International Journal of Innovative Research in Engineering
Volume 3, Issue 1 (January-February 2022), PP: 34-43
www.theijire.com **ISSN No: 2582-8746**

Design of an Android Based Chat Application

Shashi Kumar N[1], Prof. Savita Sheelavant[2]
[1]*Department of Master of Computer Applications, RV College of Engineering®, India.*
[2] *Assistant Professor, Department of Master of Computer Applications, RV College of Engineering®, India.*

How to cite this paper: Shashi Kumar N[1], Prof. Savita Sheelavant[2], "Design of an Android Based Chat Application", IJIRE-V3I01-34-43.

Abstract: *With advancements in internet technologies, online communication has become an integral part of everyone's life. Chatting applications are one of the most used ways of online communications even surpassing social media. Despite the presence of many already existing messaging applications thriving in the current market, with the right scope and niche newer chatting applications have all the chances of appearing on people's smartphones. In this paper, a similar chatting application is designed for android having google firebase as the backend. The application has a secured login and registration, one-to-one and group chatting functionalities, and the profile of the users can be managed efficiently.*

Key Word: *Android Application; Firebase; Instant Messaging; Chatting Application; Online Communication*

I. INTRODUCTION

Nowadays instant messaging has become very important in every area of society. Be it for personal use or business purposes chatting applications have all the functionalities to fulfill the needs of various user groups. New chatting applications with different functionalities and features will definitely find their way to users' smartphones.

There is an ever-increasing market for chat applications and there are many chat applications already in the market which are doing very well. Despite the presence of so many chatting applications, people use many chat applications side by side. This shows that there will still be a good market for newer chat applications with the right features and niche. The developed application is an instant messaging application that sends and receives instant messages across the internet. The system has a good user interface and provides a smooth and nice user experience.

II. DESIGN AND METHODOLOGY

Existing System

The existing applications rely on phone number-based authentication and require everyone to possess a registered sim card. The user must first register the sim card with the application to use the application, which might be a little expensive. Also, it takes a lot of effort to get a registered sim card. But in comparison, it is economical and takes little or no effort to create and use an email account. It does not give the advantage of the OAuth standard and every time the user has to log into his account he has to type in his phone number and wants to get verified through OTP.

Proposed System

The proposed system is an android application that aims at providing the user with a simple and clean messaging functionality where the user can

- Successfully log in and out of his/her account using email-based authentication
- Send and receive one-to-one messages over the internet seamlessly
- Have a group chat with registered users
- Maintain and manage user profile with picture and status

The application also has an option for google based authentication wherein with the help of OAuth the user can log into his account with a single click of a button without having to compromise with the security and privacy of the user's account.

Methodology

The application is built for the android platform. It makes use of google firebase. Google Firebase provides many benefits such as authentication and storage. The firebase provides its own authentication mechanism through FirebaseAuth and stores the chat data in its central database with the help of FirebaseDatabase. Thus the application will have its own sign-in and sign-up activities, will have a one-to-one and group chatting facility and also the user profile can be customized to upload and maintain profile pictures and status.

Assumptions and Dependencies

- The application assumes that the user knows how to use a basic chat application
- The user needs to be registered to avail the benefits it offers
- The application is designed for the Android platform hence assumes the user has an android phone

Functional Requirements

- **Module Specification**

User Registration & Login Module

The user can register or log in through a valid email or google account.

- Input- Username, email address, and password or use google account
- Process- Checks the provided username with the database and registers the account if the account is not already registered. Logs the user in if already registered.
- Output- Message indicating the successful or unsuccessful registration or login.

User Chat Module

The user can send messages to other users of the application.

- Input- User input message
- Process- Reads the message from the buffer, stores it in firebase DB, and passes the message to the receiver
- Output- Sent message appears on the screen and a confirmation toast message

Profile Module

User profiles can be updated wherein we can update the profile picture and user status.

- Input- Image for profile picture, a string of text for user status
- Process- Updates the user's profile picture and user status details in the real-time database with the applied changes
- Output- Updates reflected on the user account along with a confirmation toast message

Group Chat Module

The user can send broadcast messages to the group members of the application.

