**UNIVERSITY OF MUMBAI**



**Master Of Computer Application**

**Mini Project Report on**

**WeChat : A Chat Application**

**Submitted By Kajal Purbiya & Sundaram Gupta**

**Roll No. 32 & 10.**

**Under the Guidance of Prof. Rashmi Pathak.**

|  |  |  |  |
| --- | --- | --- | --- |
| **1. Introduction** | | 3 | |
| Introduction of Project | | 4 | |
| Project Purpose | | 4 | |
| Project Objective | | 4 | |
| Scope of Project | | 5 | |
| **2. System Study** | | 6 | |
| Requirement Analysis | | 7 | |
| Planning and Scheduling | | 7 | |
| Preliminary Product Description | | 8 | |
| Justification of Platform | | 8 | |
| **3. Analysis and Design** | | 9 | |
| Hardware Requirements | | 10 | |
| Software Requirements | | 10 | |
| Actual Gantt Chart | | 10 | |
| System Design | | 11 | |
| UML Diagrams | | 11 | |
| **4. Testing and Validation** | | 12 | |
| Code Efficiency | | 13 | |
| Testing Approach | | 13 | |
| Unit Testing | | 13 | |
| Integration Testing | | 13 | |
| **5.User Manual** | | 14 | |
| User Documentation | | 15 | |
| Outputs | | 15 | |
| Code Snippets | | 18 | |
| **6.Conclusion** | | 22 | |
| Conclusion | | 23 | |
| Limitations | | 23 | |
| Future Scope | | 23 | |

**MODULE 1**

**1. Introduction**

**a. Introduction of Project**

**b. Project Purpose**

**c. Project Objective**

**d. Scope of Project**

1. **Introduction**
2. **Introduction of Project**

The project aims to develop a robust and efficient Chat Application named "WeChat" In an era where digital security and Information is of paramount importance, WeChat serves as a secure platform for transporting sensitive messages from one user to another. With a core focus on security, the system employs AES-128 Encryption for the purpose of securing and Organizing user data. This application is designed to streamline the process of managing and retrieving messages while prioritizing data confidentiality. Leveraging advanced cryptographic techniques and arithmetic encoding, this Chat app provides users with a reliable solution to safeguard their chatting Info.

1. **Project Purpose**

Traditional chat management systems may fall short in providing adequate protection. The project addresses these issues by offering a messaging application with end-to-end encryption solution with enhanced security features. The primary purpose of the password vault is to address the common challenges users face in managing multiple passwords across various online platforms. It provides a secure and user-friendly environment for storing, retrieving, and organizing passwords. The use of arithmetic encoding contributes to the overall security of the system, ensuring that stored passwords remain confidential.

1. **Project Objective**

The primary objective of the WeChat project is to design and implement a state-of-the-art password vault that ensures the highest standards of security for user credentials. The system aims to:

• Provide a user-friendly interface for effortless password management.

• Implement strong encryption algorithms to protect stored passwords.

• Utilize arithmetic encoding to enhance the encoding and decoding processes.

• Mitigate the risk of password-related security breaches

1. **Scope of Project**

The scope of the WeChat project encompasses various dimensions:

**• Security:**

Implementing robust encryption mechanisms to safeguard user passwords and sensitive data from unauthorized access.

**• Usability:**

Creating an intuitive and user-friendly interface for easy navigation and efficient password management. Integration: Exploring the integration of arithmetic encoding to enhance the security of password encoding and decoding processes.

**• Compatibility:**

Ensuring compatibility with multiple platforms and devices to accommodate diverse user preferences

**Module 2**

**2. System Study**

**Requirement Analysis**

**Planning and Scheduling**

**Preliminary Product Description**

**Justification of Platform**

**Conceptual Model**

1. **Requirement Analysis:**

Requirement analysis is a crucial step in the development of any software application, including a chat application. It helps in understanding and defining the scope of the project, identifying user needs, and establishing the features and functionalities that the application must have. Here's a comprehensive list of requirements for a chat application.

1. **Planning and Scheduling**

Planning and scheduling are critical aspects of developing a chat application. Below is a suggested plan and schedule for the development of a chat application. Keep in mind that the timeline and tasks may vary based on the complexity of your specific project.

**B.1. Project Kick-off and Planning:**

* Define project goals, objectives, and success criteria.
* Identify stakeholders and establish communication channels.
* Conduct a thorough requirement analysis.

### B.2. ****Technical Feasibility:****

* Assess the technical feasibility of the project.
* Choose the technology stack (backend, frontend, database).
* Set up development environments.

1. **Preliminary Product Description**

Describe the core features, such as secure password storage, retrieval, and user authentication.

Highlight the security protocols in place, including encryption methods and access controls.

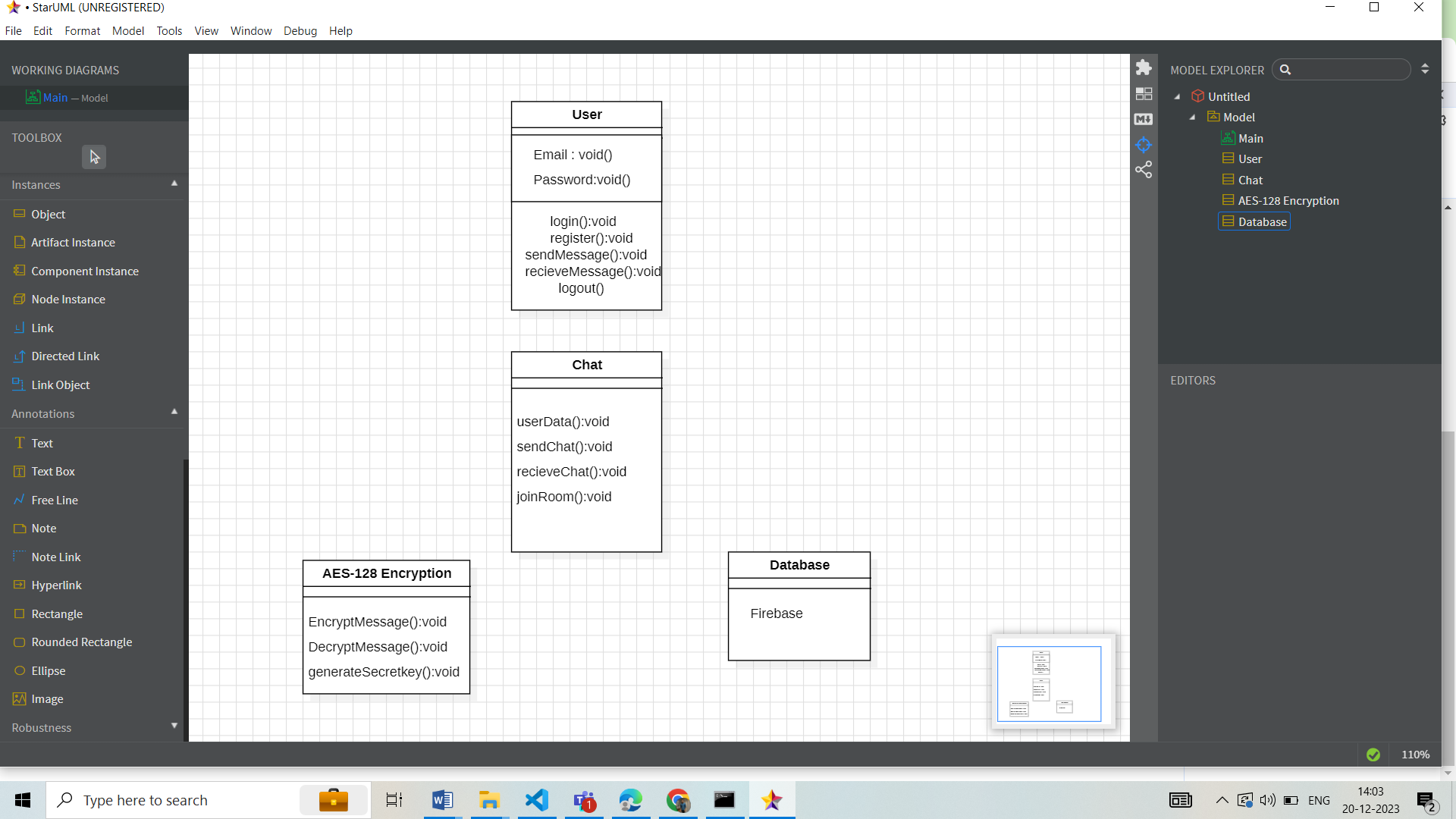
Briefly mention how the system ensures compatibility across different browsers and devices.

