**+**



**Hope Foundation's**

**International Institute of Information Technology,**

**Pune-57**

**DEPARTMENT OF COMPUTER ENGINEERING**

**2019 Pattern**

**Academic Year: 2023-2024 Semester: II Year: T.E Date: 16/04 /2024**

**Mini Project Title: De-Link Chat App**

**Team Members:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Roll. No.** | **Name** | **Mail-id** | **Contact No.** |
| TA16 | Pradnya Bornar | pradnyab\_c21079@ students.isquareit.edu.in | 9822598347 |
| TA23 | Nishigandha Deore | nishigandhad\_c21053@students.isquareit.edu.in | 8806877866 |
| TA66 | Aranav Mahalpure | aranavm\_c21020@students.isquareit.edu.in | 9975659711 |

**INDEX**

**1. Project Details**

1. Project Title
2. Problem Statement

1. **Purpose**

1. **Technical Requirements Details**

1. **Flowchart**

1. **Detailed Description**

1. **Installation & project Execution steps**

1. **Conclusion**

**Project Details**

* **Project Title:   
    
  De-Link Chat App**
* **Problem Statement:**

Develop an online web Application for communication between the Users by implimenting its Frontend and Backend using respective technologies like:

* + MongoDB (Database)
  + ExpressJS (Web application framework)
  + NodeJS (JavaScript Runtime)
  + React (Frontend framework)
  + Socket.io (web sockets for real time chat)

**Purpose**

We also aim to learn various web technologies that can be used to develop a web application, by creating such a project we get hands-on practice to impliment and understand the technologies that we are using and understand its working.

The Online Chat Application allows you to communicate with others in real-time, making it simpler, easier, and more convenient. It aims to bring people closer by removing barriers and giving everyone a chance to connect and interact. It helps in keeping conversations safe from unauthorized access. Also, it saves time and money by handling tasks like message delivery and storage automatically. By using technology like computers and the internet, it's all about making the communication process easier, faster, and more efficient, so that everyone's voice can be heard in the digital world.

**Technical Requirements Details**

* Frontend: This is the user interface of the application. We will use React for this
* Backend: This is the server-side of the application. We will use Node.js and Express for this.
* Database: This is where we will store all the data. We will use MongoDB for this.

Now, let's discuss each component in detail:

Frontend (React):

* React: This is a JavaScript library for building user interfaces. It's used for handling the view layer in web and mobile apps. We will use it to create the UI components of our application.

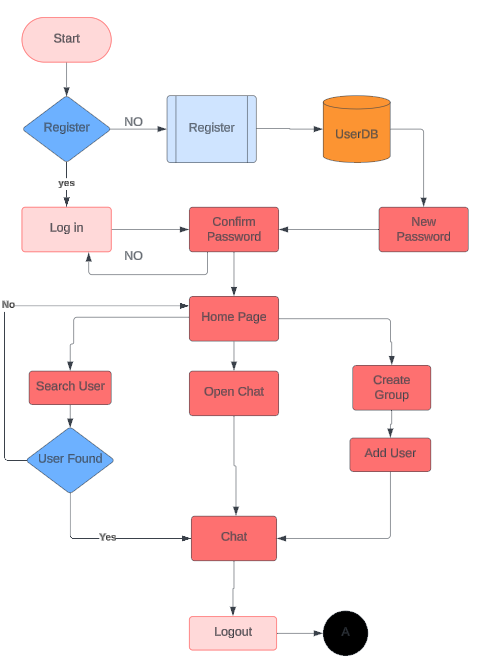
Backend (Node.js and Express):

* Node.js: This is a JavaScript runtime built on Chrome's V8 JavaScript engine. It's used to build scalable network applications.
* Express: This is a web application framework for Node.js. It's used to build web applications and APIs.

