**JAVA PROJECT REPORT**

(Project Term January-May 2023)

## Aikhyaam Chat: A Comprehensive Secured Messaging Platform for Collaborative and Instant Messaging

Submitted by

**Sriirazz Kolla 12103936**

**Project Group Number:13**

**Course Code: CSE310**

**Under the Guidance of the Faculty**

**Dr. A. Ranjith Kumar**

**School of Computer Science and Engineering**

**A picture containing logo

Description automatically generated**

**DECLARATION**

We hereby declare that the project work entitled (“Aikhyaam Chat: A Comprehensive Secured Messaging Platform for Collaborative and Instant Messaging”) is an authentic record of our own work carried out as a requirement of Project for the award of a B.Tech degree in Computer Science Engineering from Lovely Professional University, Phagwara, under the guidance of Dr. Ranjith Kumar, during January to May 2023. All the information furnished in this project report is based on our own intensive work and is genuine.

Project Group Number: 13

Name of Student: Sriirazz Kolla

Registration Number: 12103936

(Signature of Student)

Date:

**TABLE OF CONTENTS**

**Inner first page…………….………………………………………..….01**

**Declaration….......………………………………………………..……..02**

**Table of Contents…......……………………………………..…………03**

1. **Introduction --------------------------------------------------------------------04**
   1. **Scope of the project ------------------------------------------------------05**
2. **Propose Technology-----------------------------------------------------------05**
   1. **Login In module ----------------------------------------------------------07**
   2. **Sign Up module -----------------------------------------------------------09**
   3. **Homepage Module -------------------------------------------------------11**
   4. **ChatPage Module --------------------------------------------------------12**
3. **Code -----------------------------------------------------------------------------13**
4. **Conclusion ---------------------------------------------------------------------17**
5. **Future Enhancements -------------------------------------------------------18**

**1.INTRODUCTION:**

Communication is essential in the fast-paced world of today. With the development of technology, people may now communicate with people around the world by simply tapping on their displays. Because they enable real-time collaboration, communication, and sharing of views and ideas, chat programs have grown in popularity.

Our chat program is created to offer consumers a fluid communication experience. Our platform has a few features that make it simple and quick for users to connect with their peers, family, and co-workers. Users can speak with numerous individuals at once via chatrooms, which makes them ideal for organizing events or debating projects with co-workers. Users can communicate privately with others through private messaging, ensuring Users are able to connect privately and anonymously with one another through private messaging.

We recognize the value of media in communication, which is why our platform enables users to exchange images. This tool is especially helpful for sending images from a recent vacation to friends and family or for sharing vital documents.

Our chat program is extremely simple to use, making it accessible to people of every age and professional skill set. We have made sure that the user interface is clear and that device like computers can access our platform.

In conclusion, the goal of our chat application is to give consumers a pleasant communication experience. Our chat application provides everything you need to make communication quick, simple, and entertaining if you're catching up with loved ones, carrying out a task, or just chatting with new friends. So, sign up now and start talking through us!

**1.1 SCOPE OF PROJECT:**

The goal of this chat application project is to give users a platform for real-time communication. This offers tools like private messaging, media sharing, group conversations, and an easy-to-use interface. The purpose of the application is to link people around the world and make it simpler for them to communicate with one another.

The project's scope involves creating the application from scratch, testing it carefully to ensure that it functions as intended, and launching it across a number of different platforms, including mobile devices and the web. The application needs to be safe, with safety measures in place to safeguard user information and privacy. Also, the program must have flexibility in order to support a growing user base and add new features whenever needed.

**2. Proposed Technique**

The proposed technique for a chat application built using Java Swing library, MySQL database, and SHA256 hashing for credentials storage would involve the following steps:

1. User Authentication:

a. The user enters his/her credentials (i.e., username and password) on the login screen.

b. The password field is hashed using the SHA256 algo rithm before sending it to the MySQL database for validation against the stored hashed password.

c. Upon successful validation, the user is granted access to the chat application.

2. User Registration:

a. Upon registration, the user's credentials (i.e., name, username, password) are stored in the MySQL database.

b. The password is hashed using SHA256 before being stored in the database.