1. **Justification of Platform**

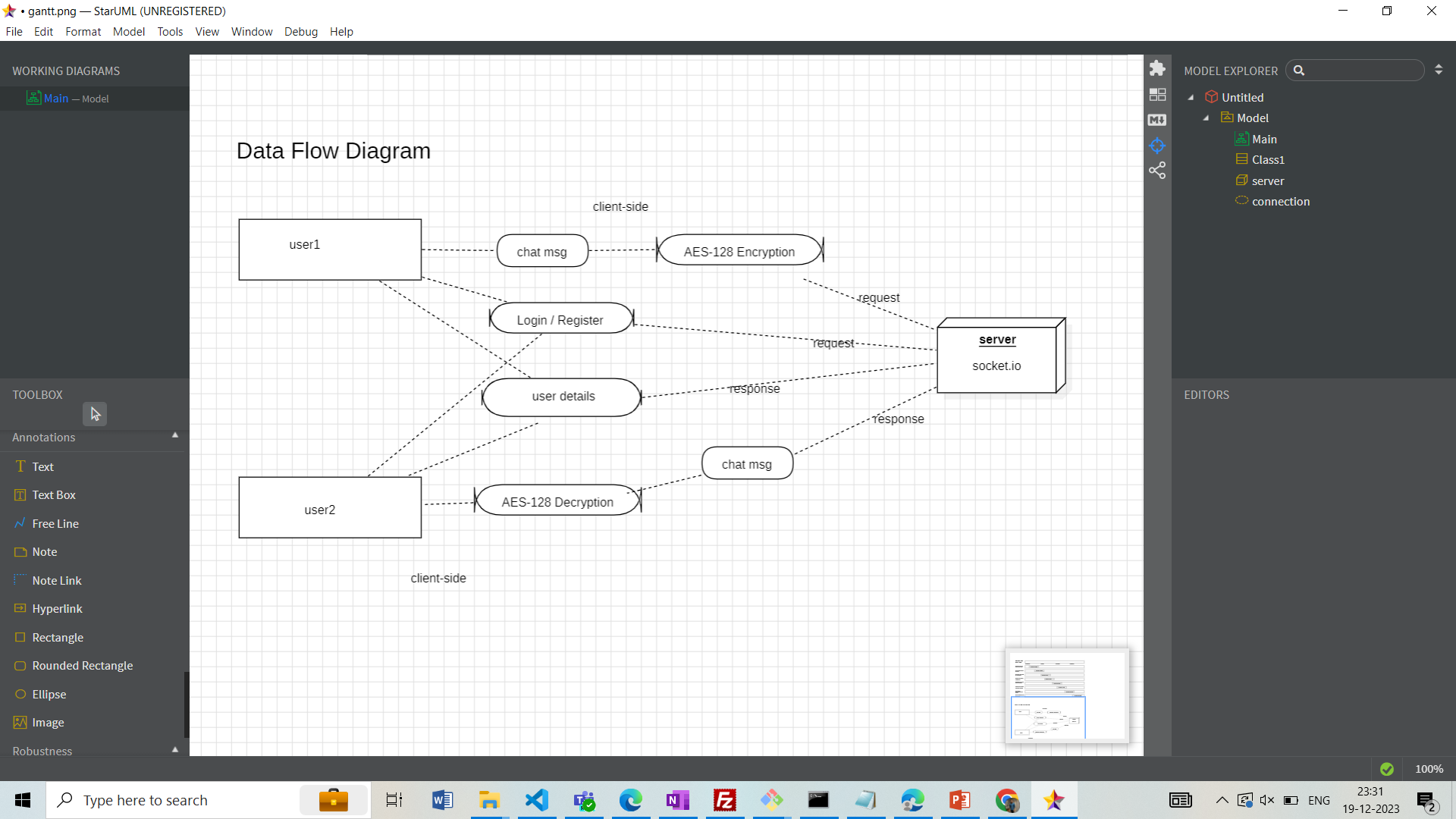
Node and Express: Justify the use of Django and Python for the backend, emphasizing their security features, scalability, and ease of development.

HTML, CSS, and JavaScript: Explain the choice of these front-end technologies, highlighting their role in creating an intuitive and responsive user interface.

1. **Conceptual Model**

****

1. **Data Flow Diagram**

****

**MODULE 3**

1. **Analysis and Design**

**Hardware Requirements**

**Software Requirements**

**Actual Gantt Chart**

**System Design**

**UML Diagrams**

1. **Hardware Requirements**

• Processor: Dual-core or higher

• RAM: 8 GB or higher

• Storage: SSD for better performance

1. **Software Requirements**

• Operating System:

• Windows

• Web Browser:

• Google Chrome, Mozilla Firefox, Safari

• Backend Development:

• Programming Language: Node

• Framework: Socket.io & Websockets

• Database: Firebase

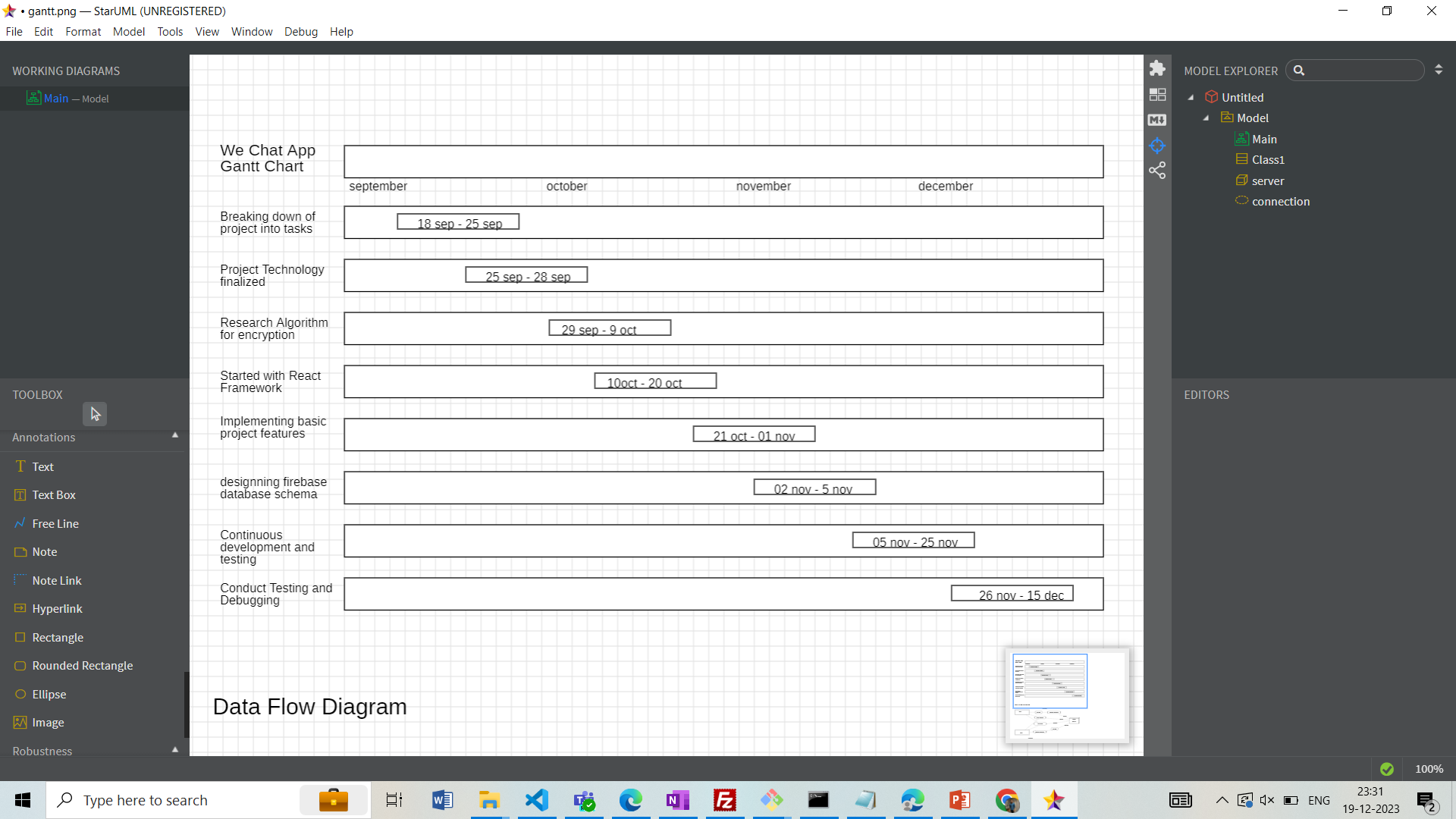
• Frontend Development:

• HTML5, CSS3, JavaScript

• AES-128 Encryption & Decryption

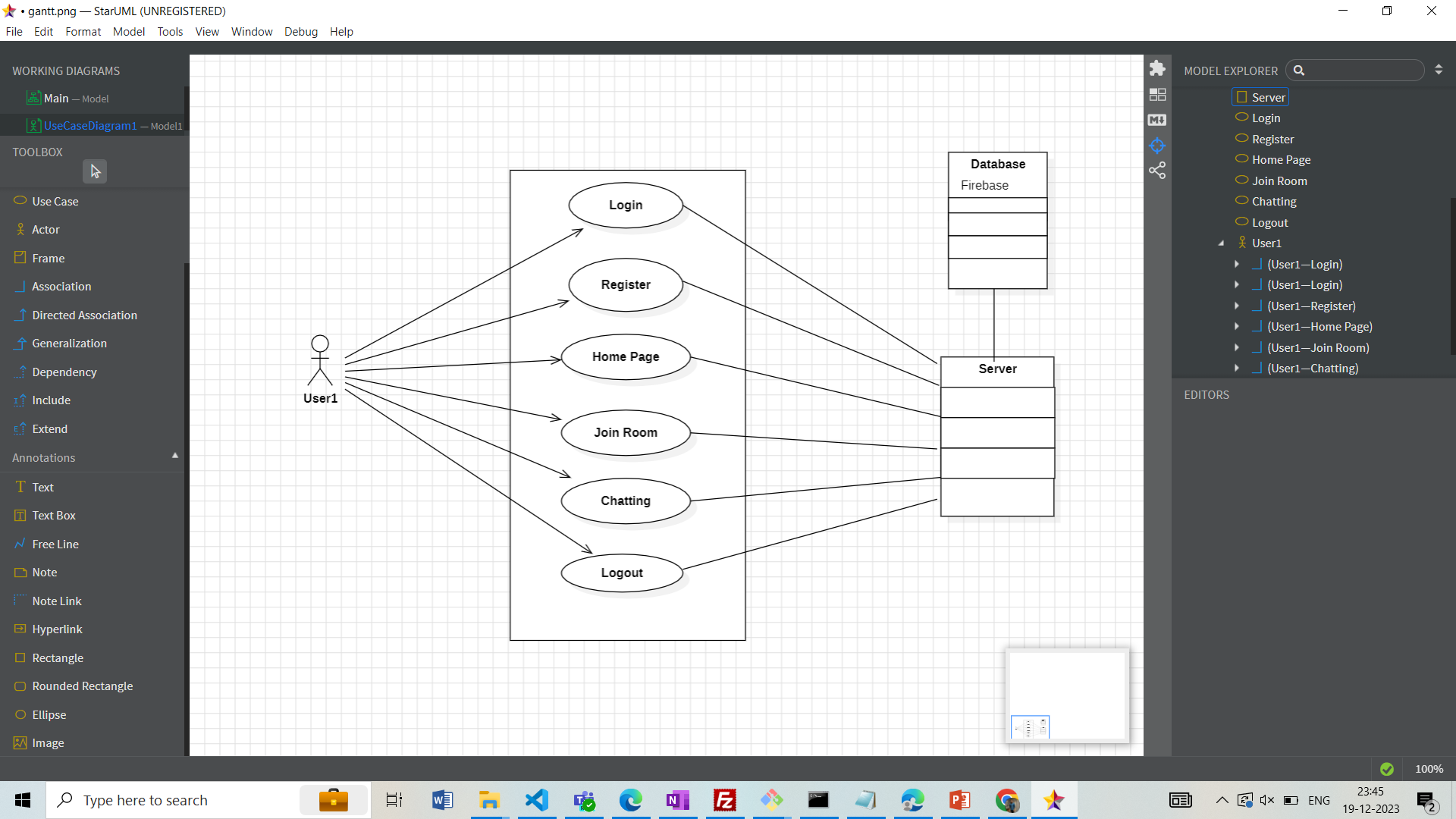
• Version Control: Git and GitHub for source code management