- Input- User input message
- Process- Reads the message from the buffer, stores it in firebase DB, and passes the message to all the members of the group
- Output- Sent message appears on the right of the screen. Messages received from other users appear on the left. And a Confirmation toast message

Non Functional Requirements

- Scalability: The number of concurrently connected users to the application can be up to 100 users.
- Availability: The ChatsApp internal server will be available 24 hours a day with a Monthly Uptime Percentage of at least 99.95.
- Security: Firebase services provide authentication and encrypt data in transit using HTTPS and logically isolate customer data.

System Design

The system design diagrams aim in visualizing the underlying system's design in terms of the object-oriented concepts. The design diagram helps a reader to understand the interaction between the objects in the system to produce the final results.

Object Modeling

Class Diagram

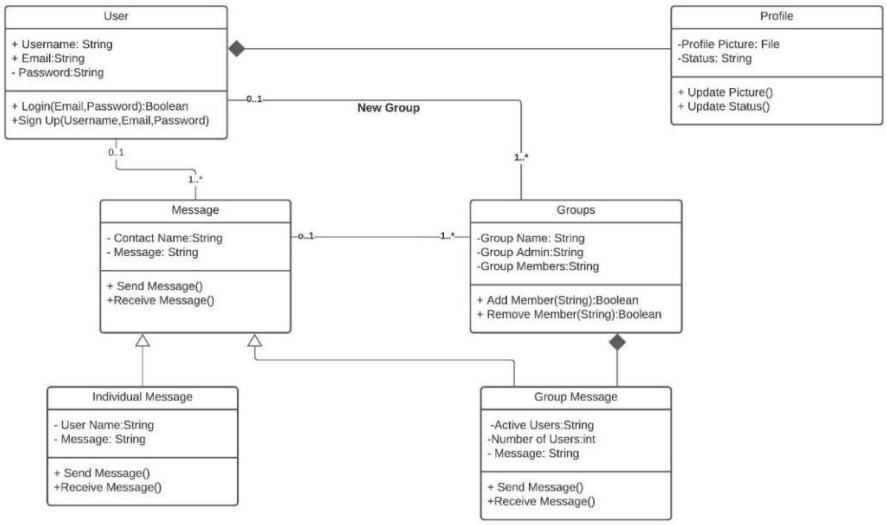


Figure 1. Class Diagram of ChatsApp

Figure 1 depicts the relationship between various entities in the system. The user can send and receive individual and group messages and can update his profile.

Dynamic Modeling

Use Case Diagram

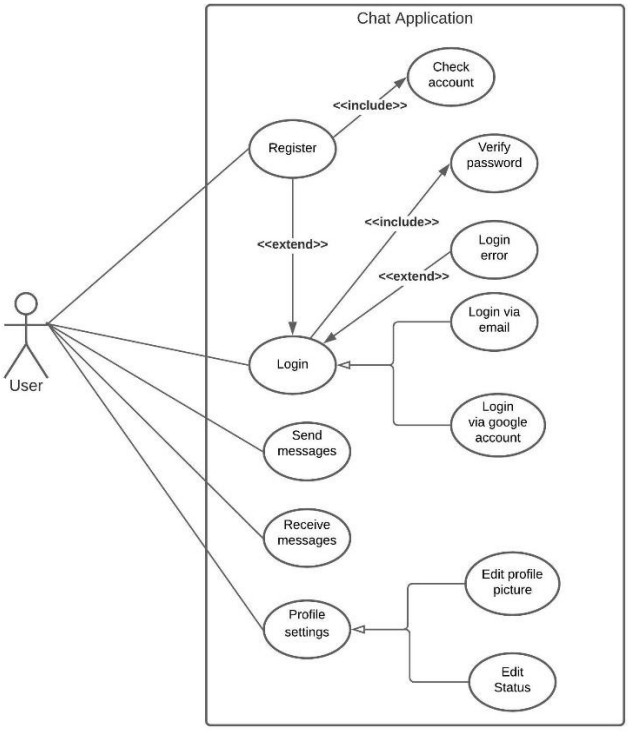


Figure 2. Use Case Diagram of ChatsApp

Figure 2 is a use case diagram depicting the various functionalities in the chat application system. The user's each use case including registering, logging in, sending and receiving messages and profile settings is shown in detail.

