Innovative Messaging: Building a User Friendly and Secure Communication platform

1st Shiv Arora
Department of CSE
CGC College of Engineering
Punjab, India 2121466.cse.coe@cgc.edu.in

5th Vandana
Department of CSE
CGC College of Engineering
Punjab, India Vandana.5523@cgc.edu.in

2nd Satyam
Department of CSE
CGC College of Engineering
Punjab, India 2121459.cse.coe@cgc.edu.in

3rd Shivam Kumar
Department of CSE
CGC College of Engineering
Punjab, India 2121468.cse.coe@cgc.edu.in

4th Shaffi
Department of CSE
CGC College of Engineering
Punjab, India 2121462.cse.coe@cgc.edu.in

***Abstract*—The rise of instant messaging apps has revolutionized communication, providing instant, dynamic exchanges through various mediums such as text, images, voice, and video. This paper delves into the development and execution of a user-friendly, secure, and adaptable instant messaging application, striving to deliver an unobstructed chatting experience across all devices. Powered by Java, Firebase, and Android technologies, the app guarantees message privacy with cutting-edge end-to-end encryption, enables multimedia sharing, and includes practical features such as group chats and file transfer. Furthermore, the paper explores the app's architecture, user interface design, and the obstacles encountered in ensuring its dependability and scalability. Experience seamless communication with our versatile messaging application, which offers both private and public messaging options. Our robust system surpasses traditional methods in reliability and security. Our methodology prioritizes efficiency and continuous improvement through the integration of agile methodologies and user feedback. We prioritize security with features like end-to-end encryption and rigorous user authentication. The architectural and design section underscores scalability, flexibility, and responsiveness through cloud services, real-time messaging, and an intuitive interface. The outcomes clearly showcase how well the app delivers a smooth and seamless chat experience, without compromising message security and dependability.**

***Index Terms*—Java, Firebase, Android, Cloud services.**

1. INTRODUCTION

In the modern digital age, traditional text messaging services like SMS (Short Message Service) have been overshadowed by the rise of innovative portable messaging applications. These applications have revolutionized the way people communicate, offering a more convenient and feature-rich experience. According to recent statistics, an astonishing 19 billion messages are sent daily through these applications, surpassing the 17 billion SMS messages transmitted globally [1]. However, amidst the abundance of messaging apps available, only a few provide robust end-to-end encryption capabilities, ensuring the privacy and security of user communications [2]. As the demand for secure messaging solutions continues to grow, there is a need to explore and develop applications that prioritize user privacy while offering a seamless and user- friendly experience. The evolution of messaging technology can be traced back to 1984, when the concept of SMS was introduced by Friedhelm Hillebrand and Bernard Ghillebaert during the Franco-German GSM collaboration. Despite its groundbreaking nature, SMS had a significant limitation – a maximum message size of 128 bytes [3] [4]. With the advent of smartphones over the past decade, a myriad of innovative messaging applications have emerged, revolutionizing the way we communicate. One such application leverages the power of Firebase, a NoSQL database that utilizes sockets, allowing users to store and retrieve data seamlessly [5]. To develop an Android application integrated with Firebase, certain requirements must be met, including an Android version higher than 2.3, Android Studio 1.5 or a newer version, and a compatible Android Studio project. Firebase offers a wide range of services that simplify the development process. Firebase Authentication provides an easy- to-use API [6] [7] for implementing secure sign-in mechanisms, eliminating the complexities of building and maintaining authentication systems from scratch. Additionally, Firebase's real-time database facilitates efficient data backup and retrieval. Furthermore, Firebase Cloud provides a robust platform for storing various types of data, including text, images, and videos, eliminating the need for developers to build complex and costly storage systems [6]. Firebase also offers crash reporting services, enabling developers to identify and address application crashes effectively. The research paper focuses on developing a chat application, a many-to-many communication system where users can exchange messages among themselves [8]. This application aims to leverage the power of Firebase and incorporate robust security measures to ensure end-to-end encryption, providing users with a secure and privacy-focused messaging experience.