1. **Actual Gantt Chart**

****

1. **System Design**
2. **User Interaction**: Users initiate the process by registering an account. Registered users access the system through secure login credentials. Users have the option to update account details, change passwords, or delete their accounts.
3. **Data Flow**: Users input data through the user interface, including passwords and account details. Passwords undergo encryption using arithmetic encoding before storage. The encrypted data is then stored securely in the MySQL database. When requested, the system retrieves encrypted data from the database. The retrieved data is decrypted using the appropriate algorithms before being presented to the user.

1. **Security Layers**: We have used Arithmetic Encoding to implement the security layer for our project. This helps us to encrypt our message and store it for retrieval for when we need to use it.
2. **UML Use Case Diagrams**

****

**MODULE 4**

1. **Testing and Validation**

**Code Efficiency**

**Testing Approach**

**Unit Testing**

**Integration Testing**

1. **Code Efficiency**

**Arithmetic Encoding Algorithm:**

Understand the arithmetic encoding algorithm thoroughly, and choose an efficient implementation that minimizes the number of arithmetic operations required.

**Error Handling:**

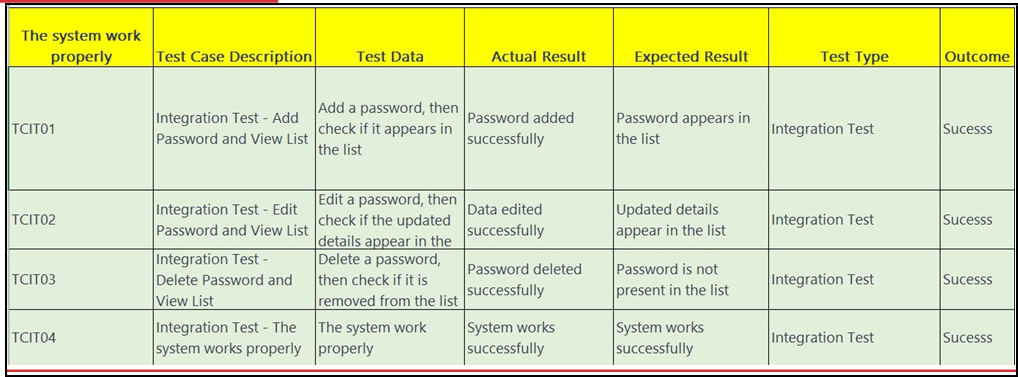
Implement efficient error-handling mechanisms without sacrificing performance. Be cautious about the impact of error checking on the overall efficiency of the encoding/decoding process.

**Platform and Language Optimization:**

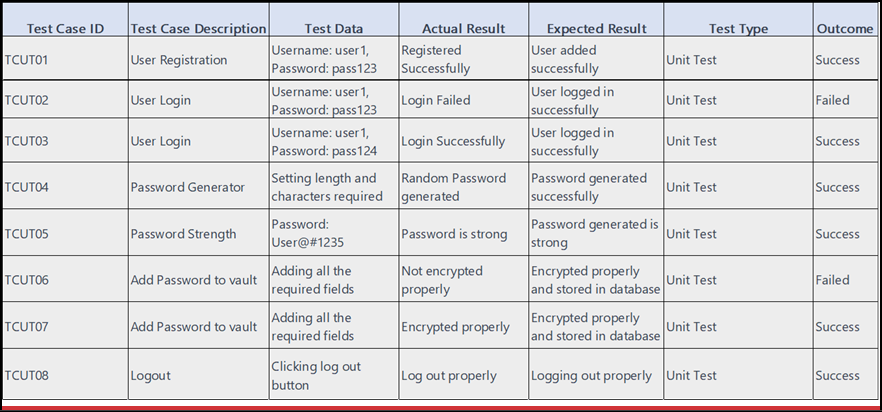
Consider platform-specific and language-specific optimizations. Take advantage of language features or libraries that may provide performance benefits.

