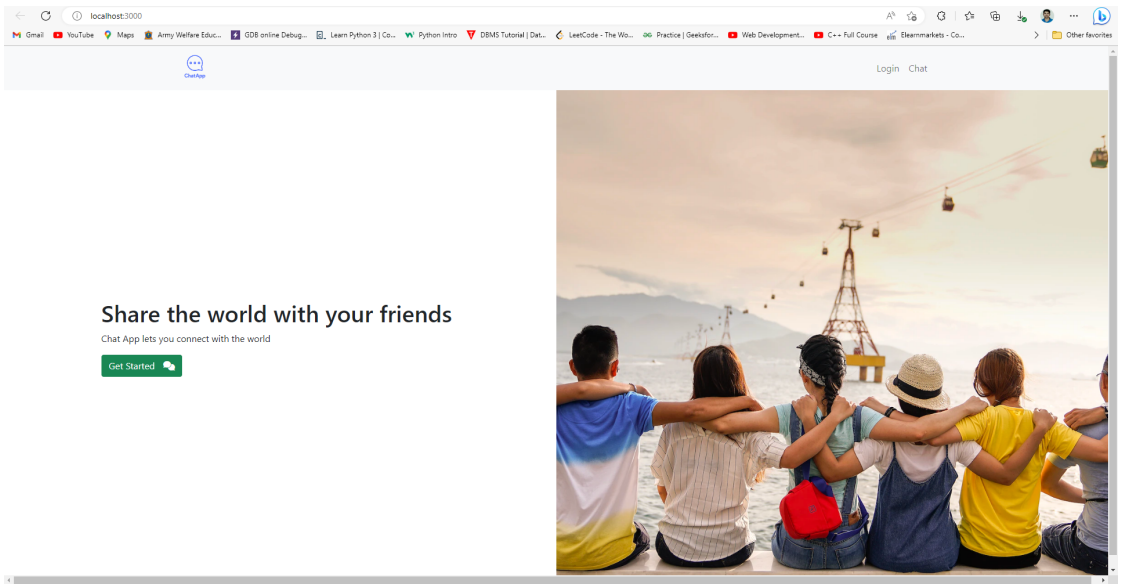
Chat Room Management: The chat room management module is responsible for allowing chat room owners to manage their rooms by removing members or deleting the room entirely. This module is implemented using MongoDB, which stores the chat room information.

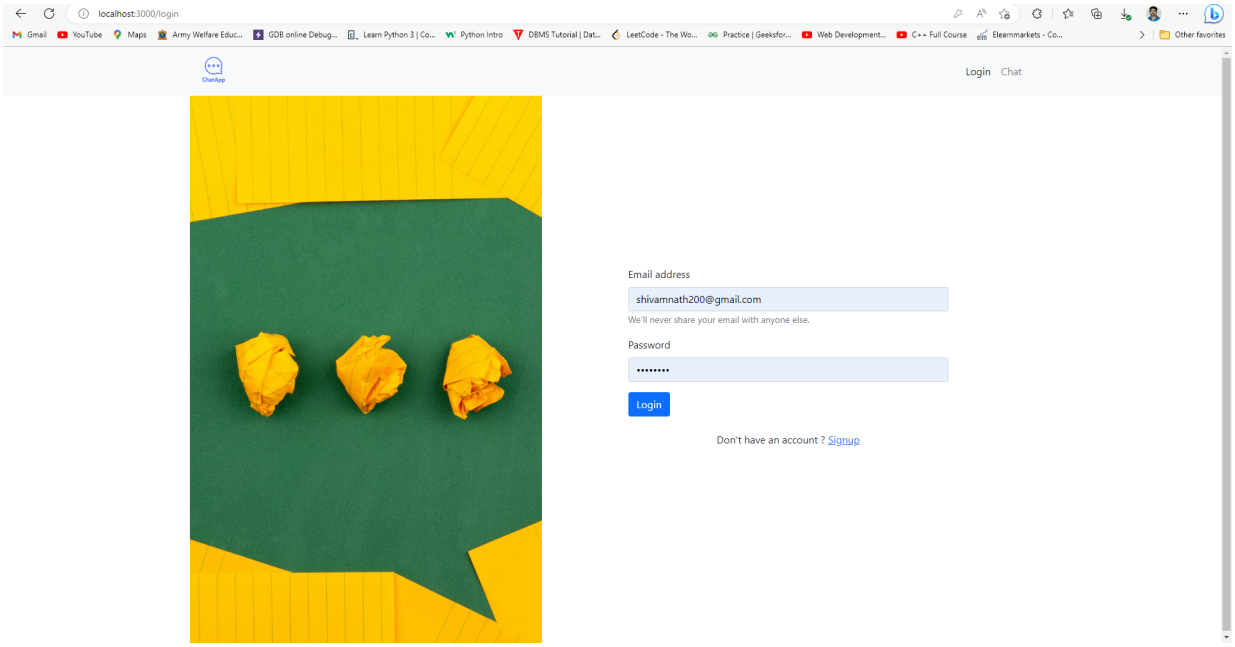
User Profile Management: The user profile management module is responsible for allowing users to update their profile information, including their profile picture. This module is implemented using MongoDB, which stores the user information.