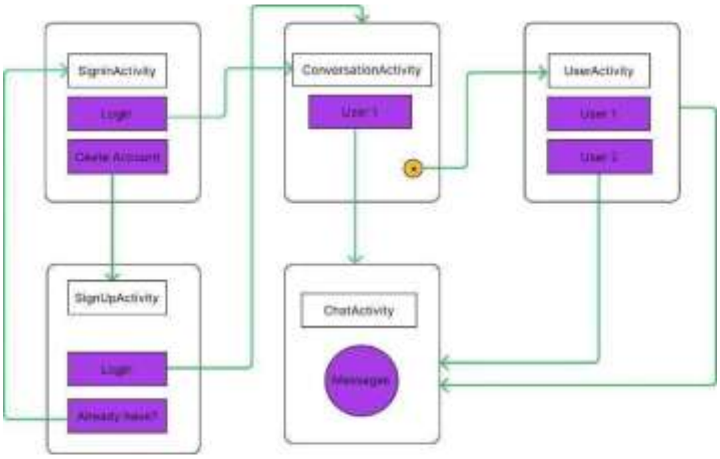


Fig.1 Use Case Diagram

LITERATURE SURVEY

2. Literature Survey

Author Name	Technology	Findings	Strengths/Limitations
Church & de Oliveira (2013)	Mobile Instant Messaging Apps	Network effects and ease of use were key drivers for adoption and use.	Strength: Insights into user motivations and behaviors. Limitation: Focused on specific apps that may now be outdated.
Piwek & Joinson (2016)	WhatsApp	Positive associations between WhatsApp use and relationship satisfaction and life satisfaction.	Strength: Examined specific well-being outcomes. Limitation: Cross-sectional design limits causal inferences.
Huang et al. (2017)	Mobile Instant Messaging Apps	Developed a framework for understanding continuance intention to use, considering factors like perceived enjoyment, critical mass, and social influence.	Strength: Comprehensive theoretical framework. Limitation: Relied on self-reported data.
Fiadino et al. (2019)	WhatsApp	Explored both positive and negative impacts of WhatsApp use on student-student and student-teacher communication in higher education.	Strength: Explored educational applications. Limitation: Focused on a specific context, limiting generalizability.
Aritz et al. (2019)	WhatsApp, WeChat, Telegram	Evaluated security and privacy implications, identifying potential vulnerabilities and risks.	Strength: Insights into cybersecurity concerns. Limitation: Rapidly evolving technology may make findings quickly outdated.
Montag et al. (2019)	WhatsApp	Associations between WhatsApp use, personality traits.	Strength: Explored individual differences. Limitation: Reliance on self-reported measures.
Abeele et al. (2020)	WhatsApp	Different motivations for using WhatsApp (e.g., relationship maintenance, entertainment) influenced well-being outcomes.	Strength: Considered user motivations as a moderating factor. Limitation: Focused on a specific app, limiting generalizability.
Yin et al. (2020)	Chatbots on Chatting Platforms	Proposed a chatbot system for providing mental health support, evaluating potential efficacy.	Strength: Explored novel applications. Limitation: Efficacy evaluation based on simulated scenarios.
Huang & Gui (2021)	Enterprise Chatting Apps	Developed a framework for understanding adoption and continuous use in organizational settings.	Strength: Explored organizational applications. Limitation: Focused on a specific context, limiting generalizability.
Aldiabat et al. (2022)	Chatting Apps in Healthcare	Examined use for providing healthcare services and support during COVID-19, identifying benefits and challenges.	Strength: Explored practical applications in healthcare. Limitation: Focused on a specific context, limiting generalizability.

3. Proposed System

First we will understand objectives after that architecture will be discussed:

- Discuss strategies to scale the app without sacrificing functionality in order to support more users. This covers real-time data handling, effective database design, and the usage of scalable cloud infrastructure.
- Evaluate how to add advanced features like interactive bots, audio and video chats, stickers affects user interaction and app usability. This objective aims to assess the technical challenges and benefits associated with integrating these features.
- To ensure that the app works well and is open on different gadgets and stages, we utilize a responsive plan.
- The app should ensure message security by implementing end-to-end encryption, which ensures that messages are encrypted on the sender's device and decrypted on the recipient's device, making them unreadable to anyone else.

The proposed architecture for the messaging application is designed to address the core objectives of providing a secure, efficient, and user-friendly platform for real-time communication. The architecture has Firebase for real-time database and authentication services, and Android, xml will be for client-side application and language used in Android is Java. The Section outlines the key components and how they interact within the system to offer a comprehensive messaging solution.