Activity Diagram

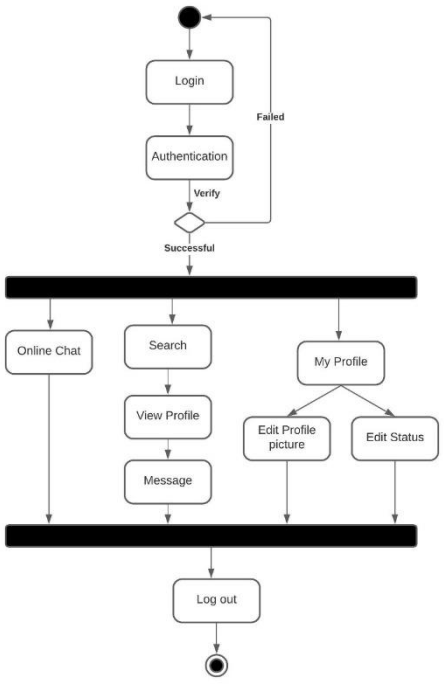


Figure 3. Activity Diagram of ChatsApp

Figure 3 is an activity diagram that depicts the flow of various user activities in the application.

Sequence Diagram

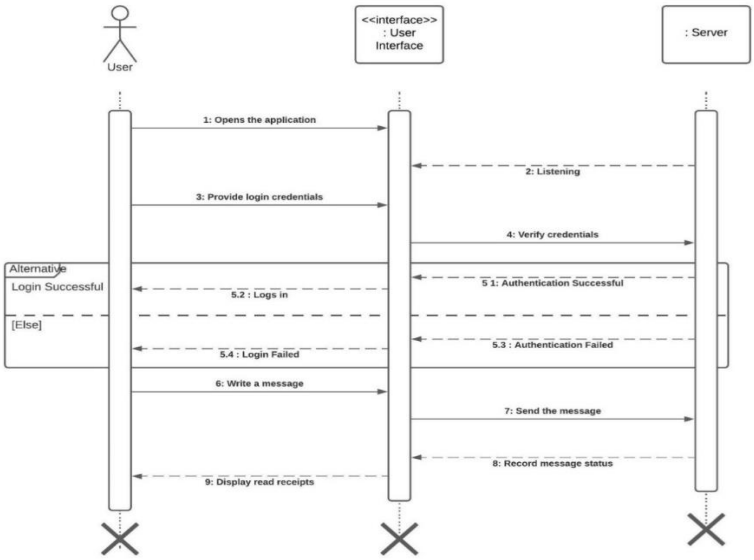


Figure 4. Sequence Diagram of ChatsApp

Figure 4 is a sequence diagram depicting how the sequence in which each of the functionalities takes place and the flow from one system to another system.

Functional Modeling
Data Flow Diagram
Zero Level DFD

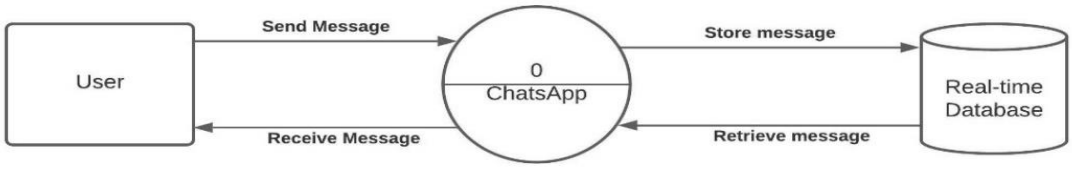
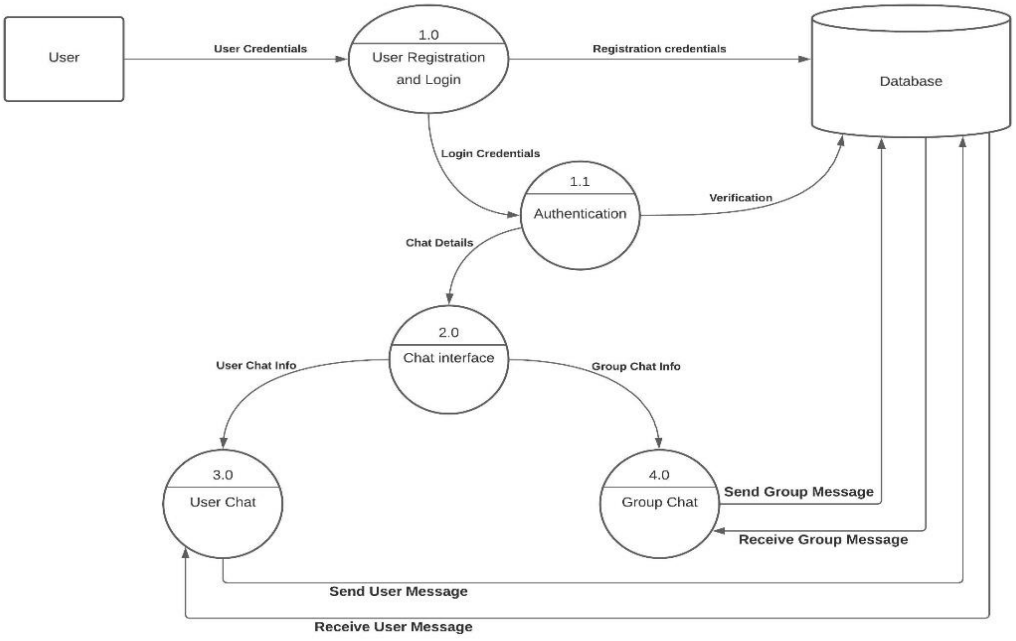