1. **Testing Approach**

* Integration Testing



* Unit Testing

****

**MODULE 5**

1. **User Manual**

**User Documentation**

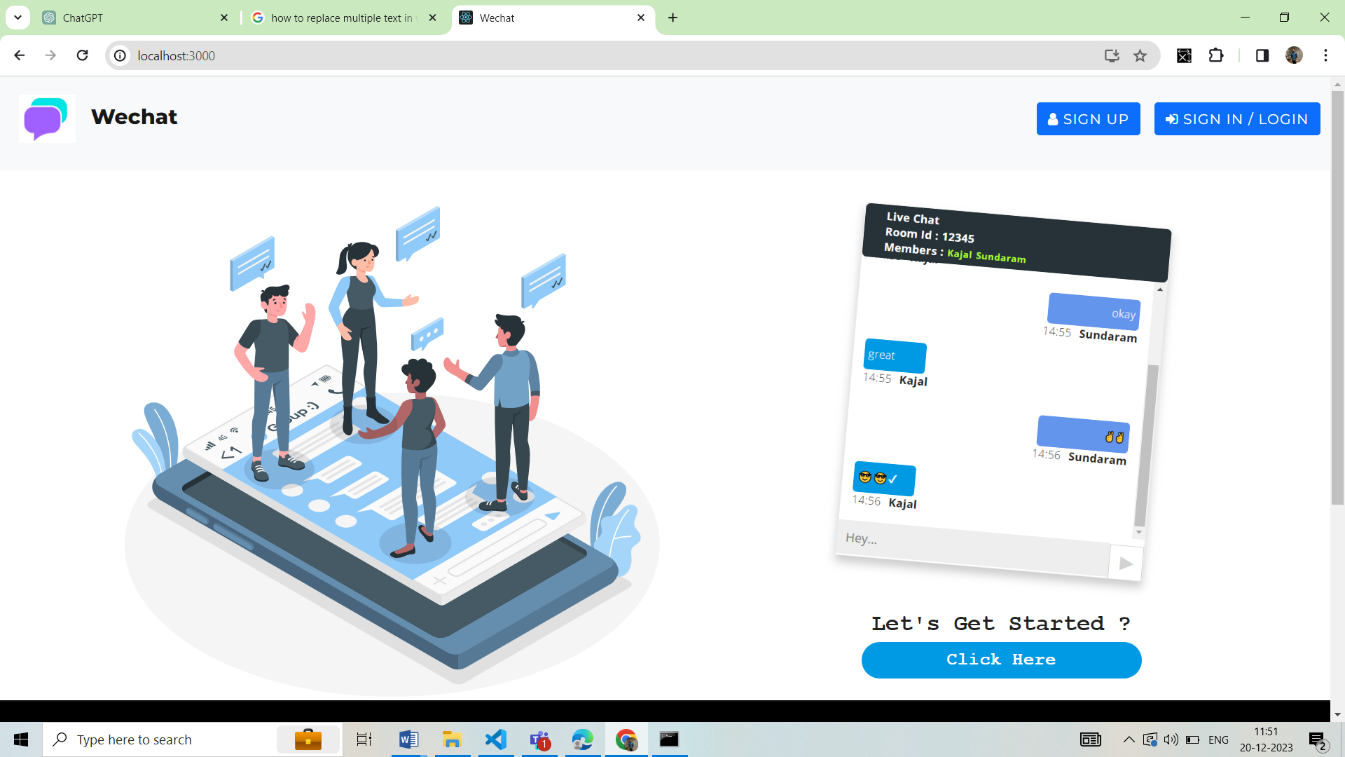
**Outputs**

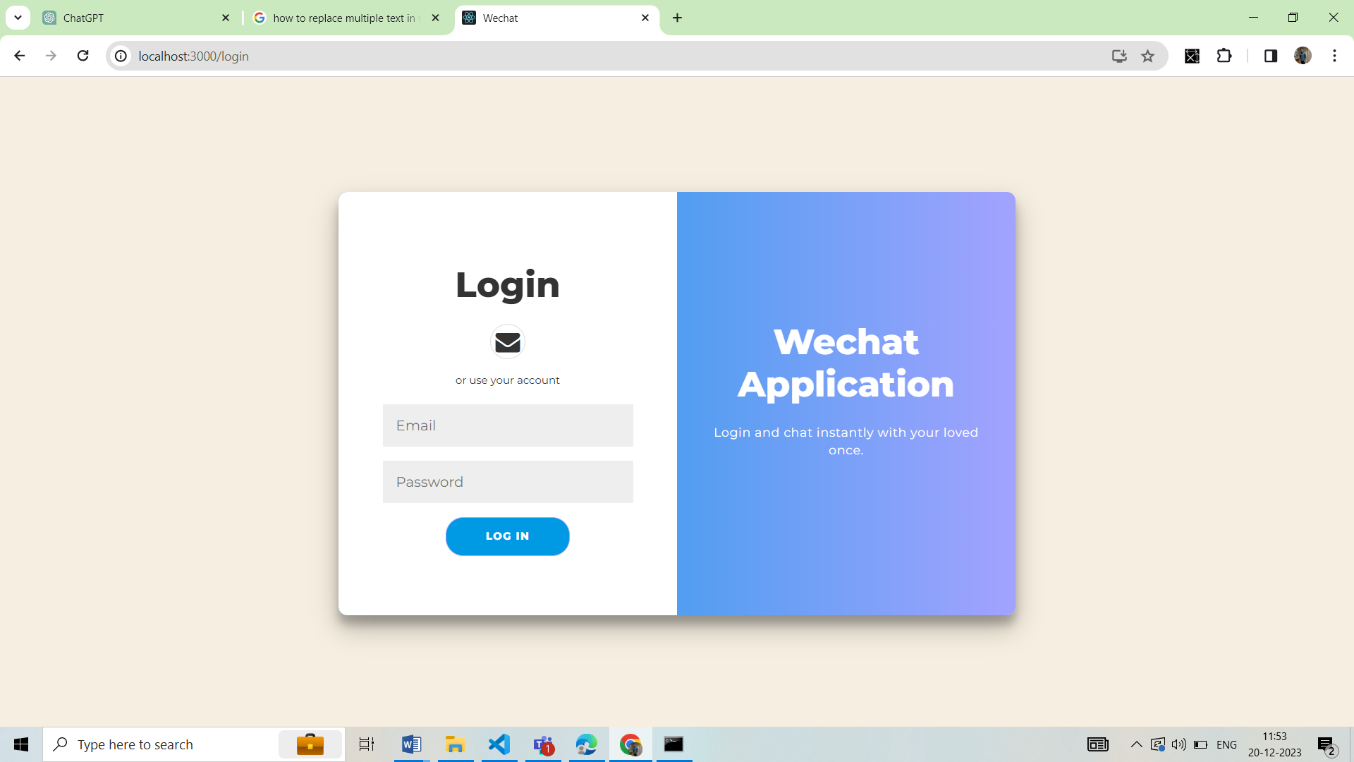
**Code Snippets**

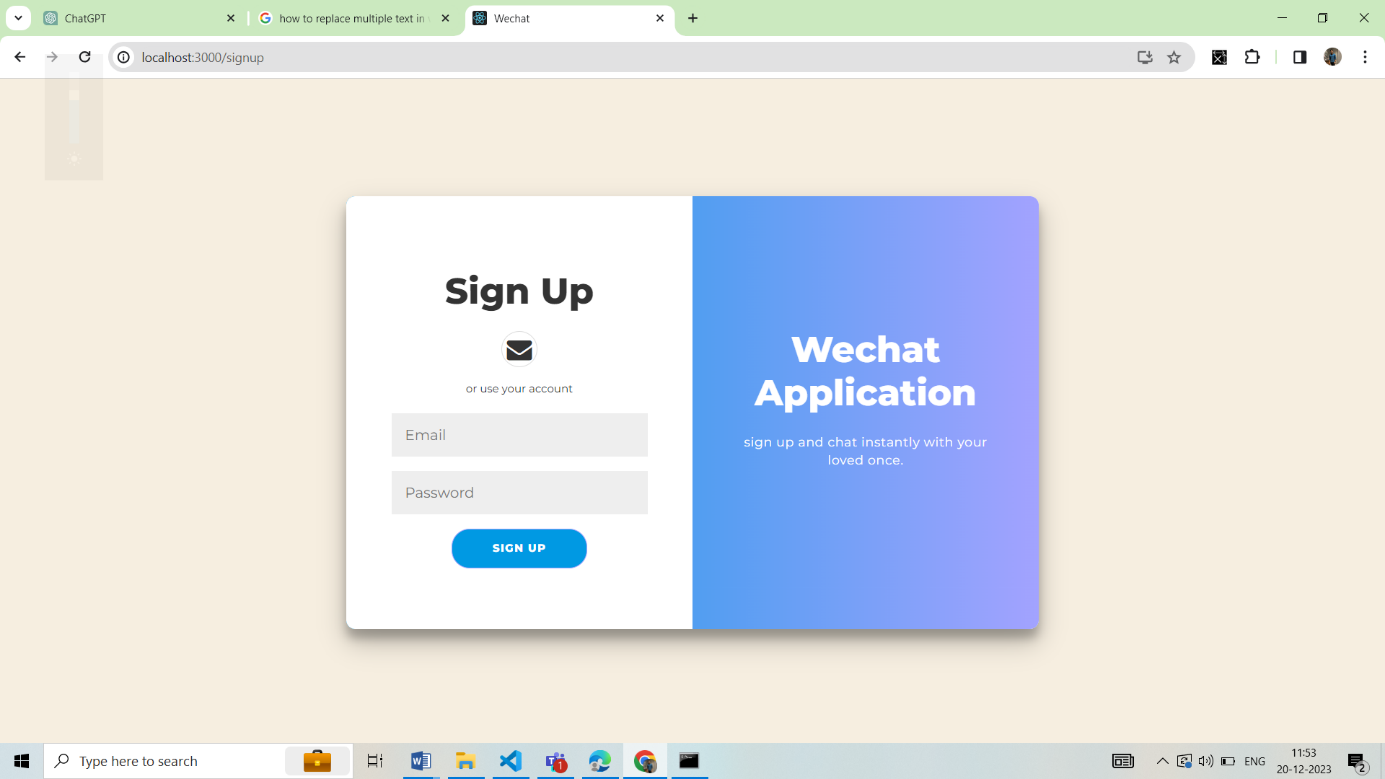
1. **User Documentation**

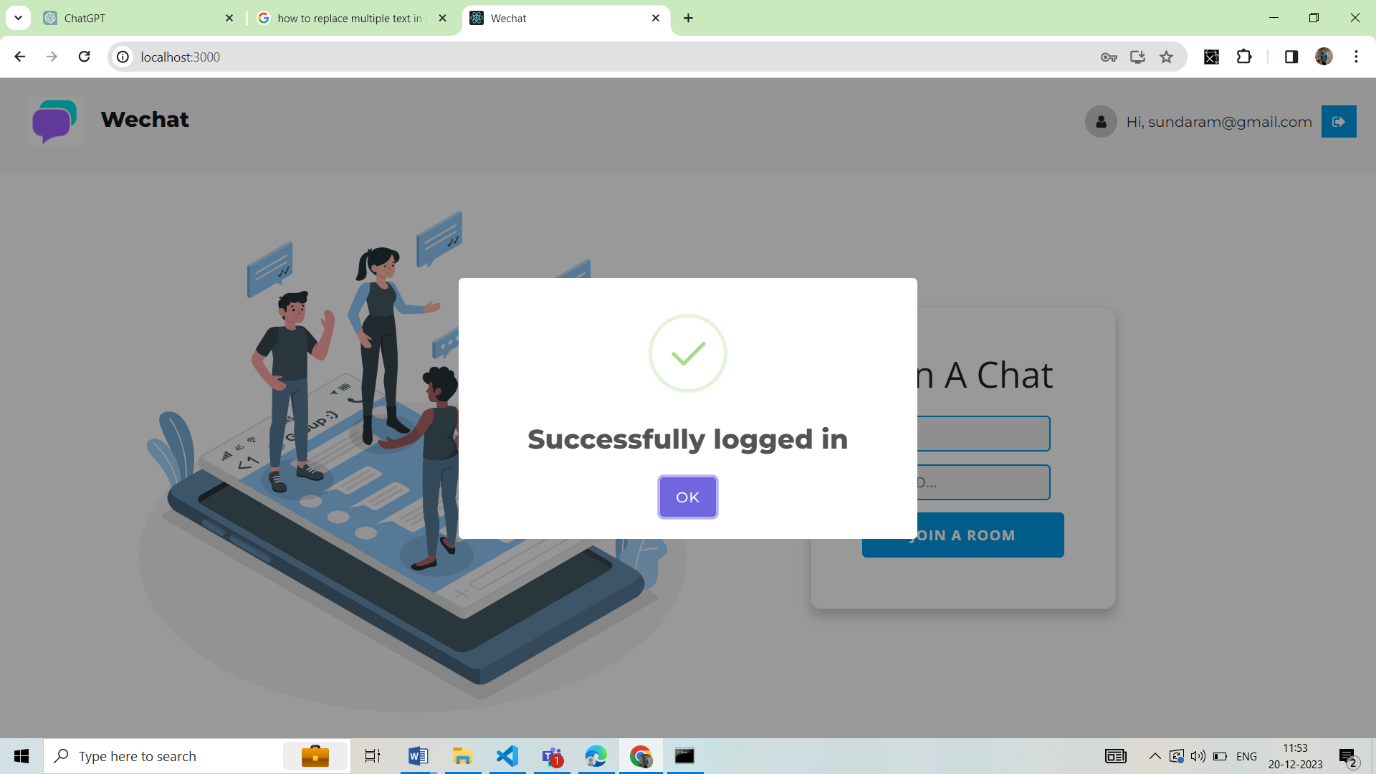
The user need to directly access the page through command npm start to runserver and can start navigating their way through the entire page.