3. Chat History:

a. The messages sent and received by the user are never stored in the MySQL database.

b. The messages are never extremely secure and cannot be intercepted by anyone even the team is unaware of your communication.

4. Real-time Chat:

a. The messaging feature of the application is implemented using Java Swing library and socket programming to establish real-time communication between users.

b. The messages are sent and received in real-time.

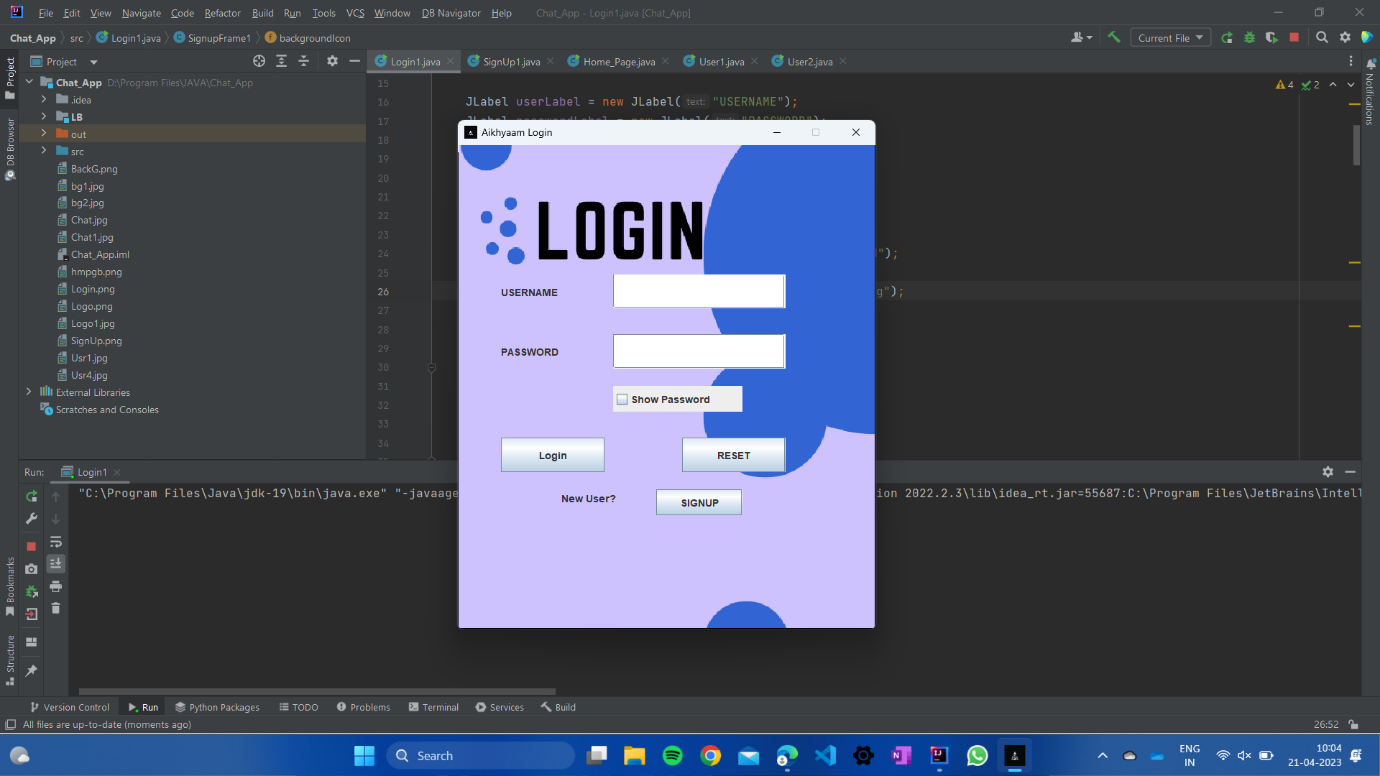
5. Secure Communication:

a. All credentials of the users are encrypted using the SHA256 hashing algorithm to ensure data security and privacy.

b. The hashed passwords stored in the database ensure that even if the database is compromised, the actual passwords cannot be retrieved by the attacker.