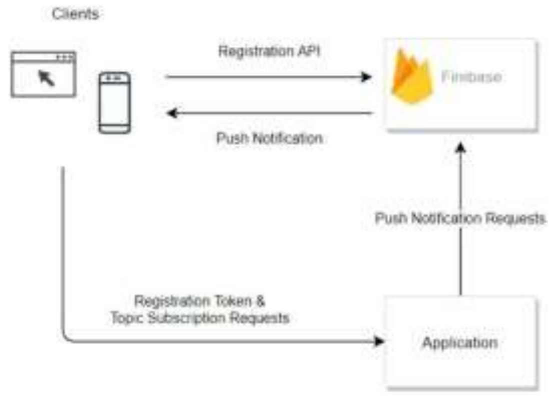


Fig. 2. Work flow on firebase

A. Client Side Application (Android)

As an OS, the task of Android is to be a translator between the user and the device. In short, Android would act and work like Windows OS, but it works only for mobile phones[9][16].

In android, API can be used to retrieve the data from the database. Data is received in the format of JSON and XML mainly. The API is implemented by a function call consisting of verbs and nouns[10]

Local Storage: SQLite is used for local storage of messages, contacts, and multimedia files, ensuring that the app remains functional even when offline[11].

B. Backend(Firebase Cloud Messaging)

Firebase Real-time Database: Used for storing and syncing messages and user data in real-time[12].

Firebase Cloud Messaging (FCM) is a service whose work is to informing between mobile applications and server applications. Google Play Services powers this cross-platform system for Android, iOS, and web. It provides a free service that sends lightweight messages from the server to devices when new data is available [13]. This real-time communication ensures seamless data delivery across platforms. The innovative approach keeps users updated efficiently.

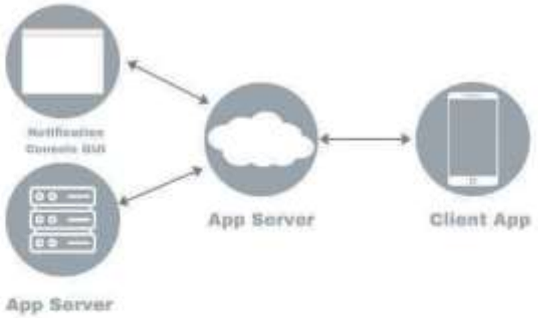


Fig. 3. Firebase Cloud messaging

C. Sign Up and Login

For registration, the customer must enter personal data. These credentials can be the client's phone number and passphrase, or an OAuth authorization from a connected authorization provider. Currently, these credentials are being passed to the Firebase Authentication SDK. Backend administrators validate the relevant credentials and return response to the client[14][15].

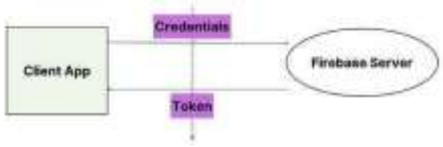


Fig. 4 Registration

After Signup, Login process is done as shown in Fig.5. All the data is stored in firebase database. One user can have multiple id's by signing up with different email id.

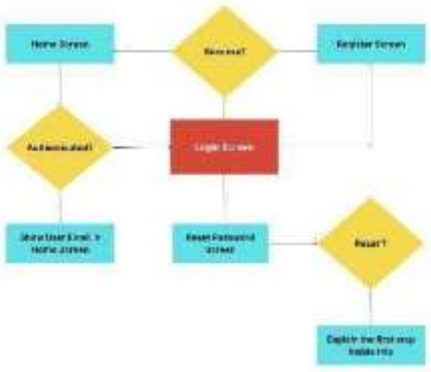


Fig. 5 Login Process

4. Implementation

The implementation of the chat application using Firebase and Android Studio focuses on creating a streamlined user experience while ensuring robust backend functionality for real- time interactions. This section outlines the implementation steps and the results observed during the development and testing phases.

A. Registration

Using Firebase Authentication, the registration process is simplified to enhance user convenience:

a) *Enter your E-mail id:* The user interface in Android Studio prompts the user to enter their email ID. This utilizes the 'EditText' component for input.

b) *Check your password:* The user is then asked to create a password, inputted through a password-secured 'EditText' field.