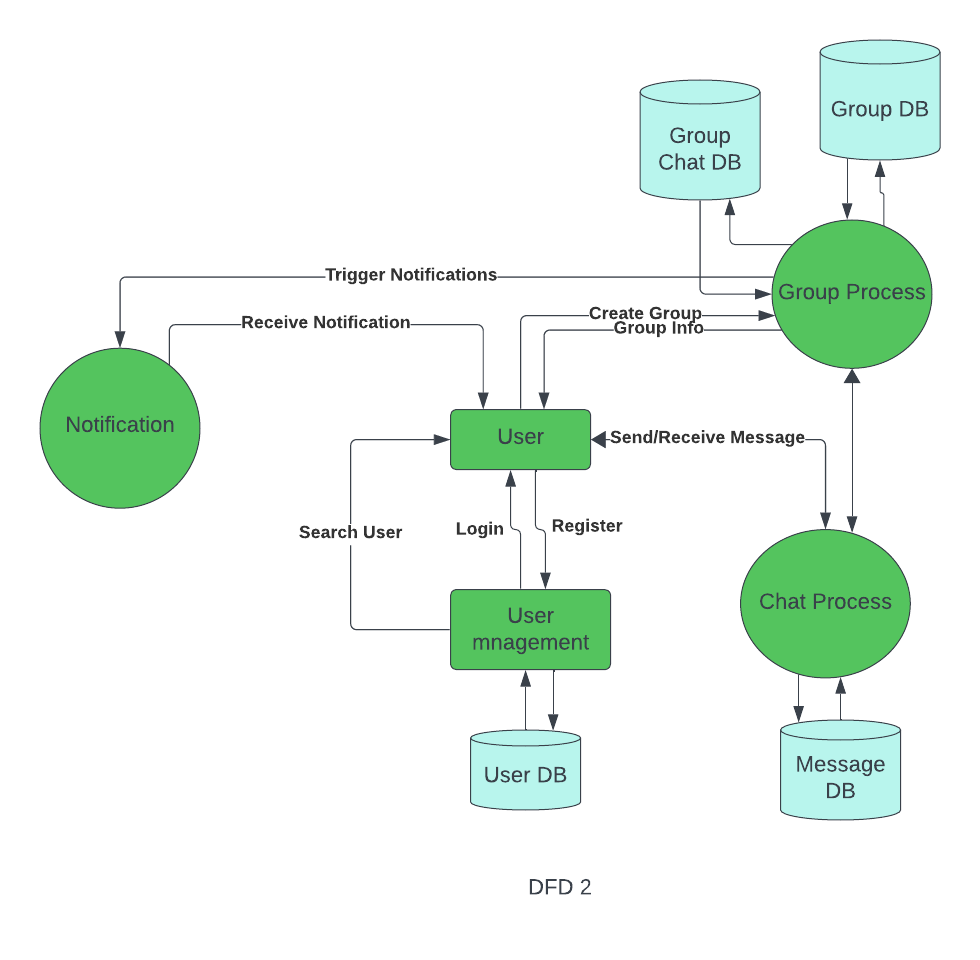
Database (MongoDB):

* MongoDB: This is a NoSQL database program. It's used for storing data in a document-oriented format.

**Flow chart**



**System Architecture**



**Detailed Description**

Frontend (React):

* React: This is a JavaScript library for building user interfaces. It's used for handling the view layer in web and mobile apps. We will use it to create the UI components of our application.

Backend (Node.js and Express):

* Node.js: This is a JavaScript runtime built on Chrome's V8 JavaScript engine. It's used to build scalable network applications.
* Express: This is a web application framework for Node.js. It's used to build web applications and APIs.

Database (MongoDB):

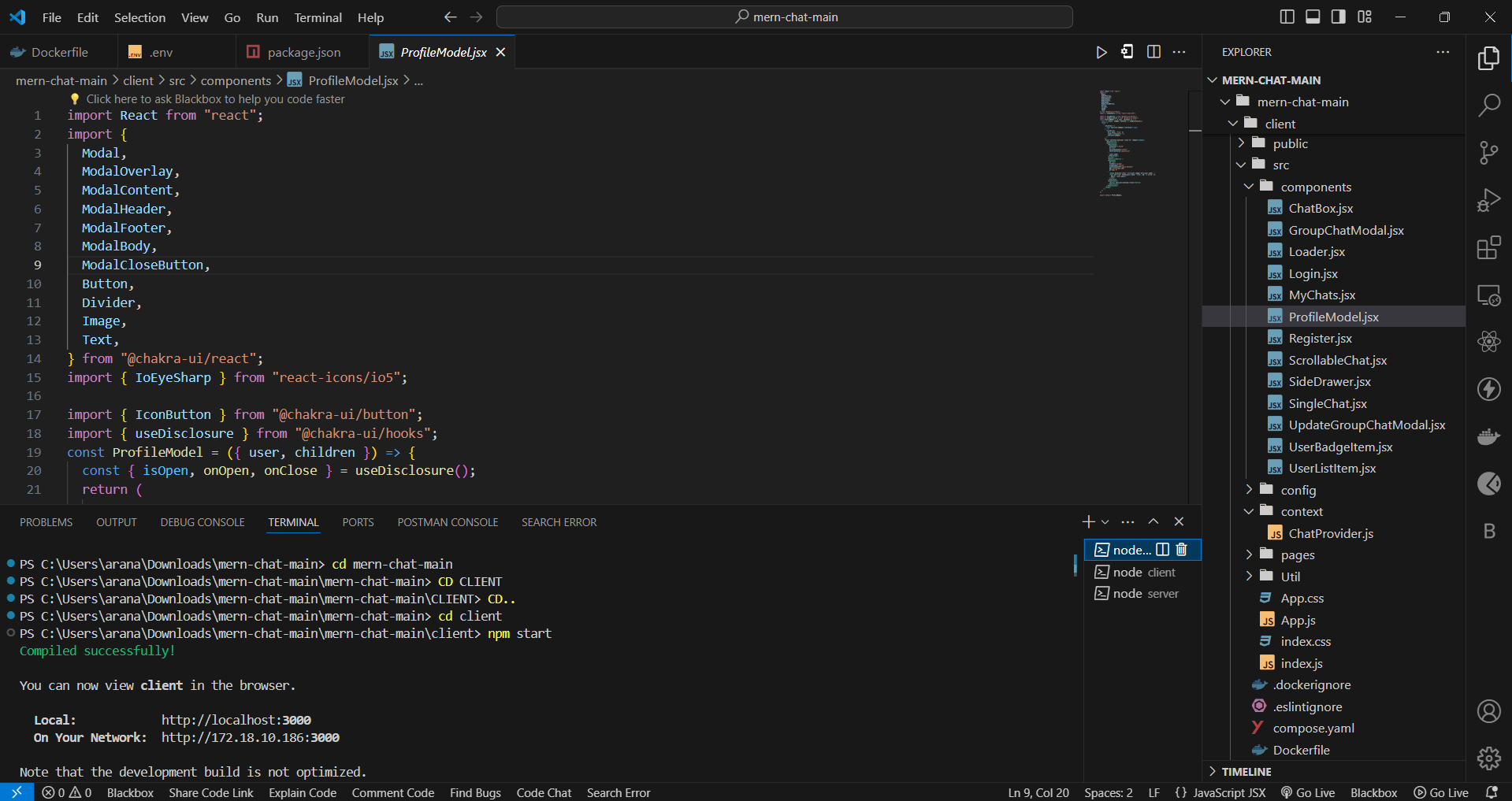
* MongoDB: This is a NoSQL database program. It's used for storing data in a document-oriented format.