Overall, the chat application built using MySQL, Java Swing, and SHA256 hashing provides a secure and reliable platform for real-time communication between users while ensuring the privacy and confidentiality of their information.

**2.1 Login-In Module:**



The Java Swing login page for the chat application that accepts user credentials and validates them against the MySQL database has the following features:

1. User Authentication:

The user enters their credentials (i.e., username and password) in the text fields provided, and the password field has a checkbox option to 'Show Password' which can be selected to view the entered password in clear text format. The login button is used to submit the entered credentials, and upon submitting the login request, the password is hashed using the SHA256 hashing algorithm.

2. Validation against MySQL Database:

Upon submitting the credentials, the entered username and hashed password are checked against the MySQL database for validation. If the entered credentials match those stored in the MySQL database, the user is redirected to the homepage of the chat application. If not, the user receives an error message that their credentials are invalid.

3. Sign-up Option:

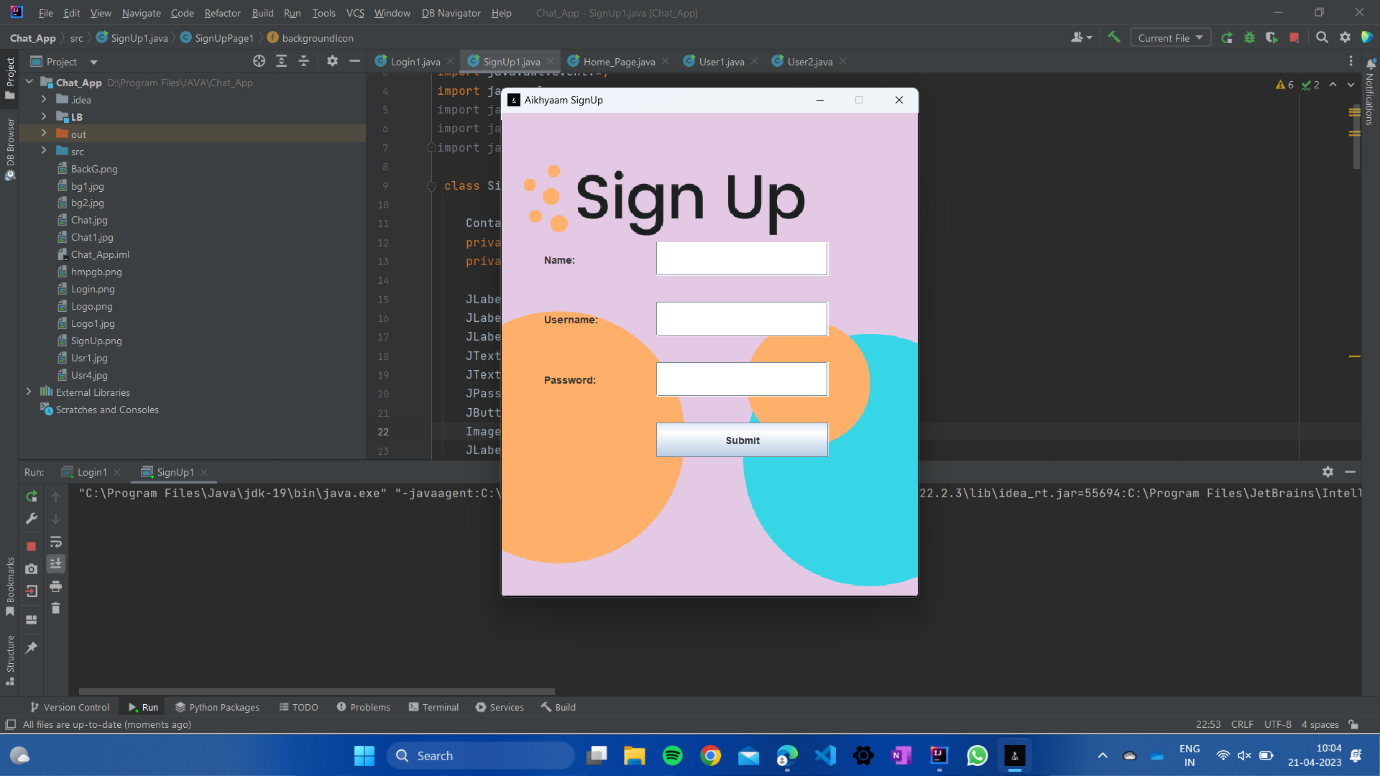
If the user is new to the chat application and does not have a registered account, a 'Sign-up' button is provided that will redirect them to the registration page of the chat application.

