iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** V **Month of publication:** May 2022

DOI: https://doi.org/10.22214/ijraset.2022.43009

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

Instant Messaging Application Based on Android

Saiyyam Singh[1], Vedant Bhatnagar[2], Sarthak Gautam[3], Deepanshu Singh Satwaliya[4]
[1, 2, 3, 4]*Department of Computer Science and Engineering, KIET Group of Institutions, Ghaziabad, U.P, India*

Abstract: ***Through communication, an individual can express his ideas, feelings, and various aspects to others. Communication via the internet is becoming increasingly important these days. Through online communication, users can communicate with others remotely quickly, and conveniently irrespective of their physical location. With this in mind, communication applications should be able to instantly transfer files and messages with no delay or with minimal delay, depending on the transmission medium. For such a system to work, it needs a database that is updated in real-time to keep track of all the data sent. Google Firebase is a service that provides a real-time database server and various other functions. Firebase makes it relatively easy to develop communication-based applications. The main purpose of this paper is to initiate real-time communication between users through a software application that works on android devices connected to the internet such that the communication gap between the users gets reduced. Google Firebase is used to handle the communication process of the backend and text translation and image optical character recognition are among the features included in this application.***
Keywords: Android, Firebase, Instant messaging, ML translate, Text recognition

I. INTRODUCTION

The world is moving online, and communication is crucial in this environment. To stay in touch with others, almost everyone has started utilizing some form of the messaging tool. Due to the present COVID 19 pandemic, the use of messaging-based applications has expanded even more, from school and college work to office work. In recent years, the popularity of these applications has risen. Not only are these applications utilized for one-to-one communication, but also for group conversation, and we also use them to share media and data. With the rise in population, so does the use of social media. Chat applications have evolved and made significant changes to social media in recent years as a result of their specific traits that attract viewers. It delivers real-time messages and provides a variety of services such as text, photos, and data. Messaging applications have undoubtedly become a need in recent years since they have greatly enhanced productivity and are quite convenient. This paper focuses on identifying issues in the present application, doing a review of several publications and their approaches to solving the problems, and developing a solution with a safe architecture that does not monitor personal information and requires very little personal information. Additional functions such as one-to-one messaging, community group messaging, file sharing, text translation, optical character recognition for extracting texts from images, and so on are included. The suggested design will employ Google Firebase as a backend server and a TLS-encrypted communication.

II. LITERATURE REVIEW

With the advent of mobile phones, developers have sought to implement text-based messaging [1] services that enable instant communication capabilities. For this reason, in 1984, the idea of SMS was created in the Franco-German GSM collaboration by Friedhelm Hillebrand and Bernard Ghillebaert. However, the main drawback for the new idea was the size limit that can be used to write a message, primarily 128 bytes. The first SMS was sent in 1992 after making various improvements based on the original idea. The first commercial SMS service was launched the following year by Sweden's Telia and Aldiscon. SMS was the predominant form of communication in the 2000s, and although SMS was often used for rapid communication and delivery of emergency messages, its high cost was a disadvantage. With the advent of smartphones in the late 2000s, different messaging applications based on different operating systems became available and became popular with the general public.

The most notable ones included WhatsApp, WeChat, Telegram, Viber, Snapchat, and a few others. Android is an operating system that was created by google [2] for smartphones and is frequently utilized by users all over the world. Android application [3] is generally built with the help of android studio as IDE by using different programming languages like kotlin, java, dart, and others. Firebase handles most of the server-side work when developing applications. It is a NoSQL database [4], it uses sockets that allow users to store, retrieve and sync the data in real-time. From the standpoint of a developer, many elements make firebase an essential tool for development. In this way, minimizing work delays helps maintain harmony between developers and clients. When developing communication or chat applications, the main components or services provided by firebase are:

Realtime Database: Realtime Database is a database hosted in the cloud. The data is saved in JSON format and is continuously synchronized with each associated client. When developing cross-platform applications using the iOS, Android [5], and JavaScript SDKs, most user requests are based on a real-time database instance, which is updated with new data. This feature allows developers to skip database development steps and have Firebase handle most of the application's backend. It provides a flexible expression-based rule language to define how data should be arranged and when information can be read or created.