Firebase Setup:

a) *User Data Structure:* A Firebase Realtime Database schema is designed to store user information securely. The schema includes fields such as user ID (generated by Firebase Auth), email, and user's status.

b) Authentication: Firebase Authentication handles the registration process, creating a new user with email and password. This process is managed through the FirebaseAuth class in Android.

B. Login to Your Account

If valid user, move to next step if not go to registration process. Firebase Authentication's signInWithEmail method is used.

C. Setup your profile:

User can put their picture as profile and write their bio, so as to be recognizable to others.

D. Search friends to add them

Utilizing the Firebase Realtime Database, users can search for friends by email or username. This feature queries the database for matching entries and displays them in a user- friendly list view.

E. Logout when done

The logout functionality is straightforward, utilizing Firebase Auth's signOut method.

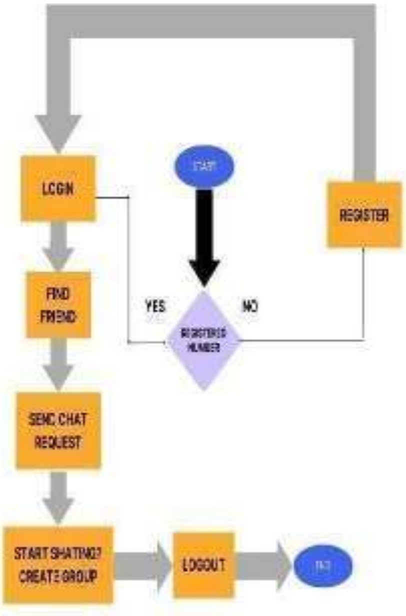
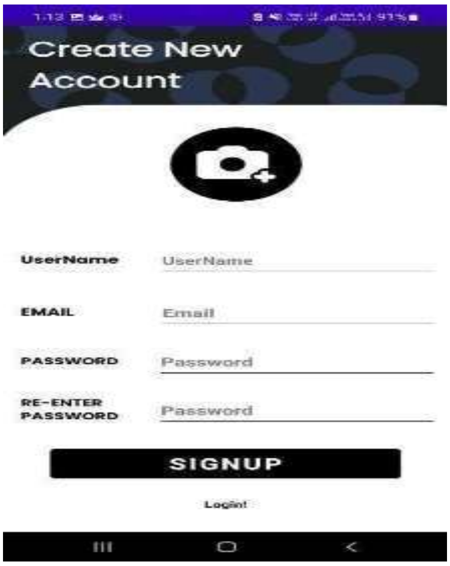


Fig. 6 Work flow under chat application

5. Results and Discussion

The latest framework is a real-time application that allows customers to easily communicate with each other. The application has a splash screen when clicked then a registration page for new users, if already registered then login page through which user login with credentials. Users send and receive instant messages.

The authorization feature allows a user to access their account anywhere, anytime on any mobile device. User should ask the other user by a request for chat feature. If request is accepted they can send the messages, otherwise can't. The user must login to their app with their email address and password.



Create New Account

UserName: UserName

EMAIL: Email

PASSWORD: Password

RE-ENTER PASSWORD: Password

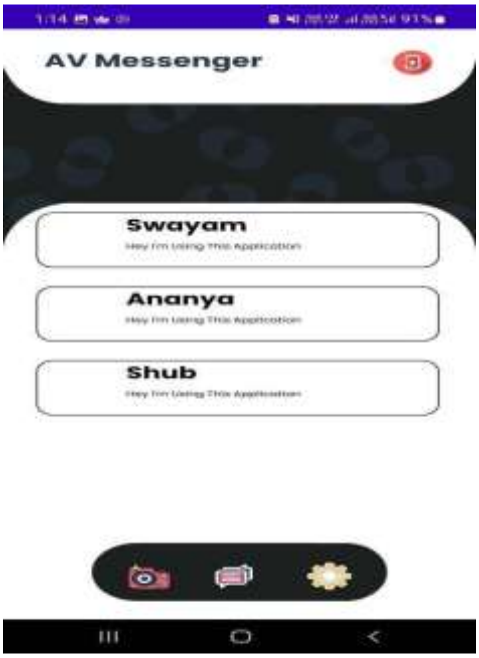
SIGNUP

Login!