4. Reset Button:

If the user wants to clear the text fields and reset their credentials, a 'Reset' button is included that clears the text fields and allows the user to start again from scratch.

The Java Swing login page is a crucial module in the chat application, as it validates user credentials and ensures that only authorized users can access chat rooms and interact with other users. By using SHA256 hashing to encrypt passwords, the application ensures that user passwords are secure and cannot be accessed by unauthorized entities. The additional features of the 'Show Password' checkbox, 'Sign-up' button, and 'Reset' button enhance user experience and make it easy for users to navigate the login page.

**2.2 Sign-Up Module:**



The sign-up module for the chat application that accepts user inputs and stores the account details in the MySQL database has the following features:

1. User Input:

The user enters their details, including their full name, username, and password, in the text fields provided. The password field is hashed using the SHA256 hashing algorithm to ensure secure storage of the password in the MySQL database.

2. Validation Against MySQL Database:

The sign-up module verifies whether the entered username is already present in the MySQL database. If the entered username already exists in the database, the user receives an error message that the account already exists, and that they should choose a different username. However, if the entered username is unique, the user account is created, and the account details are stored in the MySQL database.

3. Secure Storage of Password:

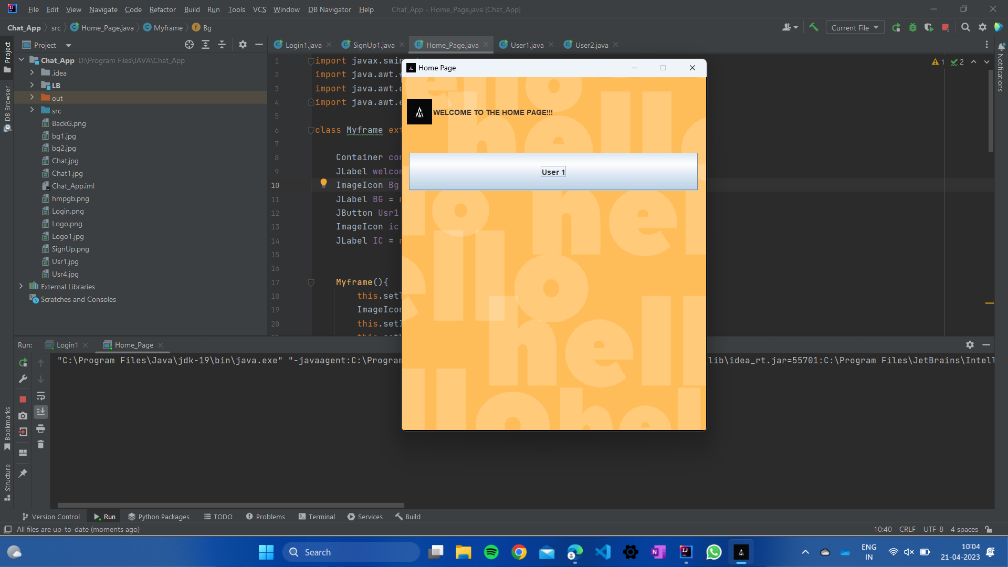
Upon submitting the sign-up request, the password is hashed using the SHA256 algorithm before being stored in the MySQL database to ensure secure storage of the password.

4. Successful Account Creation:

Upon successful registration, the user is redirected to the login page of the chat application, where they can enter their credentials to access the chat rooms and interact with other users.