Firebase Authentication: Firebase Authentication is useful for both developers and users. Developing and maintaining the login function can be a bit difficult and time-consuming. Firebase provides a simple API for login. Users can log in to their firebase app by manually integrating one or a few login techniques by using SDK.

Storage: Firebase Storage is a service for app developers who need to store and serve user-generated material, like images or other files. Regardless of network condition, it provides safe document transfers and downloads for Firebase apps.

Cloud Messaging [6]: It's a cross-platform solution that lets developers send messages reliably and for free. Notification messages can be sent by developers to encourage user re-engagement and upkeep.

Ekata M. Lambture et al. (2016) suggested [7] using a Wi-Fi connection to send and receive text messages and files. They feature a system that allows users to chat with one other because Wi-Fi technology is low-cost and low-power. Wi-Fi hotspots that enable users to connect can be used to connect the devices. However, this method may have limitations because the Wi-Fi range is limited, and signal strength may be altered as the range increases. There is no message security, which could lead to information leaks.

In May 2018, Nikhil Chaudhari et al, [8] described a paradigm for intranet users, in which users can communicate and interpret messages in the language of their choice.

It is not necessary to have access to the internet. As a result, communication cost has been reduced. This model also includes image backup, on-demand image theft alerts, and landmark detection. The approach uses juxtaposition as a server, allowing users to communicate via a single network. The messages are limited in size, and when they reach their designated capacity, an exception arises.

Sai Spandhana Reddy Emmadi et al. in January 2019, in [9] examine the current public app and propose a way for sending and receiving messages using a backend server called Firebase to store data. An instant messaging service will be available as a result of this. Only chat rooms would be available to the user.

They are also given the option of using a pdf reader and writer. They also conducted a feedback survey that included questions such as the most loved features, the purpose of using the messaging application, the total amount of time spent on a daily basis, and the most desired chat application, as well as some conclusions.

Ashita et al. (2020) proposed a system [10] that allows two users on the network to communicate using text and text-based media such as images, audio, video, and text online in real-time. To manage contact back functionality, they employed the Android and Google Firebase operating systems, emphasizing numerous characteristics of the application and service.

Noor Sabah et al. (2017), described a concept [11] in which users communicate via a Firebase cloud messaging server, and data is kept locally and secured with a symmetric key. The model also explains how to encrypt communications with a private and public key using the Xsalsa20 encryption algorithm, as well as how to secure sessions with session keys. The user will be able to send chat requests to other users. Node.js and MongoDB are used on the server to offer rapid access.

The paper [12] by Prabhat Kumar Patel et al. on “Android Based Chat Messaging Application Using Firebase” describes the working of an online text messaging application in which users will be able to connect and chat with one another using the application. The app has a sign-up feature that allows users to sign up using their mobile phone number, google firebase is used on the backend side. Members can also use their names to find other users, and chat with them. The authorization feature enables users to access their account from any mobile phone, at any time.

S. Nayak et al. (2017) in their paper [13] describe a concept in which a user-defined password is used to generate a hash, which is then used as the key for the aes256 algorithm to encrypt user messages. The model uses the HTTP protocol to create a connection between both the server and the user, as well as the TCP/IP protocol to transfer data. The connection is established in two modules: the first assists in the addition of new users, while the second assists in the traffic control protocol. The model further makes use of a crypto message server as a mediator between both the database as well as the Google cloud messaging server, which is responsible for establishing a unique id and communication with devices.