Figure 5. Zero Level DFD of ChatsApp

Figure 5 is the zero level DFD of the Chat application which gives the overall view of the application.
First Level DFD



.

Figure 6. First Level DFD of ChatsApp

Figure 6 gives a detailed overview of the application's data flow. The ChatsApp based on authentication logs in the user takes in messages and returns messages from other users and stores all the messages in the database.

III.IMPLEMENTATION

Logic Design

User Registration and Login Module

Provide user credentials like username and password and email
If the user is not registered
    If the credentials are in the correct format
        Register the user with the application
    Else: Ask the user the enter credentials in proper format
    endif
Else:
    If the registered user has entered valid user credentials
        Log the customer into his account
    Else: Prompt the user to enter valid credentials
    endif
end if

User Chat Module

Send and receive one-to-one messages
If the message is received from another user
Display it on the left of the screen
Else:
Display sent messages on the right
end if

Profile Module

Provide an image for the profile picture and update status
If either of the profile picture and status is left empty
    Prompt the user to fill both the fields
Else
    Update the profile picture and the status
end if

Group Chat Module

Send and receive messages from different users
If the message is received from other users
    Display it on the left of the screen
Else:
    Display sent messages on the right
end if

Application Implementation

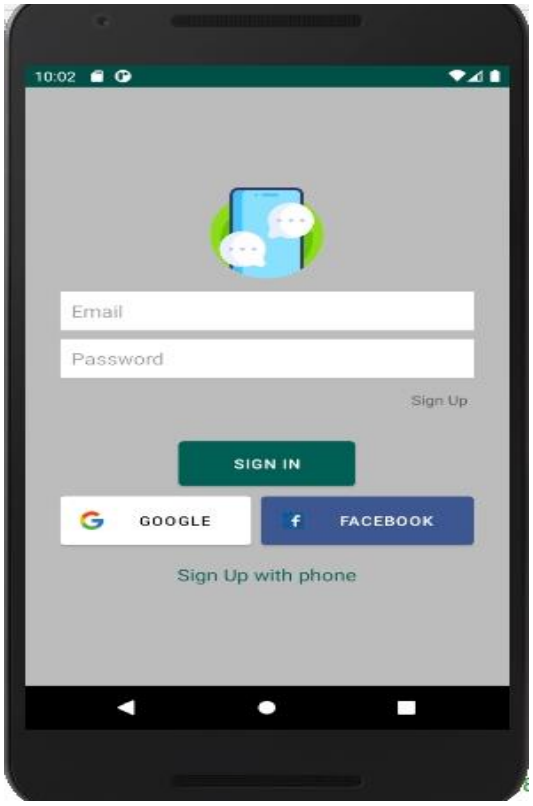


Figure 7. Sign In Activity(Login)