The sign-up module is an essential part of the chat application as it allows new users to create accounts and join the chat rooms to interact with other users. By using SHA256 hashing to encrypt passwords, the application ensures that user passwords are secure and cannot be accessed by unauthorized entities. The validation check ensures that new users cannot create duplicate accounts, preventing the overloading of user accounts in the MySQL database. It also enhances user experience by providing clear error messages, redirecting newly registered users to the login page, and allowing them to start using the chat application right away.

**2.3 Homepage Module:**



The homepage module of a chat application can be implemented using socket programming and the Java Swing library. The module would consist of a list of online users who are currently connected to the chat server, and would allow users to start new conversations with these online users.

To implement the module using socket programming, the chat server would need to maintain a list of currently connected users and broadcast their status to other users whenever they log in or log out. This could be done using TCP sockets and a custom messaging protocol, where the server sends messages to all connected clients whenever a new user logs in or an existing user logs out.

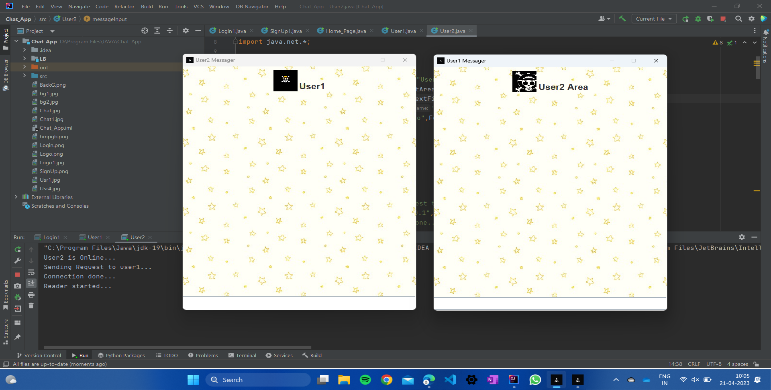
On the client-side, the homepage module would need to periodically check for new messages from the server and update the list of online users accordingly using Java Swing. This could be done using a background thread that listens for new messages from the server and updates a Java Swing component such as a JList or JTable to display the list of online users.

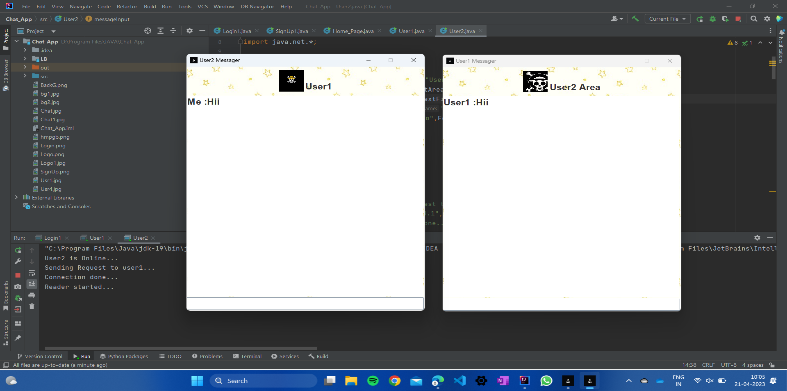
The Java Swing component for displaying the list of online users could be customized to fit the needs of the chat application, for example by adding images or icons to represent each user or by color-coding the list to indicate which users are online or offline. The component could also provide a way for the user to start a new chat with one of the online users, either by double-clicking on their name in the list or by using a context menu or button.

To implement the module using Java Swing, the client-side code would need to use various Swing components such as JFrame, JPanel, JLabel, JList, or JTable to display the list of online users and handle user interactions. The code would also need to use socket programming APIs such as Socket, ServerSocket, BufferedReader, and PrintWriter to communicate with the chat server and retrieve the list of online users.

Overall, the homepage module for a chat application using socket programming and Java Swing is a powerful and flexible way to display the list of online users and allow users to start new conversations with them. It requires a solid understanding of both socket programming and Java Swing and can be customized to fit the specific needs of the chat application.

**2.4 Chat Page Module**





The chat page module of a chat application can be implemented using the Java Swing library and socket programming. This module allows users to send text messages from one user to another user in real-time.