Stepwise Working of System:

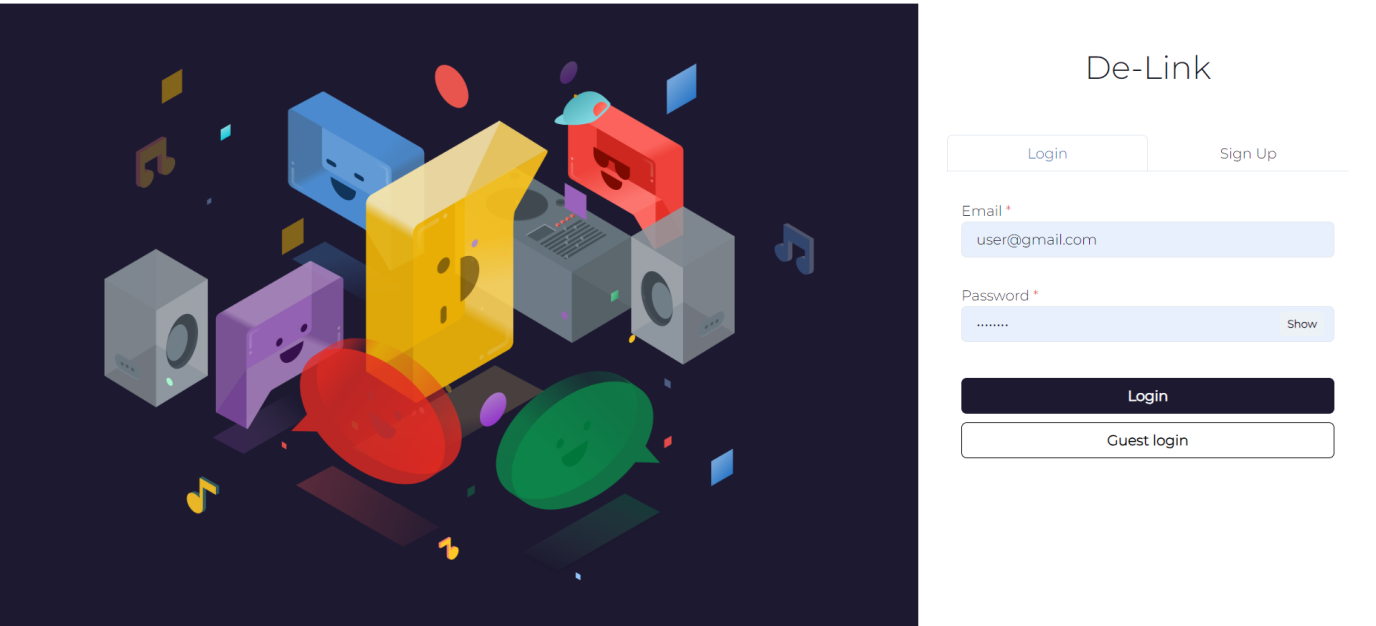
* + Step 1 : starts at the login interface , it contains option to login for the User , register new User. Or guest login.
  + Step 2: after User login voter has the option to search for begin the chat or Resume the chat with his/her friend.
  + Step 3: after clicking on the registration we are able to register a new User which gets added to the database .
  + Step 4: added the feature of creating Group link What-app minimun requirement of this is 3 user only .