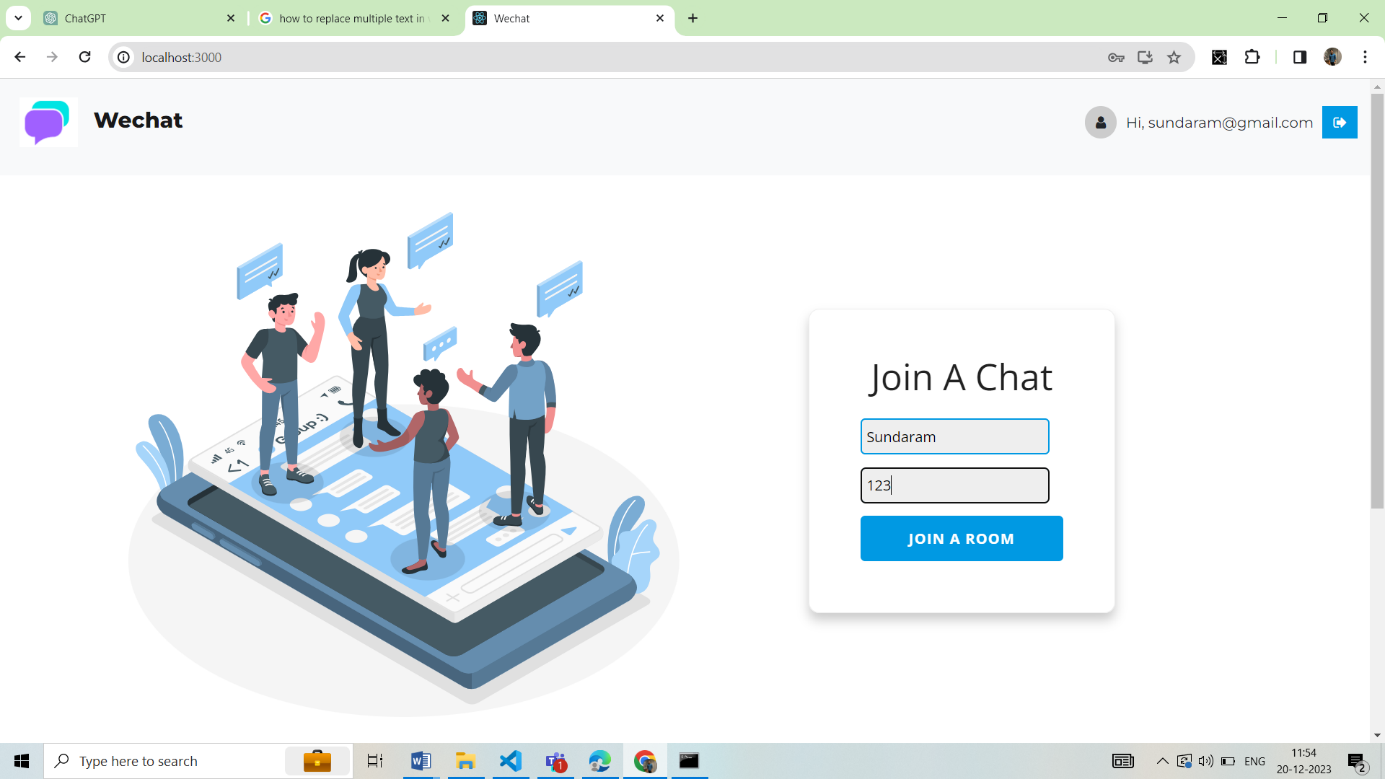
1. **Output**

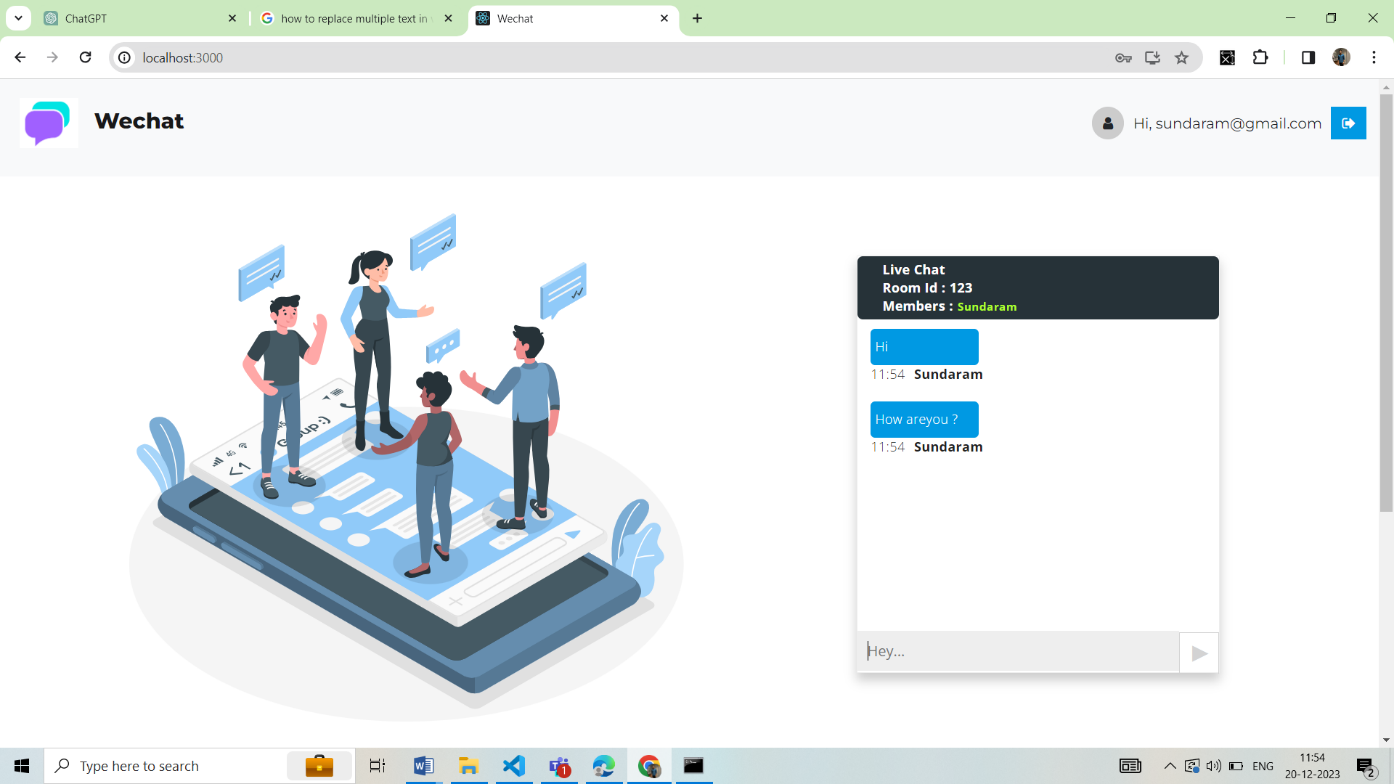
****

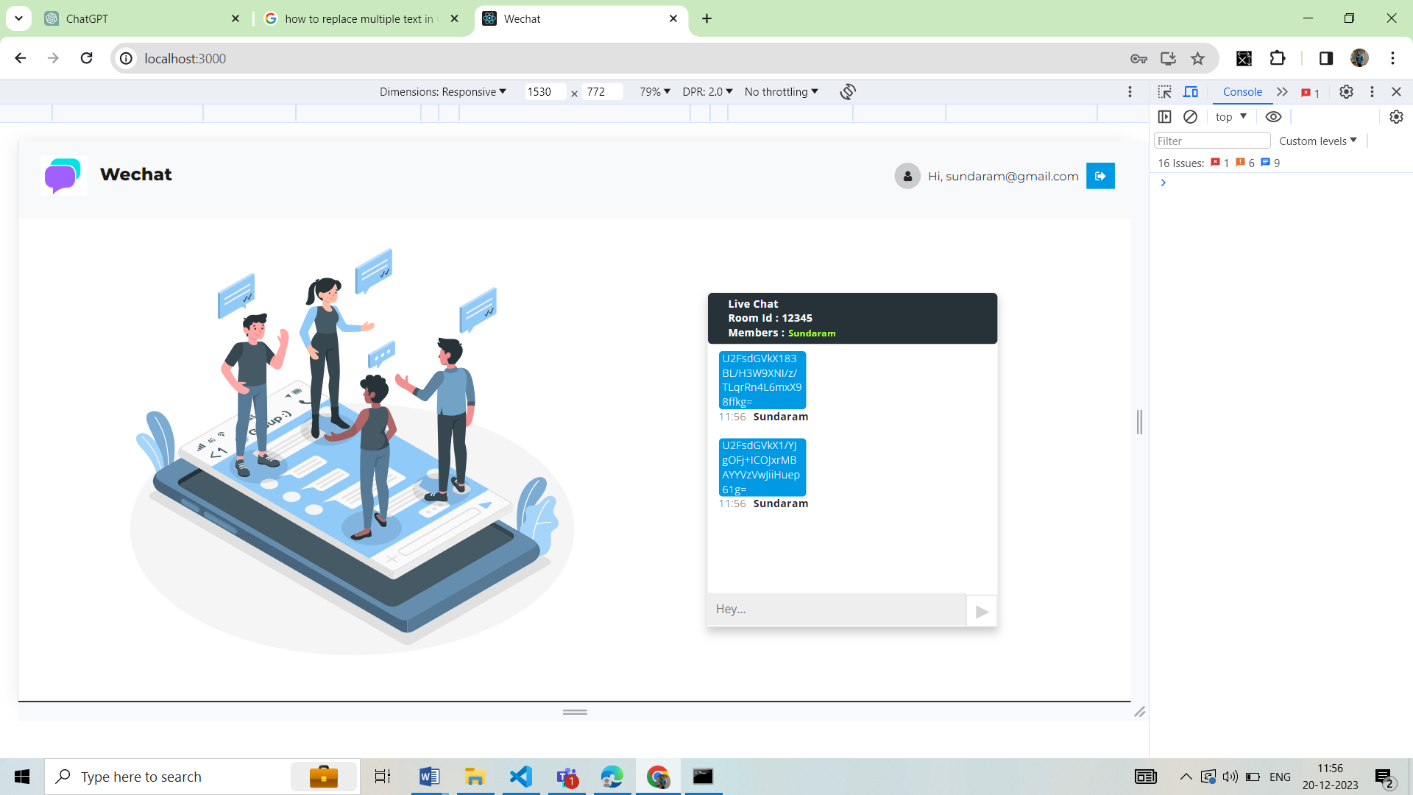
****

****

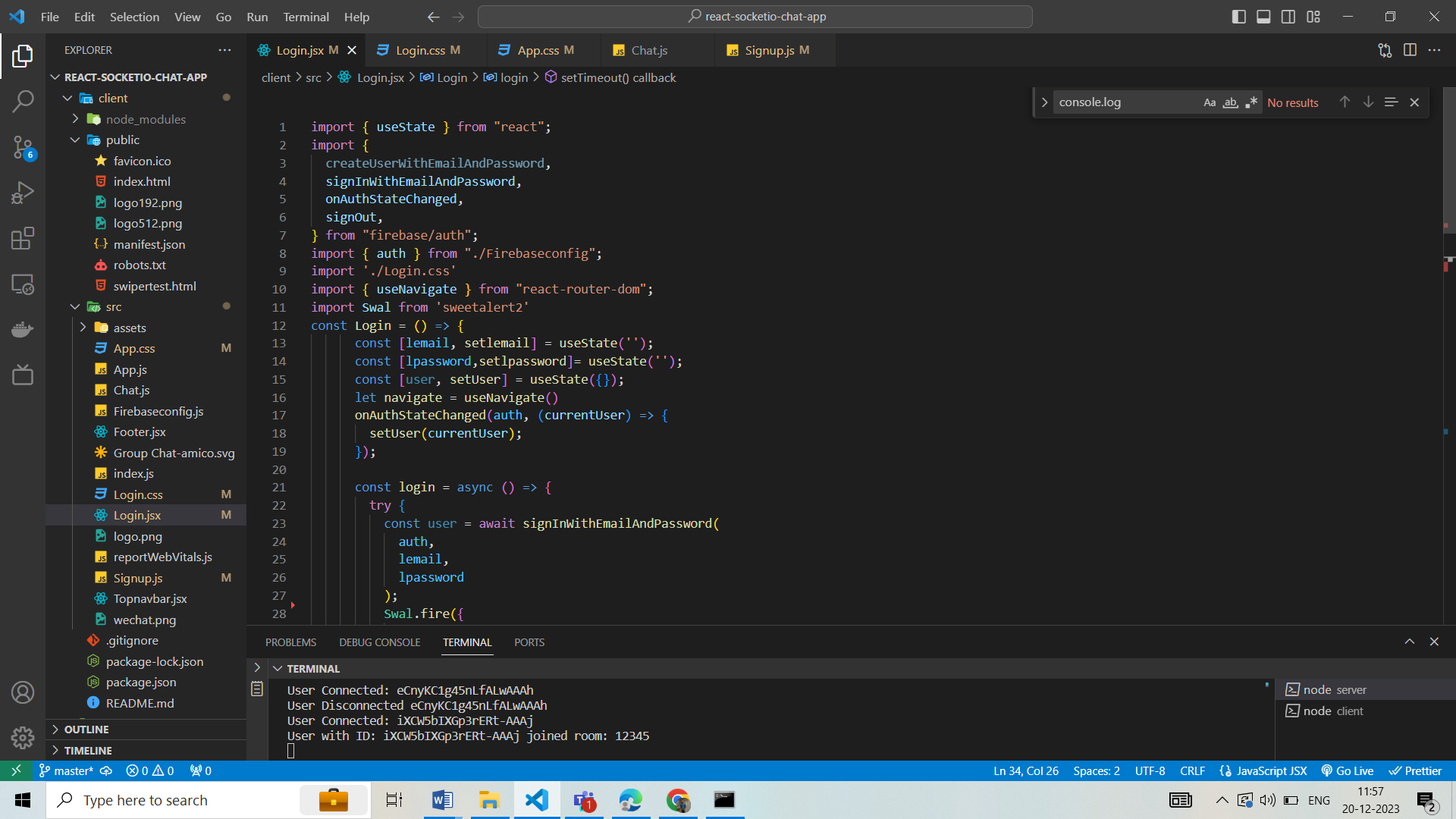
****

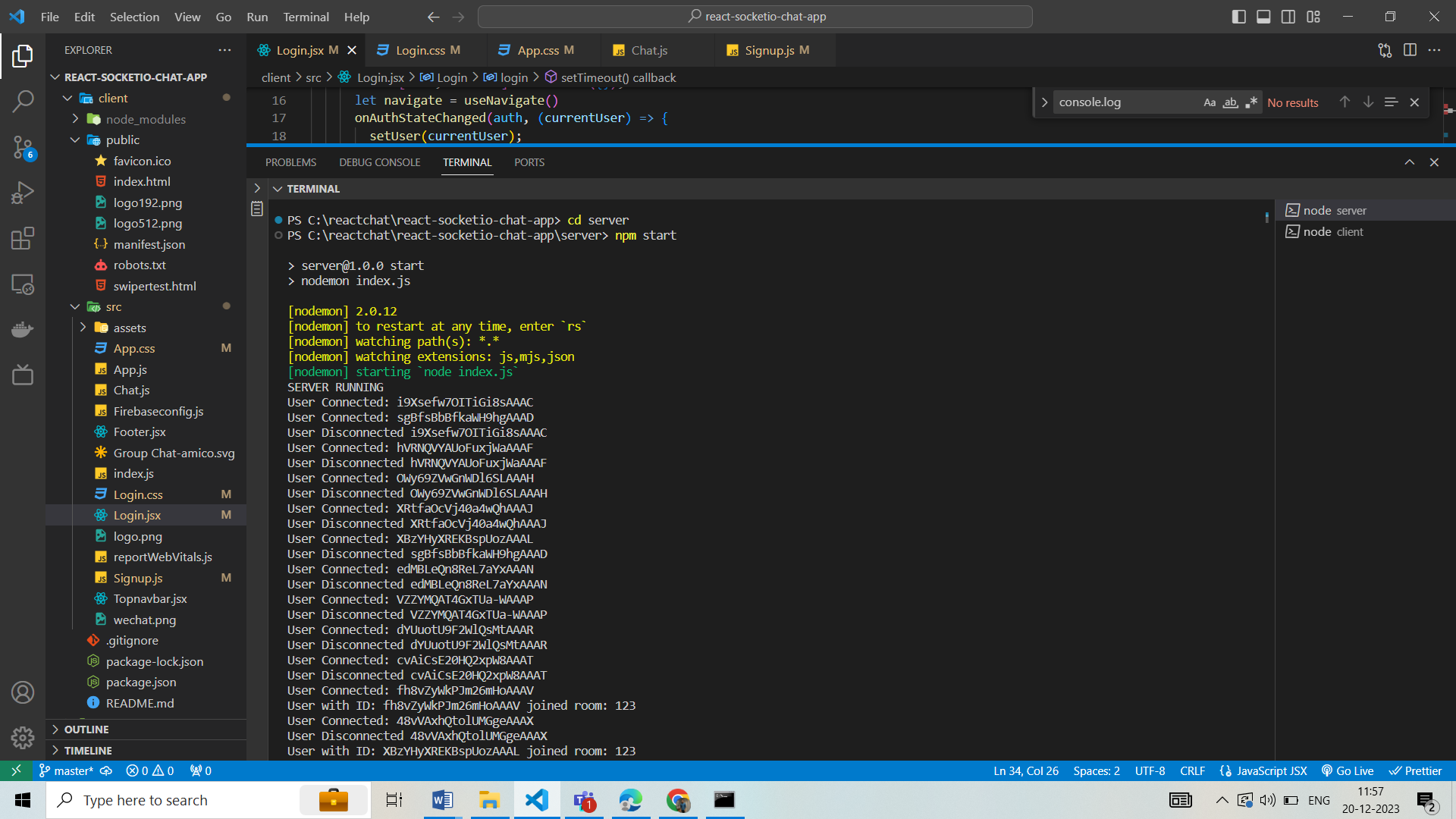
****

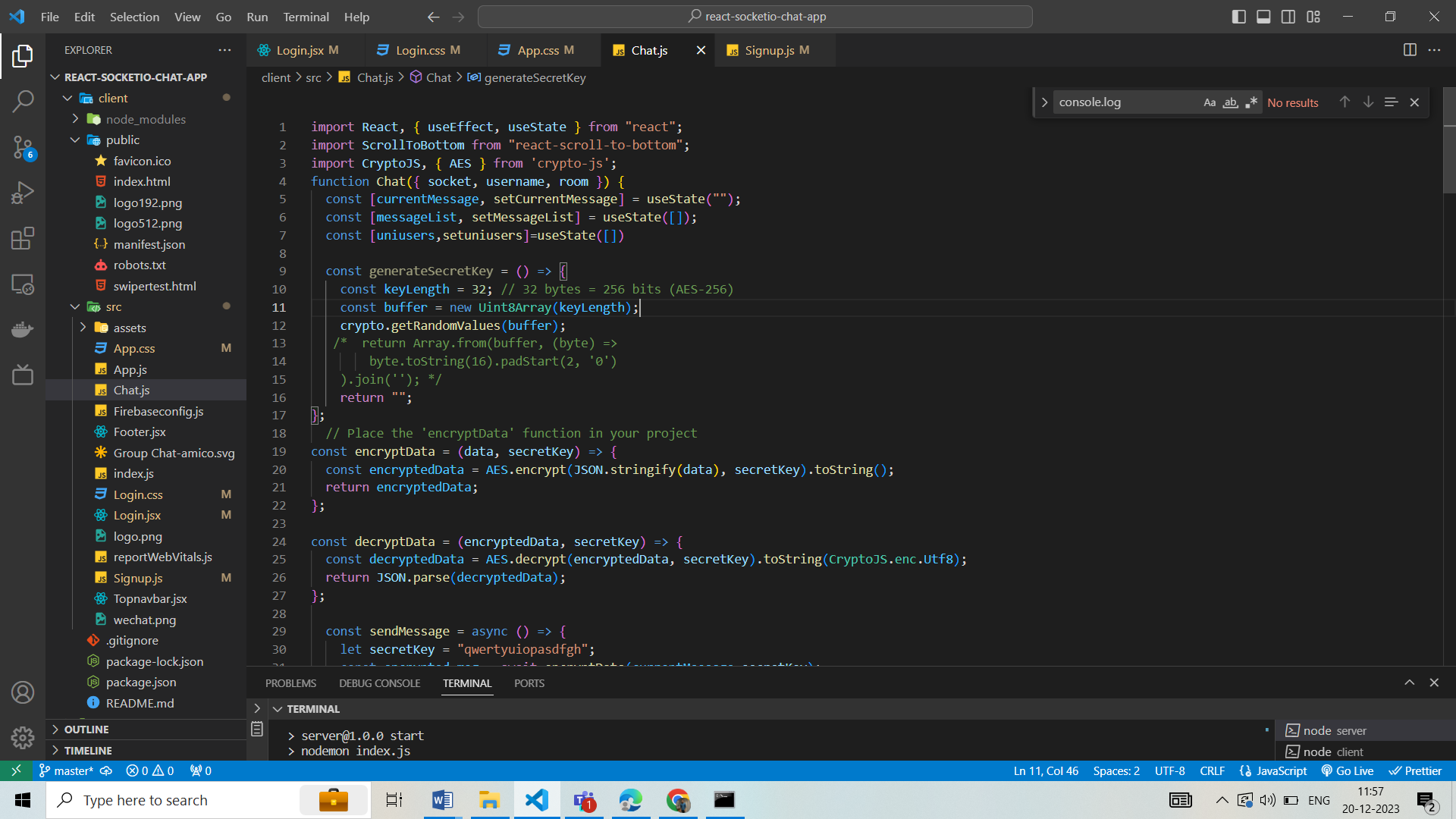
****

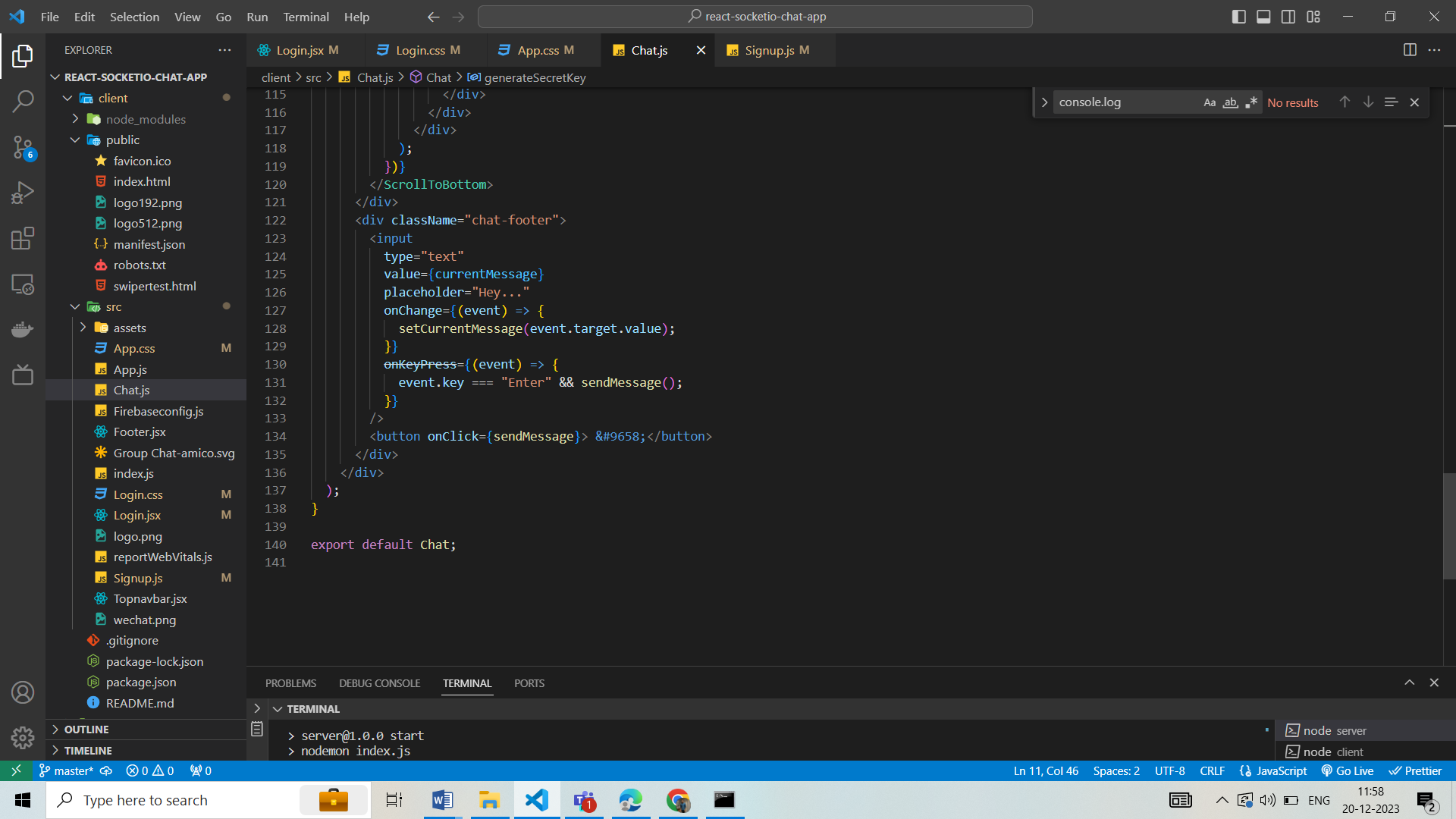
****

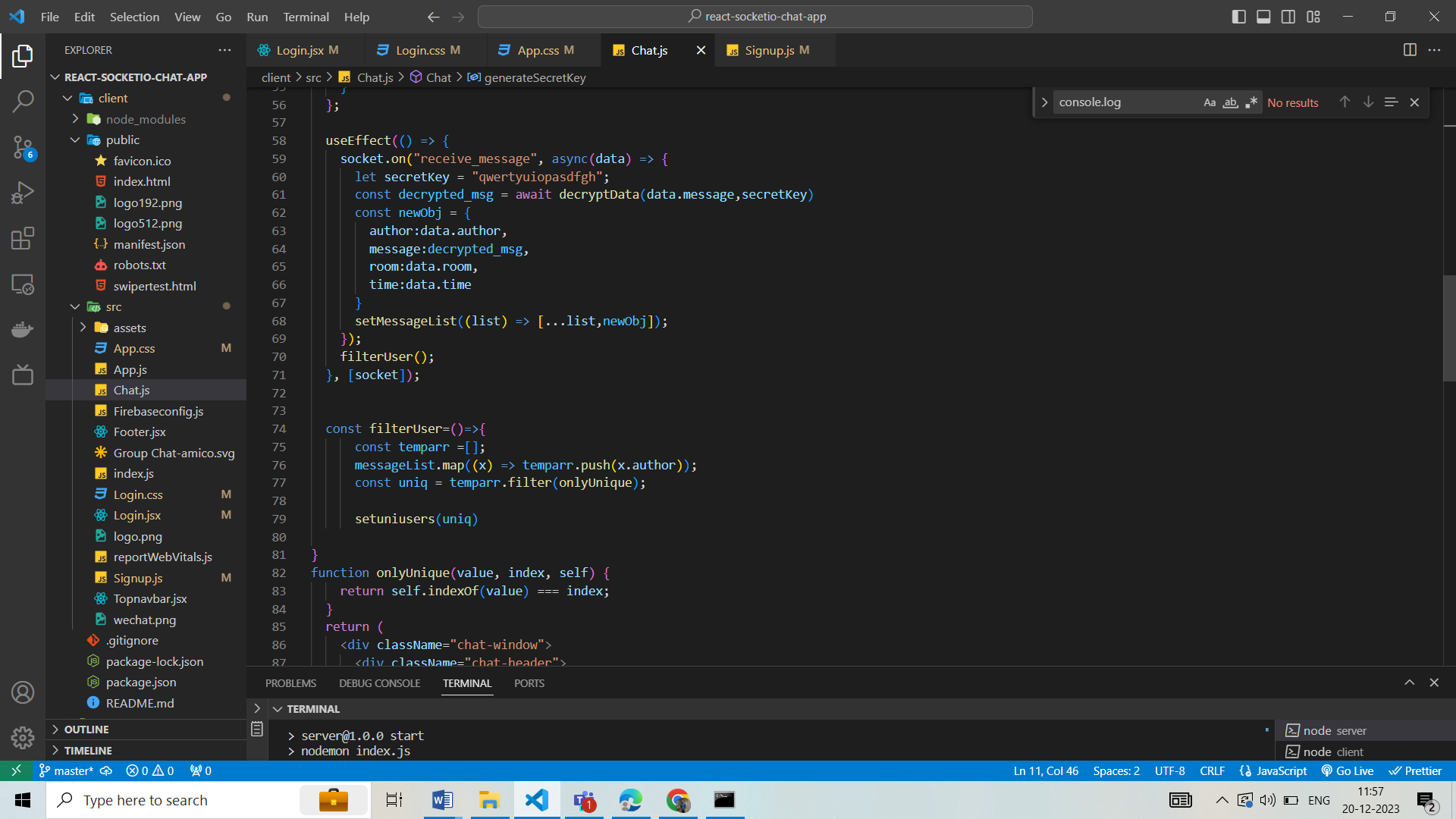
1. **Code Snippets**

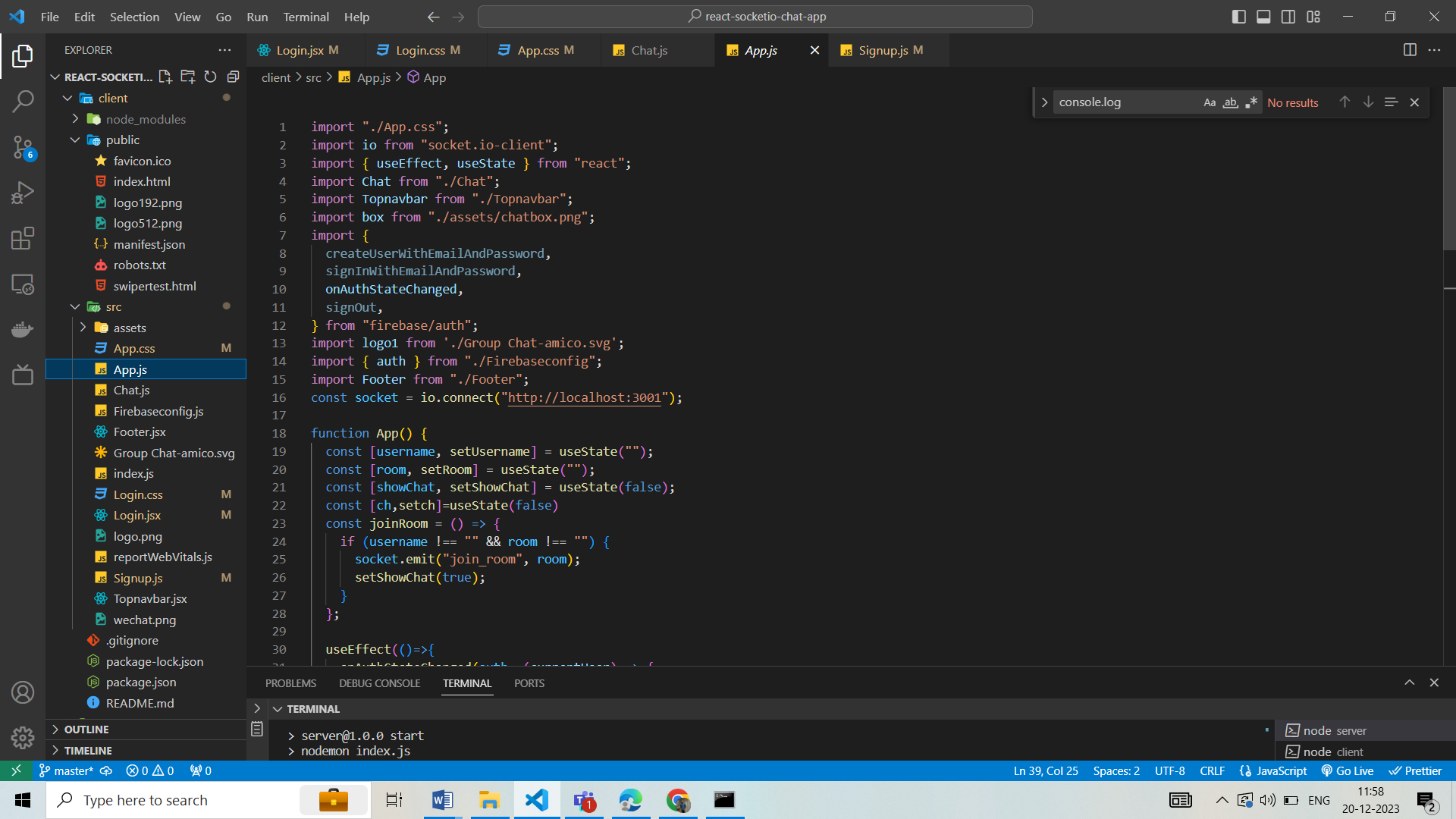
****

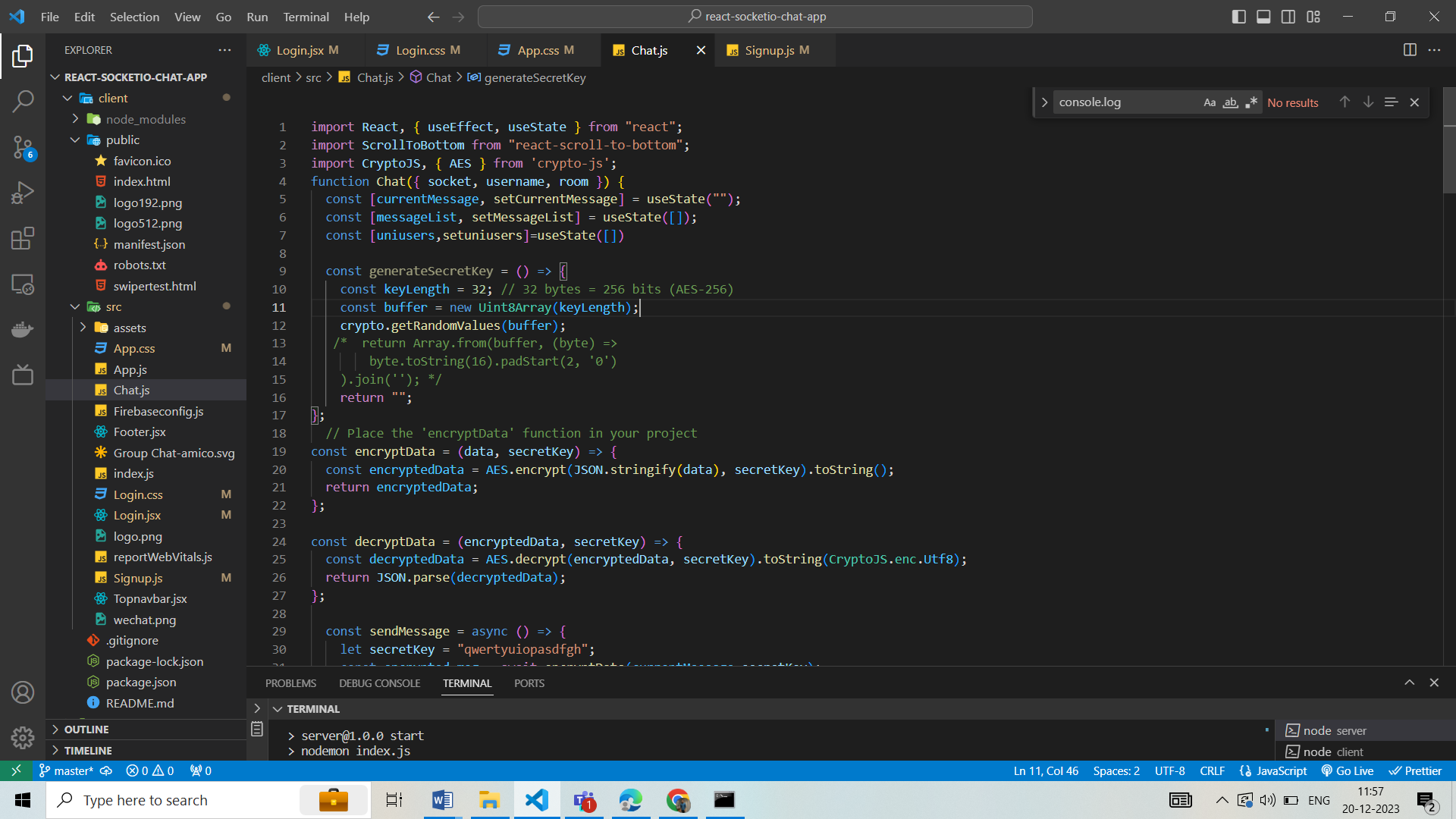
****

****

****

****

****

****

**MODULE 6**

**6.Conclusion**

**Conclusion**

**Limitations**

**Future Scope**

1. **Conclusion :**