Fig. 7 User Registering Page

A shown in Fig.7, [18] Firebase sends the user to the account creation page if the username is not registered in the Firebase database. This page prompts users to enter their username and password. Users can then click SignUp and their account is created.

Once account is created they can use the login page to enter their password and lands on main chat page to search users to chat. The sign in page also offers an option for users to check list of other users shown in Figure 7 and in firebase database we can see all the registered users.



AV Messenger

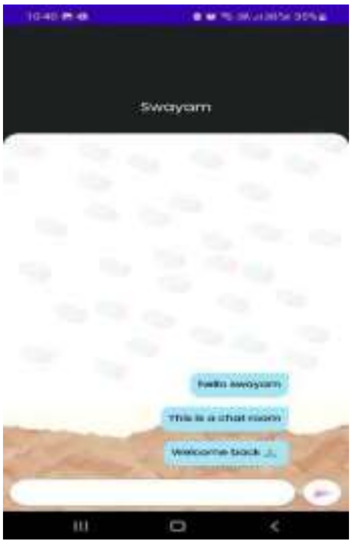
Swayam
Hey I'm Using This Application

Ananya
Hey I'm Using This Application

Shub
Hey I'm Using This Application

Fig. 8 List of Users to chat

The Fig.8 shows list of users one can chat with and also settings icon to update status and to put profile picture. The page also contain camera to post a story which can lasts 24 hours or can take pictures to send multimedia etc.



hello swayam

This is a chat room

Welcome back

Fig.9 Chat Room

The chat room has attractive User Interface with changeable background and theme according to user need as shown in Fig. 9

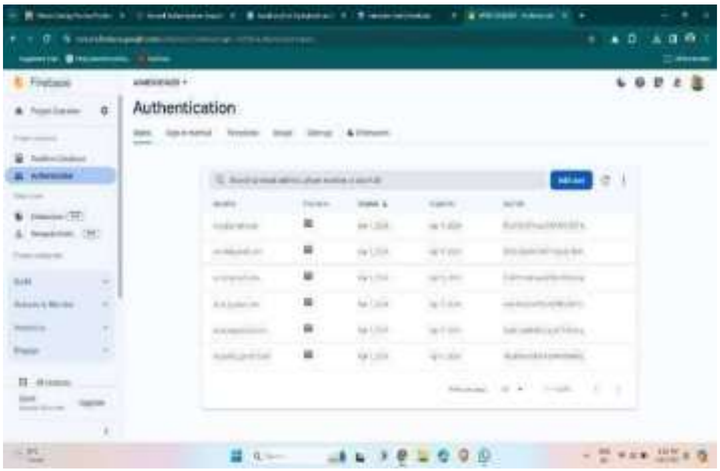


Fig. 9 Authentication Database

The Admin will have database of all users registered in the application as shown in Fig.9. Admin can check analytics of active users and regular usage on the application. Engagement Analytics and traffic management will be checked on regular basis after launching of app.

Many chat apps are emerging today and people are using them very effectively. To develop the application, the customer provides feedback on what is needed and what is in the current applications. The survey based on the features of existing applications, such as WhatsApp, Telegram and WeChat[14]. The following are some of the features that were liked by people of different age groups:

- Status saving
- GIF's
- Reaction to messages
- Disappearing of messages
- One time watchable
- Edit messages after sending
- Delete for everyone

6. CONCLUSION

In conclusion, the development of our messaging app is an important step in providing a secure, efficient and user-friendly text-based communication platform[18]. By focusing on the security and privacy of user data, applying modern encryption technologies such as the XSalsa20[14] algorithm and using lightweight methods to achieve optimal performance, we have built a strong foundation to provide high-quality messaging.

Android, the predominant mobile operating system, demands stringent security measures to safeguard users' sensitive information [19].

Chat applications have become ubiquitous platforms for personal communication, often facilitating the exchange of confidential data [20]. Alarmingly, recent security breaches in major chat services have exposed vulnerabilities, raising doubts about their ability to robustly implement privacy and security features [2] [17]. Ensuring the confidentiality and integrity of user communications through secure messaging solutions has emerged as a critical imperative. Developers must prioritize robust encryption protocols and data protection measures to fortify these applications. Comprehensive security audits and timely updates are essential to mitigate potential vulnerabilities that could compromise user trust and sensitive information. Striking a delicate balance between functionality and security is paramount in the ever-evolving landscape of mobile chat applications.