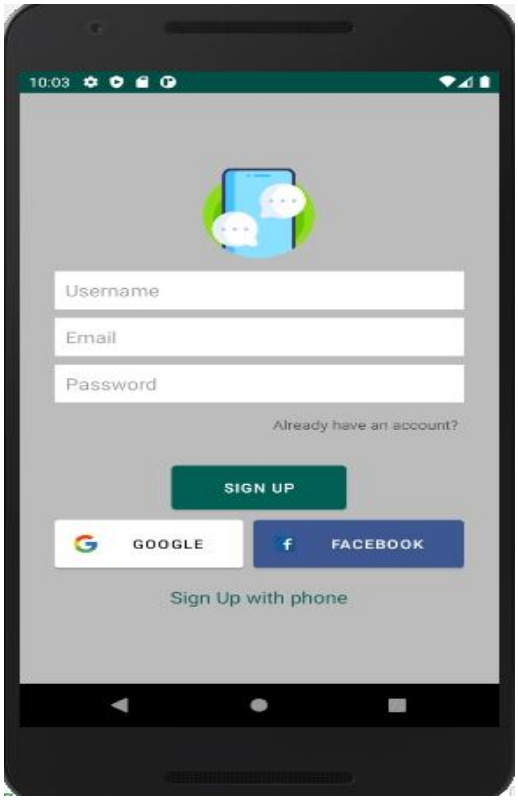


Figure 8. Sign Up Activity(Registration)

Figure 7 shows the sign-in activity that takes login credentials like email and password. It also has the option for google based login. Figure 8 is the sign-up activity that takes register credentials like username, email, and password and it too has the option for google based registration.

Figure 9. User Chat List

Figure 10. User Chat Activity

Figure 9 shows the interface where a list of all the chat activities is visible. Figure 10 shows the overview of the user chat

interface where the actual sent and received messages are seen.

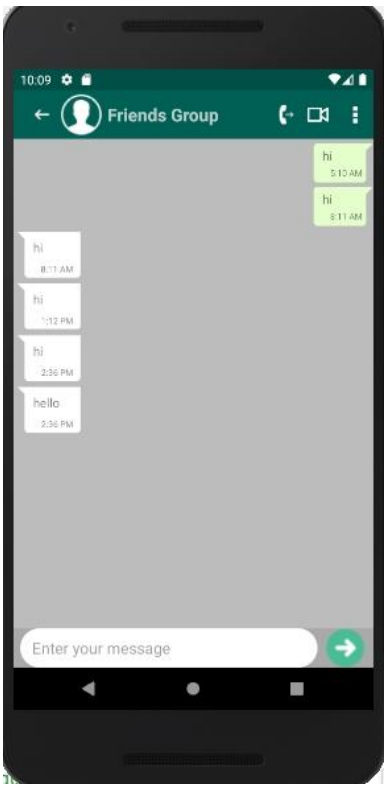


Figure 11. Group Chat Activity

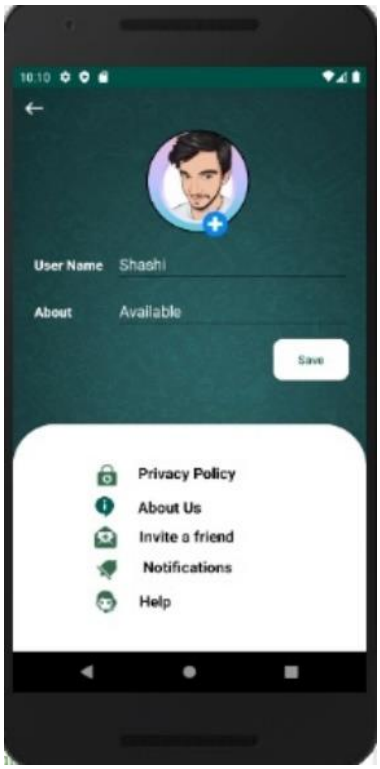


Figure 12 User Profile

Figure 6.5 shows the group chat interface where the group chats are visible and Figure 6.6 shows the interface for profile settings which contains information about user profile pictures and status.

IV.TESTING

Testing is important to confirm that the developed product is reliable and meets all the requirements. Unit Testing is done for the smallest unit i.e., code snippets during development. Unless the unit testing for small units is not successful units can't interact with one another.

Unit Test Cases for ChatsApp

Registration and Login Module

Table no 1 Unit Test Case table for Registration and Login Module

Test case ID	Feature Tested	Sample Input	Expected Output	Actual Output	Remarks (Pass/Fail)
01	Login with an inexistent username	Email- Shashiz@gmail.com (Inexistent user) Password- *****	There is no user account corresponding to the identifier	There is no user account corresponding to the identifier	pass
				Any other output	fail
02	Login with the wrong password	Email- Shashi@gmail.com (Existent user) Password- ***** (Wrong password)	The password is invalid	The password is invalid	pass
				Any other output	fail
03	Try to register with the wrong email format	Email- Shashi Password- *****	The email is badly formatted	The email is badly formatted	pass
				Any other output	fail

Table no 1 shows the unit test cases for the registration and login module with different user inputs.