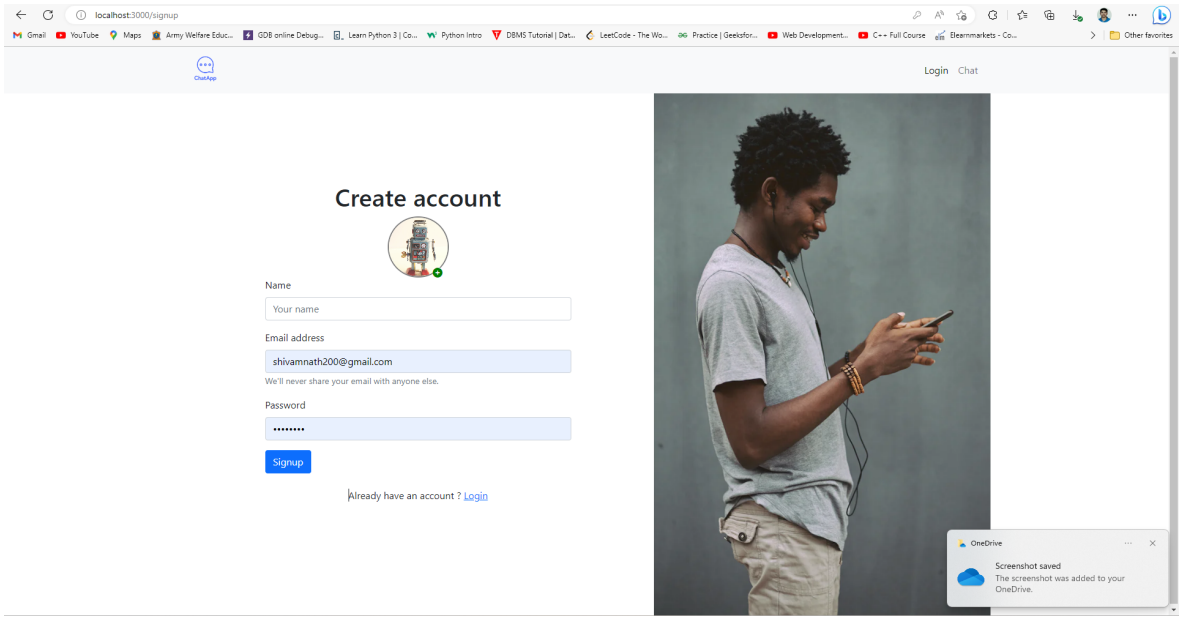
styled-components: This is a popular CSS-in-JS library that allows for writing CSS as JavaScript code. It is used in the typing speed test application for styling React components using tagged template literals, providing a convenient and scalable way to manage styles within the application.

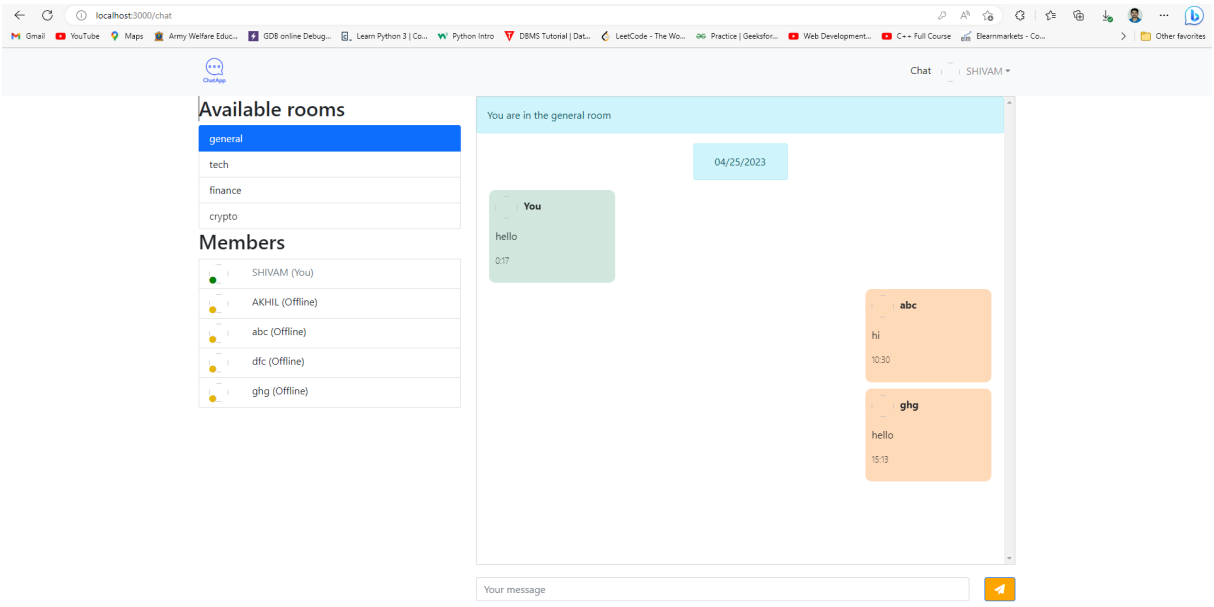
Website Snapshots:

Below are some snapshots of the website:




localhost:3000/login
Login
Chat
Email address
shivamnath200@gmail.com
We'll never share your email with anyone else.
Password
Login
Don't have an account ? Signup


localhost:3000/signup
Login
Chat
Create account
Name
Your name
Email address
shivamnath200@gmail.com
We'll never share your email with anyone else.
Password
Signup
Already have an account ? Login
OneDrive
Screenshot saved
The screenshot was added to your OneDrive.



Conclusion: In conclusion, the chatting website made using MERN stack provides users with a reliable and easy-to-use platform for real-time communication through text messages. The website is built using the latest web technologies and provides a smooth and responsive user experience. The website snapshots demonstrate the features and functionalities available to users.

Github Link: https://github.com/shivamnath2001/NODE-AND-REACT-PROJECT

THANKYOU