**Installation & project Execution steps and Output Screenshots**

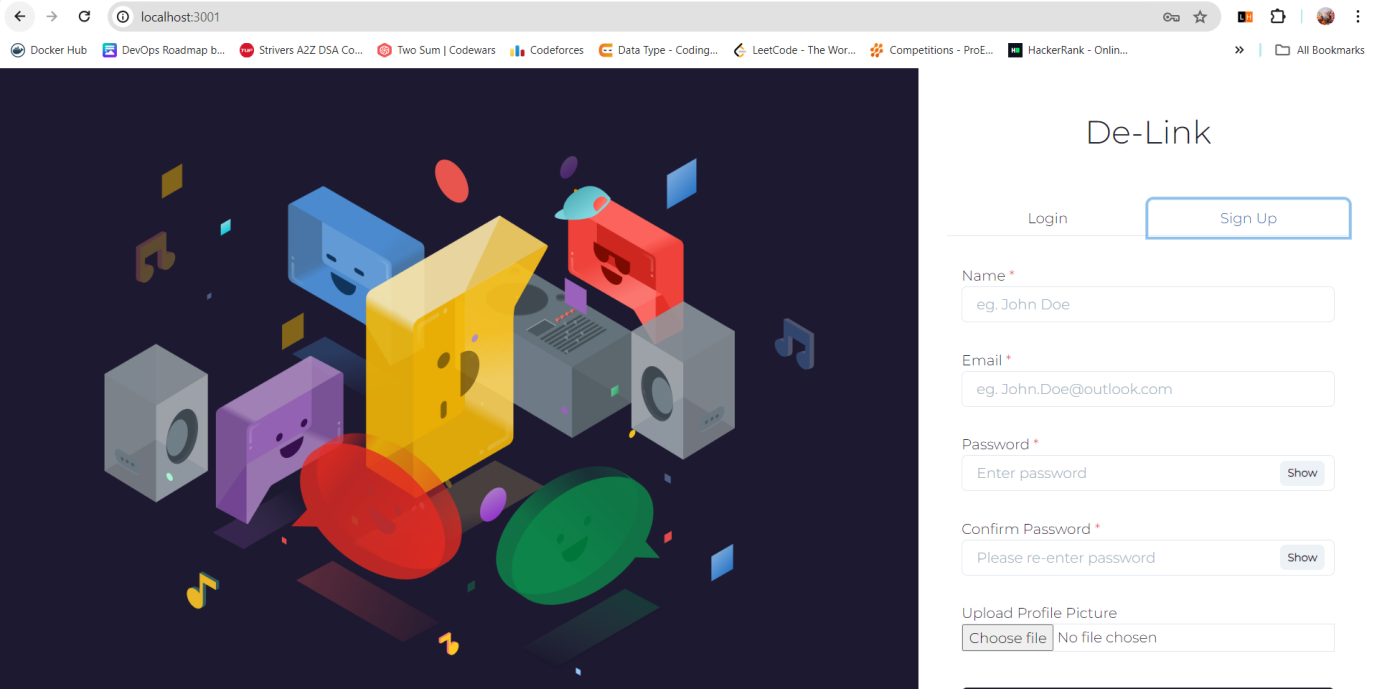
1: Running Client Side



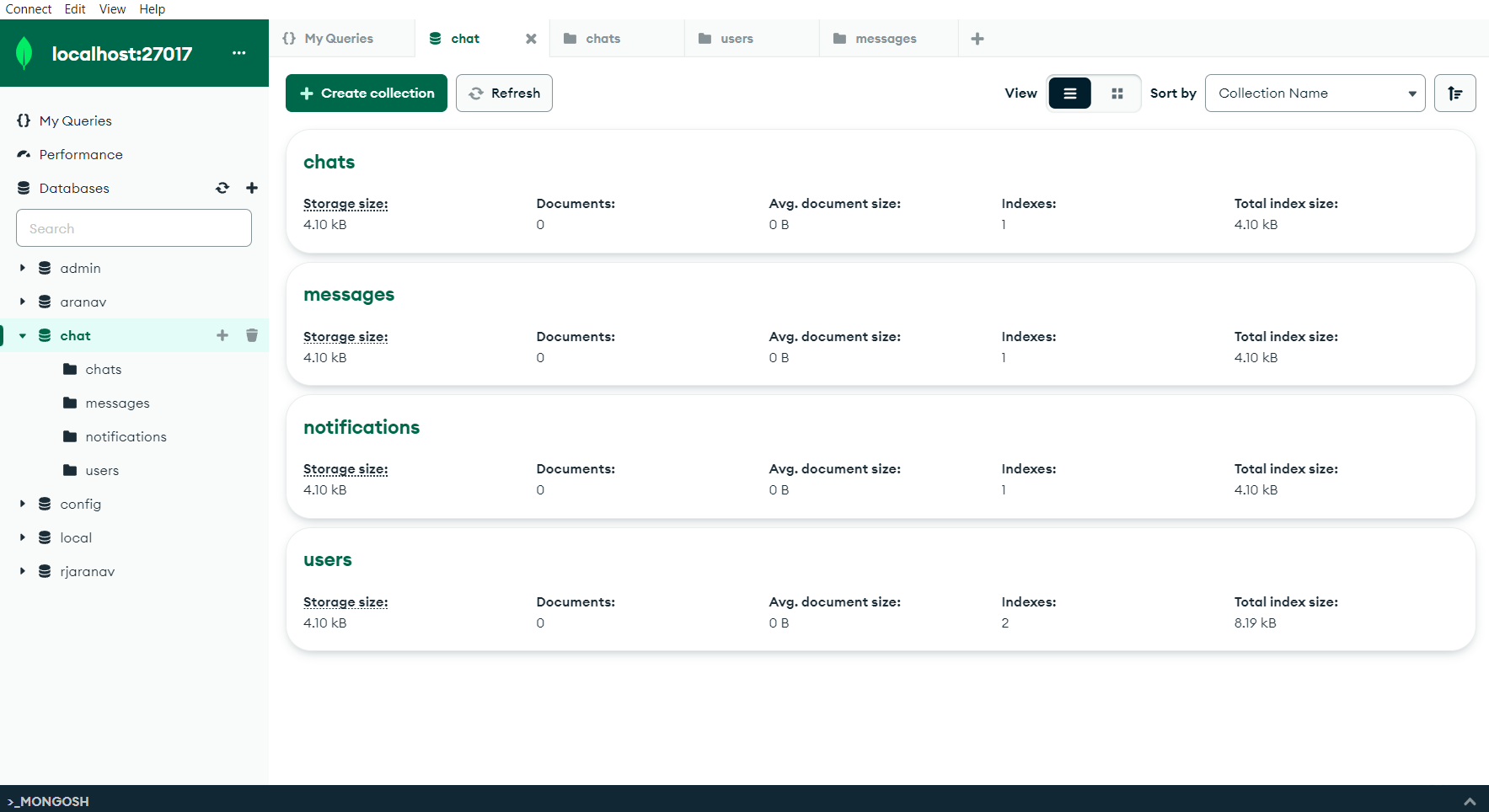
2; Front End Side (Login Page )