To implement the chat page module, a user interface using Java Swing components such as JFrame, JPanel, JTextArea, JTextField, and JButton can be created. The JFrame will serve as the main container of the user interface, the JPanel will contain the JTextArea and the JTextField, whereas the JButton will be used to send the message. When a user enters text in the JTextField, and clicks the send button, the message is dispatched to the chat server.

To implement the module using Java Swing and socket programming, both the client-side and server-side code would need to use APIs such as Socket, ServerSocket, BufferedReader, and PrintWriter to communicate with each other over the internet. The code would need to handle different events and exceptions that could arise while sending and receiving messages to and from the server.

Overall, the chat page module is an important component of a chat application that allows users to send messages to each other in real time. It requires a solid understanding of both the Java Swing library and socket programming and can be customized to fit the specific needs of the chat application.

**3. Code**

**Sign In Page**

public void actionPerformed(ActionEvent e) { if (e.getSource() == loginButton) { String userText = userTextField.getText(); String pwdText = passwordField.getText(); MessageDigest messageDigest = null; try { messageDigest = MessageDigest.getInstance("SHA-256"); } catch (NoSuchAlgorithmException ex) { throw new RuntimeException(ex); } byte[] hash = messageDigest.digest(pwdText.getBytes(StandardCharsets.UTF\_8)); StringBuilder hexString = new StringBuilder(2 \* hash.length); for (byte element : hash) { String hex = Integer.toHexString(0xff & element); if (hex.length() == 1) { hexString.append('0'); } hexString.append(hex); } String url = "jdbc:mysql://localhost:3306/UserLogin"; String username = "root"; String password = "Muni"; String query = "SELECT UserPassword FROM UserLogin WHERE UserId = ?"; boolean isAuthenticated = false; try (Connection con = DriverManager.getConnection(url, username, password); PreparedStatement ps = con.prepareStatement(query)) { ps.setString(1, userText); ResultSet rs = ps.executeQuery(); if (rs.next()) { String storedPassword = rs.getString("UserPassword"); if (storedPassword.equals("")){ isAuthenticated=false;}else if (storedPassword.equals(hexString.toString())) { isAuthenticated = true; } } } catch (SQLException ex) { ex.printStackTrace(); } if ( isAuthenticated) { JOptionPane.showMessageDialog(this, "Successful Login Redirecting to Home Page"); this.setVisible(false); Myframe c=new Myframe(); c.setVisible(true); } else { JOptionPane.showMessageDialog(this, "Invalid username or password"); }}

The Sign In page is build using jswing library and connected to my sql database using mysql library. It validates the user credentials with the credentials stored in database. If there’s any mismatch it throws an error message which displays a message box that credentials are wrong. If the credentials are matched the user will be forwarded to the homepage. It also displays a sign-up button in case a new user wants to create an account.

**Sign Up Page**

public void actionPerformed(ActionEvent e) { if (e.getSource() == submitBtn) { String name = nameField.getText(); String username = usernameField.getText(); String password = new String(passwordField.getPassword()); if (name.equals("") || username.equals("") || password.equals("")) { JOptionPane.showMessageDialog(this, "Please enter all fields."); return; // stop execution if any field is empty } String url = "jdbc:mysql://localhost:3306/UserLogin"; String query = "INSERT INTO UserLogin(UserId, UserPassword) VALUES(?, SHA2(?,256))"; try (Connection con = DriverManager.getConnection(url, "root", "sriirazz"); PreparedStatement ps = con.prepareStatement(query)) { ps.setString(1, username); ps.setString(2, password); int rowInserted = ps.executeUpdate(); JOptionPane.showMessageDialog(this, "User created successfully!"); } catch (SQLException ex) { JOptionPane.showMessageDialog(this, "Error creating user."); if (ex instanceof SQLIntegrityConstraintViolationException) { JOptionPane.showMessageDialog(this, "Username already exists."); } else { ex.printStackTrace();}} }}}

The signup page follows the same structure as the sign in page in terms of Gui utilized from JSwing library. The page is also connected to database that saves the credentials of the user in the database with enhanced protection and security. If the username is already taken by previous user, it displays a error message consists of a message “username already exists”.