User Chat Module

Table no 2 Unit Test Case table for User Chat Module

Test case ID	Feature Tested	Sample Input	Expected Output	Actual Output	Remarks (Pass/Fail)
01	Sending and receiving chats	senderMessage = hey receiverMessage = hello	The sent message should appear on the right and the received message on the left	The sent message should appear on the right and the received message on the left	pass
				Any other output	fail
02	Writing a very long text message	Message= Hello my friend. Here's a good message for you. "Nothing can be done without hope and confidence".	The message body must wrap itself to present a decent readable message	The message body must wrap itself to present a decent readable message	pass
				If the message does not wrap around and overlap	fail

Table no 2 shows the unit test cases for the user chat module and gives an idea about how the sent and received messages must be laid out on the chat interface.

Group Chat Module

Table no 3 Unit Test Case table for Group Chat Module

Test case ID	Feature Tested	Sample Input	Expected Output	Actual Output	Remarks (Pass/Fail)
01	Sending and receiving chats	senderMessage = hey groupMemberMmessage = hello	The sent message should appear on the right and the group member message on the left	The sent message should appear on the right and the group member message on the left	pass
				Any other output	fail
02	Check if every group member can send a group message	Group Name = Friends Message= Hi.	All the members must be able to send and receive group messages	All the members must be able to send and receive group messages	pass
				Any other output	fail

Table no 3 shows unit test cases for the group chat module and depicts how group messages must be laid out.

Profile Module

Table no 4 Unit Test Case table for Profile Module

Test case ID	Feature Tested	Sample Input	Expected Output	Actual Output	Remarks (Pass/Fail)
01	Updating profile with empty fields	Profile Picture = Image Status = Empty	Please fill all the fields	Please fill all the fields	pass
				Any other output	fail
02	Updating both profile picture and status correctly	Profile Picture = Image Status = Hope!	Profile Updated	Profile Updated	pass
				Any other output	fail

Table no 4 shows unit test cases for profile module and tests for values of various profile inputs.

Testing and Validations

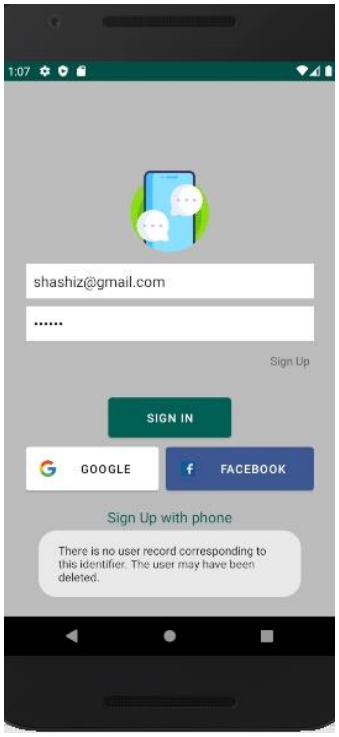


Figure 13. Login with inexistent username

Figure 14. Login with an invalid password

Figure 13. shows the case where the user enters an inexistent username which shows a toast message indicating the error. Figure 14. shows the case where the user enters an invalid password which shows a toast message indicating the error.

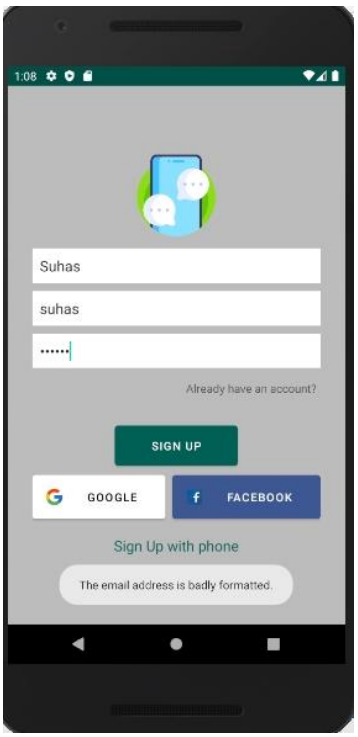


Figure 15. Register with invalid email format

Figure 16. Sending and receiving chats

Figure 15. shows what happens when the user tries to register with an invalid email format. Figure 16. shows how the sent and received messages are laid out in a user chat activity.