The project aimed to address the challenges associated with password security and management by incorporating advanced cryptographic techniques. The system provides a secure and user-friendly environment for storing and managing passwords while leveraging arithmetic encoding for enhanced encryption. Throughout the development lifecycle, rigorous testing and quality assurance measures were implemented to ensure the robustness and reliability of the system. The project adhered to the specified requirements and timelines, resulting in a functional and well-documented password vault. The user authentication module, password management module, backup and recovery module, user interface module, and security module were integrated seamlessly to provide a comprehensive solution. The use of arithmetic encoding adds an extra layer of security to the stored passwords, enhancing the confidentiality and integrity of user data.

1. **Limitations:**

While encryption methods, including arithmetic encoding, are implemented for password security, no system is entirely risk-free. Constant vigilance and updates are necessary to mitigate emerging security threats. If a user forgets their master password recovery options may be limited, potentially resulting in the loss of stored passwords. The centralization of password storage in one system creates a single point of failure. A breach or failure in the system could have significant consequences. Users may have concerns about the privacy and security of their stored passwords, particularly if the system faces a security breach or unauthorized access.

1. **Future Scope Password Expiry Notifications:**

• Provide users with notifications for password expiration, encouraging them to update passwords regularly and adhere to security best practices. Enhanced Backup Options:

• Expand backup and recovery capabilities by incorporating cloud-based solutions, allowing users to sync and access their password vaults across multiple devices. Cross-Platform Compatibility:

• Develop dedicated mobile applications for iOS and Android platforms, ensuring users can securely access their password vaults on various devices.