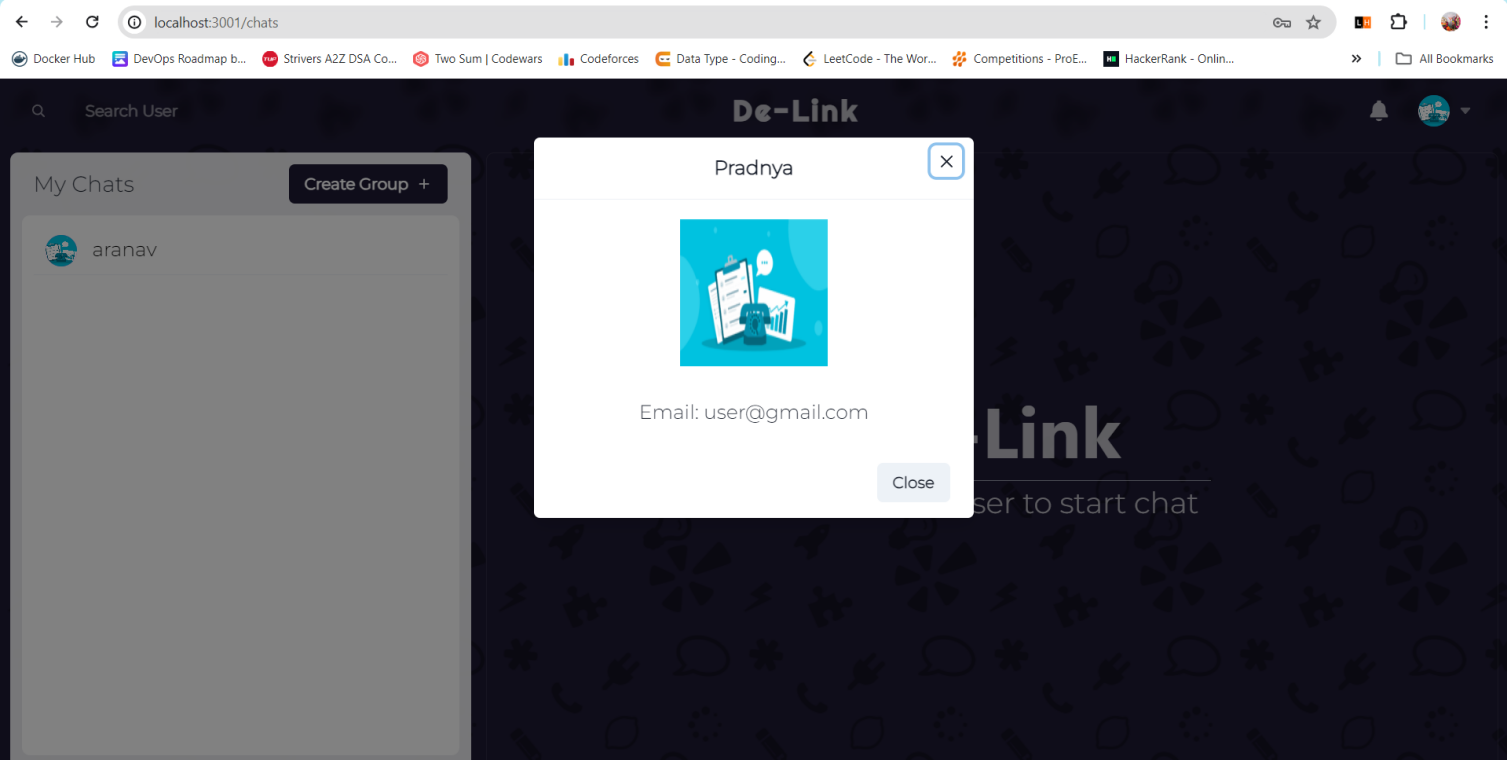
3:(Registration Page)