However, every software application needs improvement and there is always scope improvement and expansion. While our current focus lies on facilitating text-based exchanges between friends, future iterations of the application have the potential to incorporate a variety of additional features to enhance user engagement and functionality. These may include options for message deletion (both for everyone and for individual users), personalized message tones, and status updates to allow users to share their current activities or moods.

In future, we will add more features to our chat application to make it more efficient, for example,

1. Translation of languages
2. Search messages through date

These were some new features we are planning to add. However our application has deprived of so many existing features in available and launched, we will try to add those features as well as new features.

REFERENCES

.

[1] Zhang, L., Xu, C., Pathak, P. H., & Mohapatra, P. (2015). Characterizing Instant Messaging Apps on Smartphones. In Passive and Active Measurement Lecture Notes in Computer Science (pp. 83-95).

[2] Ali, A. H., & Sagheer, A. M. (2017). Design of Secure Chatting Application with End to End Encryption for Android Platform. In the proceedings of Iraqi Journal for Computers and Informatics (IJCI).

[3] Anon. (2015). Development of a Health Care Assistant App for the Seniors. In the proceedings of International Journal of Applied Science and Engineering.

[4] Liu, J., & Yu, J. (2011, December 15). Research on Development of Android Applications. In the proceedings of 4th International Conference on Intelligent Networks and Intelligent Systems.

[5] Dongre, N. M. (2017, March-April). Journal of Computer Engineering. IOSR- JCE, 19(2), Ver. I, 65-77.

[6] Mittal, J., Garg, A., & Sharma, S. (2020). Online Chatting Application. In the proceedings of International Journal of Research in Engineering, IT and Social Sciences.

[7] Shaheen, J. A. et al. (2017). Android OS with its Architecture and Android Application with Dalvik Virtual Machine Review. International Journal of Multimedia and Ubiquitous Engineering, 12(7), 19-30.

[8] Akram, R. N., & Ko, R. K. L. (2014). End-to-End Secure and Privacy Preserving Mobile Chat Application. In Information Security Theory and Practice. Securing the Internet of Things Lecture Notes in Computer Science (pp. 124-139).

[9] Schmidt, C. (2016). What is Android? Here is a Complete Guide for Beginners. androidpit.com. Retrieved from https://www.androidpit.com/what-is- android

[10] Nolle, T., & Li, T. (2017). What is Application Program Interface (API)? techtarget.com. Retrievedfrom https://searchmicroservices.techtarget.com/definition/application- program-interface-API

[11] A. K., S. (n.d.). Mastering Firebase for Android Development: Build real- time, scalable, and cloud-enabled Android apps with Firebase. s.l.: s.n.

[12] Khan, S. N., & Firdous, I. U. (2017, April). Review on Android App Security. International Journal of Advanced Research in Computer Science and Software Engineering, 7(4).

[13] Firebase. (n.d.). Firebase Cloud Messaging. Retrieved from https://firebase.google.com/docs/cloud-messaging/

[14] Sharma, D., Sharma, M., Upadhyay, H., & Akilarasu, G. (2021). Developing Chat Application using Firebase. In Proceedings of the International Research Journal of Engineering and Technology (IRJET).

[15] Auti, P., Mahale, S., Zanjad, V., & Dangat, M. (n.d.). An Android Based Global Chat Application.

[16] Bernstein, D. J. (2011). Extending the Salsa20 nonce (No. Mc 152, pp. 1– 14).

[17] Reddy, S. S. (2019, January). Android based instant messaging application using firebase. Article.

[18] Kathuria, A. et al. (2015, May). Challenges in Android Application Development: A Case Study. Vol.4 Issue.5, 294-299.

[19] Karthick, S. (2017, July 13). Android security issues and solutions. In 2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA).

[20] Angelo, V., Hardjono, B., & Panduwinata, F. (2019). Android Based Chatting Application Using Firebase for Facilitating The Communication Among Indonesian Police Force Members. In the proceedings of International Journal of Advanced Trends in Computer Science and Engineering.