**User 1 Page:**

public User1() { try { server = new ServerSocket(7777); System.out.println("User 1 is ready to accept connection"); System.out.println("Waiting...."); socket = server.accept(); br = new BufferedReader(new InputStreamReader(socket.getInputStream ())); out = new PrintWriter(socket.getOutputStream()); createGUI(); handleEvents();

startReading(); }catch(Exception e){ e.printStackTrace(); }}

**User 2 Page:**

public User2(){ try{ System.out.println("Sending Request to user1..."); socket = new Socket("127.0.0.1",7777); System.out.println("Connection done...");

br = new BufferedReader(new InputStreamReader(socket.getInputStream())); out = new PrintWriter(socket.getOutputStream()); createGUI(); handleEvents(); startReading();}catch(Exception e){ e.printStackTrace();}}

The user pages build using the same structure. They communicate over the web using socket programming. Users are allowed to chat after the validation of their credentials only .

**4. Conclusion**

In conclusion, the development of the chat application project was a challenging task for our team, but we managed to overcome the obstacles and create a functional and user-friendly application. This project not only helped us improve our technical skills, but also taught us valuable teamwork and communication skills. Our approach to developing the chat application involved utilizing a range of modern tools and technologies to create a functional and user-friendly application that met the project requirements. We utilized the JSwing library for the user interface design, which allowed us to create a visually appealing design that was intuitive and easy to use. To ensure the security of the application, we implemented SHA 256 hashing, a widely recognized method for data encryption. For communication purposes, we used socket programming which played a critical role in facilitating communication between the clients and the server. Finally, to store user data, we used MySQL,

which is a widely-used Relational Database Management System (RDBMS) that provided excellent performance and scalability.

Overall, utilizing these technologies in our project resulted in an application that matched the current industry standards while being reliable and easy-to-use. Our team learnt valuable technical and interpersonal skills while developing the chat application, which will be beneficial in our future projects. We believe that this experience will be a great addition to our portfolio and will help us have a technical edge in our career ahead. We are proud of what we have achieved and look forward to making further improvements to the application in the future.

**5. Future Enhancements**

Moving forward, we envision several potential enhancements to the chat application that could bring significant value to users. First and foremost, text-heavy messaging applications like ours have become somewhat outdated, and we believe our users would benefit greatly from the inclusion of other file formats such as images, videos, and other media. This would enhance the interactivity of the application and allow users to share and collaborate in new ways.

We recognize that security is a top priority for our users, and we plan to incorporate additional security measures to further enhance the protection of sensitive user data. For instance, incorporating biometric authentication, two-factor authentication, or using blockchain technology could boost the security of the overall communication process.

Improving the user interface of the chat application is also a vital aspect that we plan to address in the future. By enhancing the design and user experience, users will have better interaction with the application, leading to higher engagement and overall user satisfaction. Some possible enhancements could include a redesign of the homepage, customizable themes, and handy automation tools that improve user experience.

In addition to the above-discussed improvements, we also plan to introduce a new feature to the chat application known as random chat. This feature would allow users to chat with strangers anonymously. It would provide users with an opportunity to meet and engage with people around the world and have interesting conversations with people who share similar interests.

The random chat feature would be designed with a secure infrastructure to ensure the safe interaction of users. Users can filter the selection of other users involved, depending on the demographics or interests to enhance the likelihood of a successful chat. The feature will provide a reliable and secure connection so that users have nothing to fear about data theft or online abuse or harassment.

By adding this feature, users can connect with fresh individuals, which can help break the monotony from the intent of a long-time chat with an acquaintances, family or friends. We believe that the addition of the random chat feature will provide a more holistic experience to our users, making the chat application more versatile, secure, and engaging.

In conclusion, the future development of the chat application will involve the expansion of features to enhance the user experience, and secure communication. Random chat feature, in particular, would be a valuable addition to the functionality of the application, allowing users to broaden their social circles and have interesting conversations with others . The future enhancements of the chat application will focus on improving user experience, security, and the incorporation of newer technologies. Our team believes that these enhancements will make the application more versatile, secure, and user-friendly to cater to users' needs in the future.