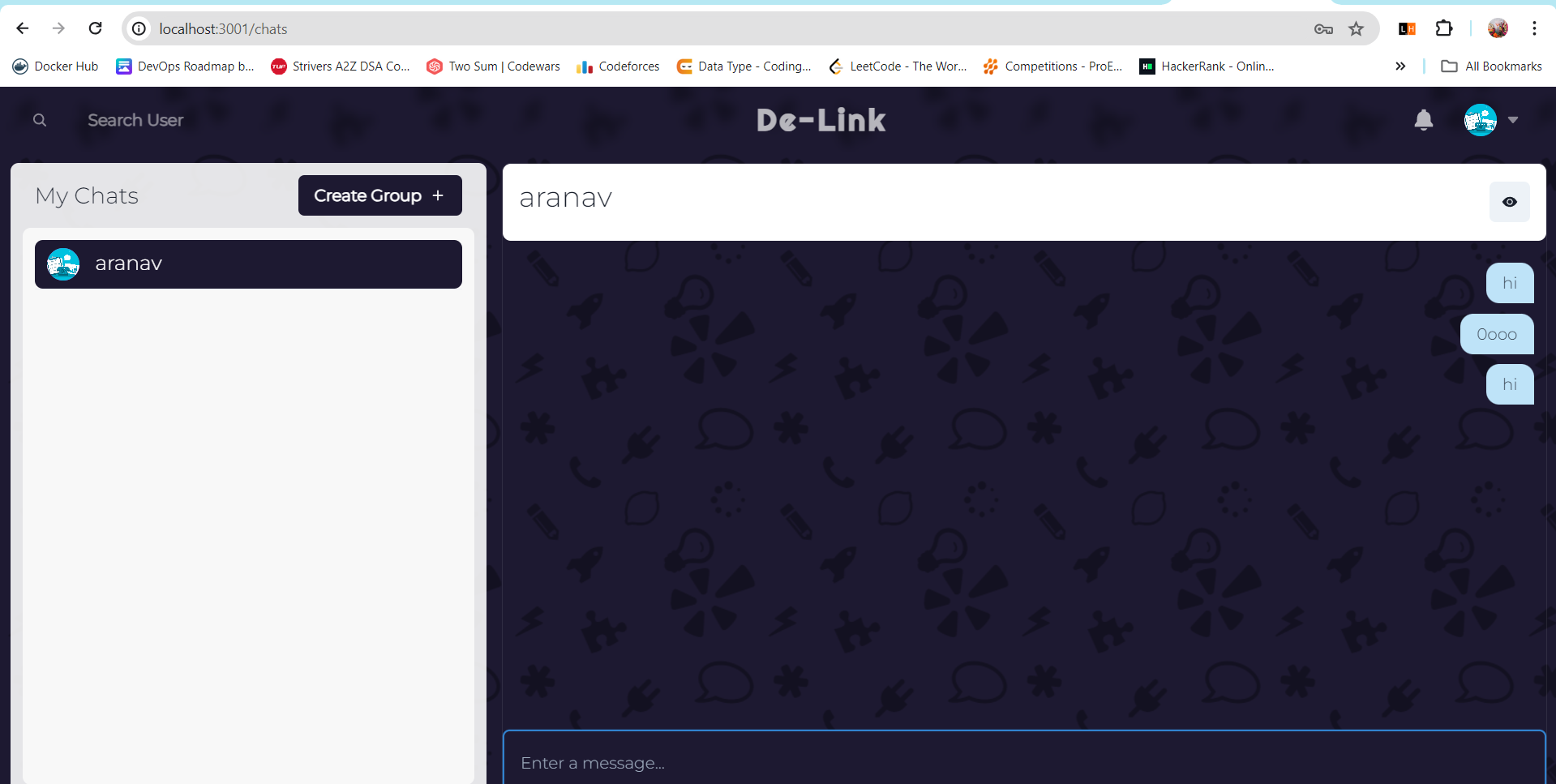


4: MongoDB compass set up

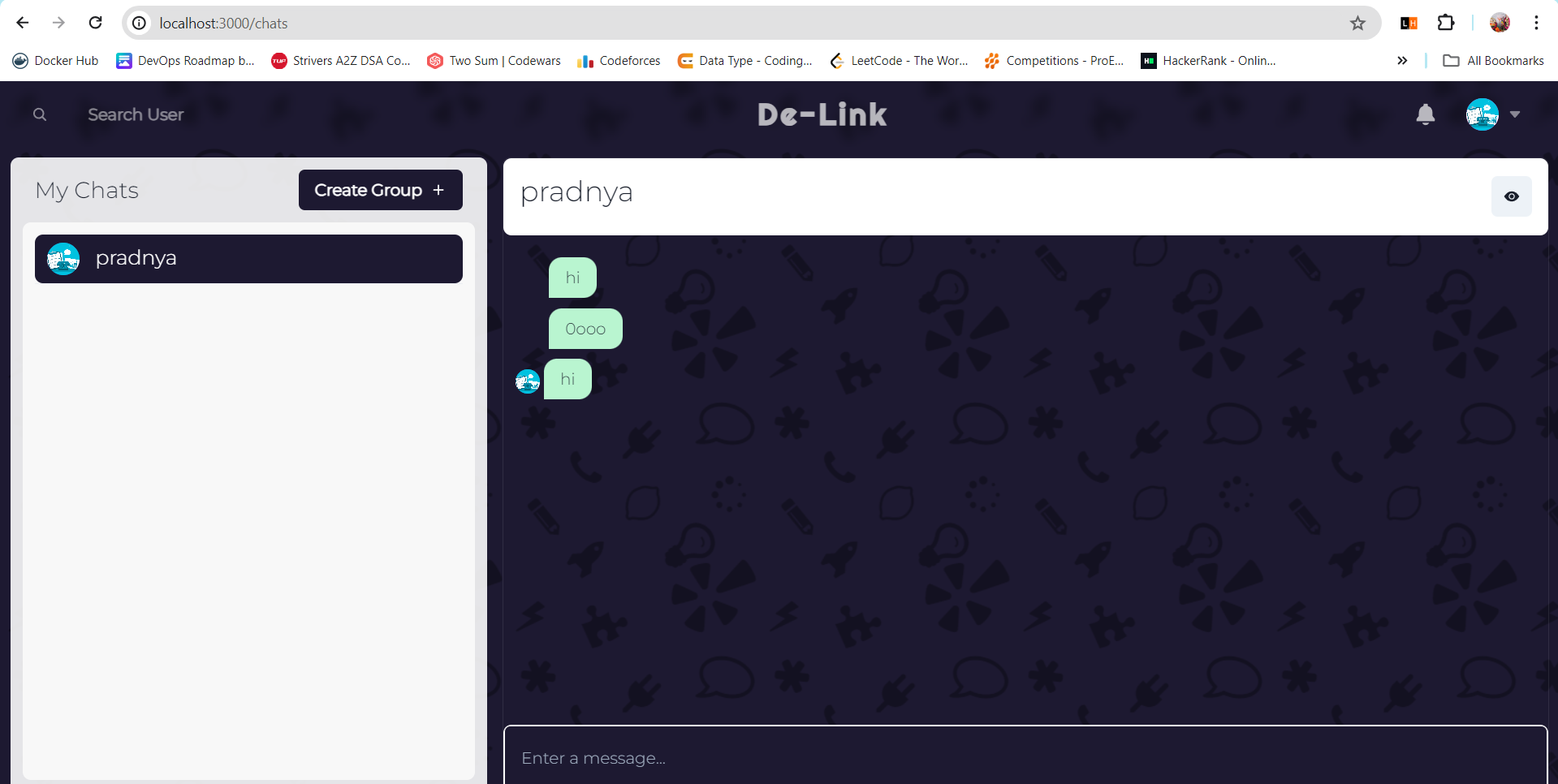


5: After Login Successful

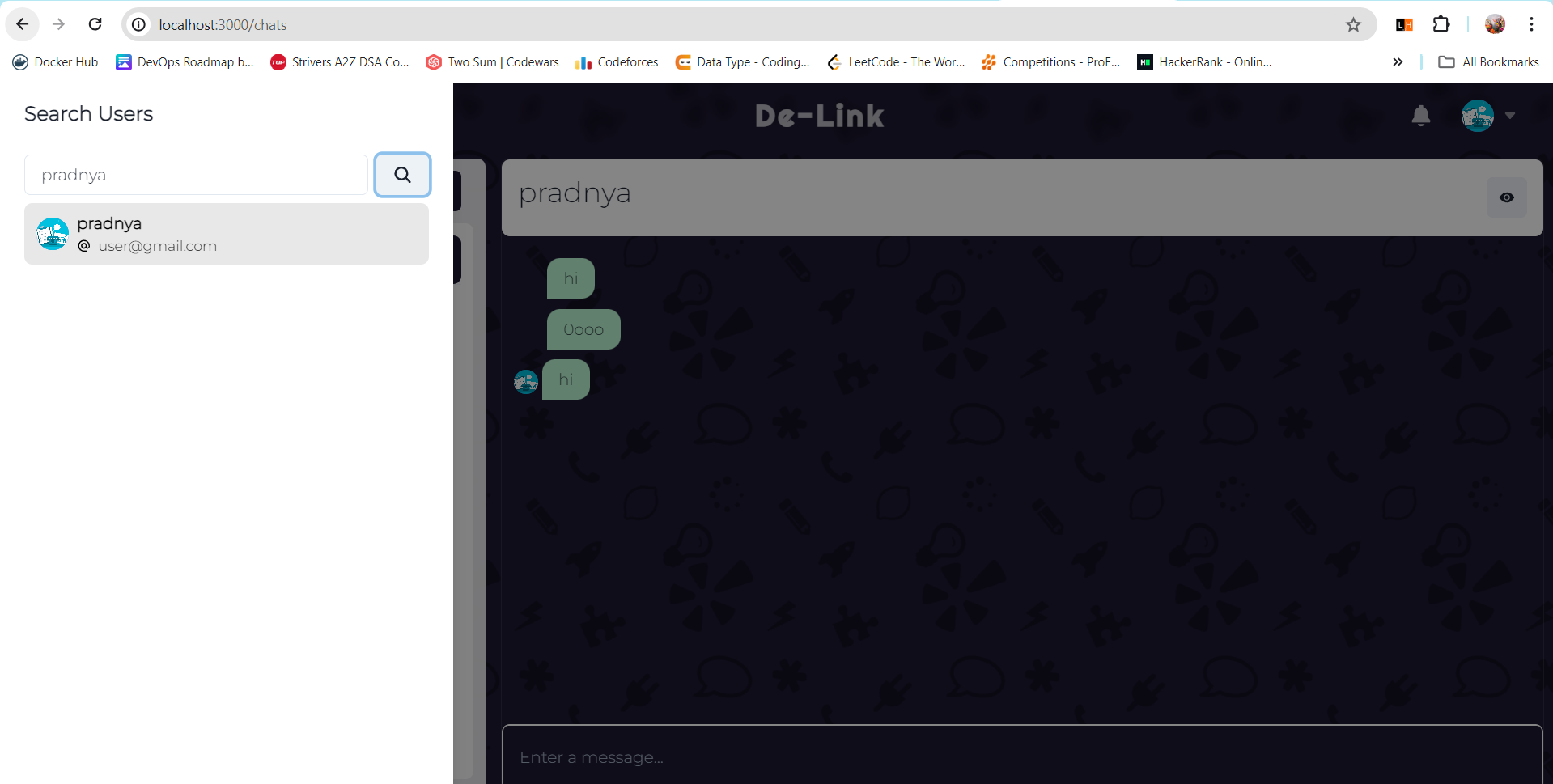


6: Initializing Chat with Other User

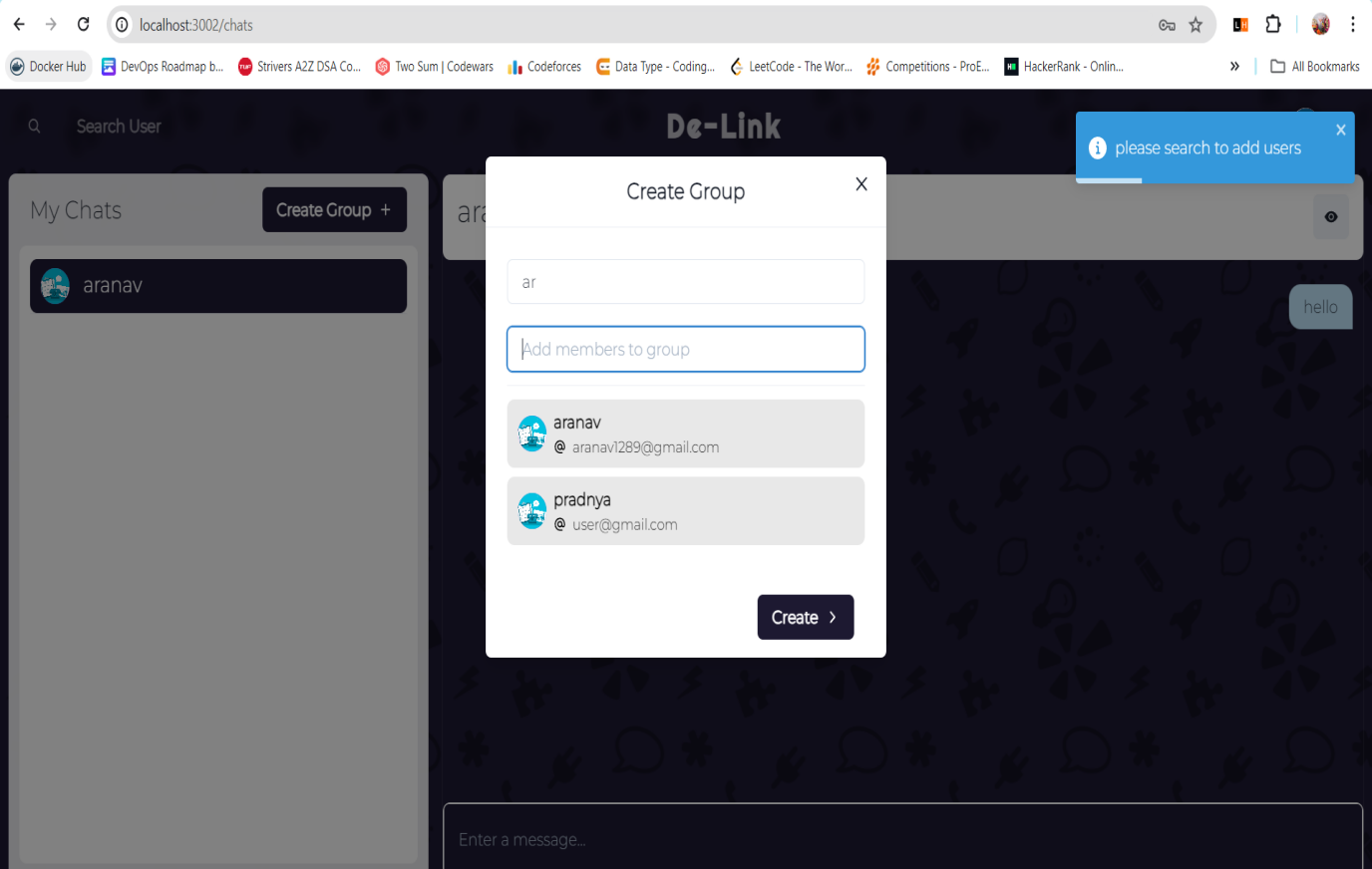
7. Receive Notification to Other User



8: Searching For Other User



9: Creating Group



**Conclusion**

The development of our online chat web application make use of various technologies to ensure its functionality and reliability across multiple platforms. Using React, HTML along with frontend technologies like ChakraUI and CSS, we created an user interface accessible on various platforms. The use of MongoDB Compass as the server and MongoDB as the database management system ensured easy operation and data integrity.Additionally, using Express and Node for backend technology. By using these technologies, we made a safe, fast, and easy Online Chat Application to make Communication Between Users.