Figure 17. Writing a Long Message

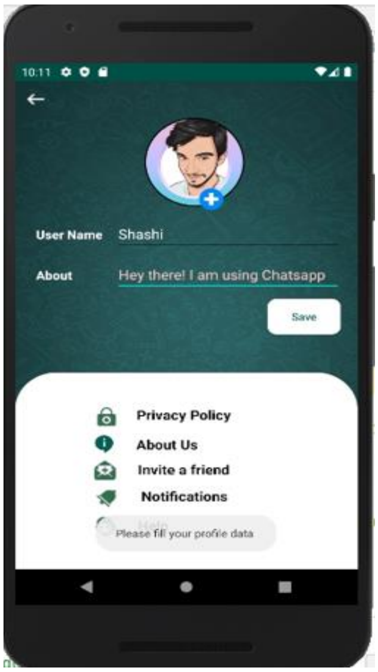


Figure 18. Empty Profile Fields

Figure 17 shows that when the user tries to type in a long message, the message should be wrapped around to make it more readable. Figure 18 shows that an error toast message will be shown if the user does not fill in all the profile fields i.e, profile picture and the status.

V. CONCLUSION

Internet is growing at a fast rate and online communication has become an integral part of everyone's life. With the ever-growing market for chat applications, if the application hits the right niche by fulfilling users' needs, then the application has all the chances to appear on people's smartphones thriving even in the current competitive market.

The developed application has several important features: Sign Up, Sign In, Edit Profile, Chat, and Group Chat. The testing was carried out in the form of simulations using emulators and smartphones, and each of the application's main features runs successfully as expected. Each of the activities designed in this project has independent functionality. Other activities can be added as and when needed. Every attempt has been made to ensure that the application is fully functional and works effectively and efficiently. The application has been tested with all possible data to cover all possible options and checked for all outputs. Since the application is flexible and modular, further modification can be easily incorporated. The application is highly extensible as well. New updates and functionalities can be easily integrated without having to reconfigure the entire application thus making further development easier. The application also acts as a starting point in building a highly commercial and secure messaging application with other high-end functionalities which takes the user experience to another level.

References

[1]. *Ashita, Vasudha Bahl, Dr.Amita Goel, Nidhi Sengar, "Android Based Instant Messaging Tool Using Firebase as Backend", International Journal for Modern Trends in Science and Technology, 6(12), pp. 198-201, 2020.*

[2]. *Vipul Chaudhari, Rohit Gawade, Vishal Naik, "Android based Instant Messaging Application using Firebase for Insurance", International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 12, Dec 2020.*

[3]. *Vincent Angelo, Benny Hardjono Frans Panduwinata, Andree E. Widjaja, Aditya R. Mitra, "Android Based Chatting Application Using Firebase for Facilitating The Communication Among Indonesian Police Force Members", International Journal of Advanced Trends in Computer Science and Engineering Volume 8, No.1.5, 2019.*

[4]. *Sai Spandhana Reddy Emmadi, Sirisha Potluri , "Android Based Instant Messaging Application Using Firebase", International Journal of Recent Technology and Engineering (IJRTE)ISSN: 2277-3878, Volume-7 Issue-5S2, January 2019.*

[5]. *Robi Sanjaya, Abba Suganda Girsang , "Implementation Application Internal Chat Messenger Using Android System", International Conference on Applied Computer and Communication Technologies (ComCom), Jakarta, Indonesia, 2017.*

[6]. *"Step-by-Step Guide on Custom Mobile Chat Application Development", yellow.systems, 28 May 2021*

[7]. *Mike. R, "How to Build a Chat or Messaging App: The Ultimate Guide", Internet: https://getstream.io/blog/build-chat-messaging-app, Jan 2020*

[8]. *Rosário Pereira Fernandes, "Using Firebase In-App-Messaging on an Android app, Internet:https://proandroiddev.com/using-firebase-in-app-messaging-on-an-android-app-f2802757f00b, Jul 11, 2020*

Alex Peston, Solutions Engineer, "Android chat tutorial: How to build a messaging UI", Internet:https://sendbird.com/developer/tutorials/android-chat-tutorial-building-a-messa