Chatterjee et al. (2018) created a mobile app in [14] with the idea of how communication is important in data exchange. The essential topic of this research is real-time data availability. They employed Firebase for real-time data processing communication from the user interface to the cloud and vice versa.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

III. PROPOSED SYSTEM

This application will be an instant messaging application through which the users can communicate smoothly. Both devices that is, the sender and receiver must be connected to the internet in order to communicate. There is no bound regarding the geographical area for using this application. It can be accessed from anywhere around the world. For making this application we have used core java to code this application [17] and google firebase is used to handle the backend part for storing the data. When a user logs in to this application, they can look for another user with whom they need to communicate. Users can upload their images as profile pictures such that other users can able to recognize them easily. This application can only work on android systems.

The user can send text messages, images, emojis to another user. A message notification is sent to the user in the background notification bar when the user is not using this application. The user can see his profile and he can able to edit his profile by updating his name and profile image if he wants and he can also able to upload the image in the form of status in which different users can see it. Apart from this, the user can also express his feelings by reacting to different messages and he can also able to delete the messages accordingly. A group chat room is there in which many users can interact with each other by sending text messages, emojis, and images to each other in the same chat room. Language translator is there which supports more than 50 languages.

If a user wants to translate the language, he can simply translate from the source language to the targeted language, the text translation function is implemented using the Firebase Machine Learning translate on-device Application Interface, which can be used offline and therefore can translate into three distinct languages. When a model gets downloaded, the data can be translated, and also the model can be removed after further use, resulting in a lightweight application. Image to text extraction feature is added in which text gets extracted from the image. The optical character recognition function is built using the firebase machine learning vision Application Interface, which can recognize a variety of languages but is limited to English and Devanagari script in this application. The extracted text is displayed to the user in an editable format. Users can able to see the recent conversation with the time at which the conversation takes place with another person and also, they will be able to see the current status of other users to check whether they are online/offline or typing a message. If a user wants to log out then he can simply click on the menu item and then click on the logout option and the user can also log in with a different device if he wishes.

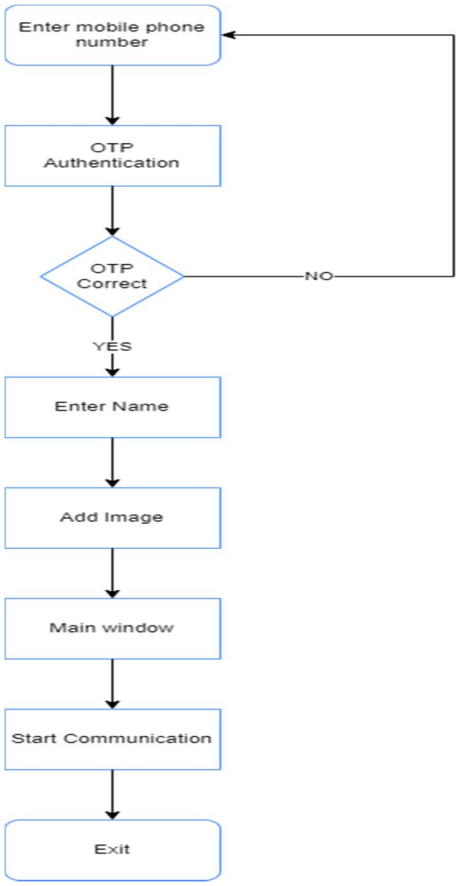


Fig. 1 Working flow chart of application

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue V May 2022- Available at www.ijraset.com

A. Start Module

When the application is launched, this is the first module to be launched. The system first determines whether or not the user is online, and then whether or not the user is connected. If the user is connected to this application and is also connected to the internet, then this module is opened, giving the current user an option to log in to their account if a user is not logged in to this application.

B. Authentication

The user's identification is required by the application, which aids in the security of the user's data in the cloud. Google Firebase [18] provides SDKs that allow developers to more securely authenticate their accounts. To use this feature, users need to integrate the firebase authentication into their application. This method allows users to log in to their app using their phone number. If the phone number matches the existing records, a message of login successfully will be displayed and it will allow the user to proceed further. If the number is not present in the existing records, then the user has to log in to an account with a valid number. If the new user has entered a phone number and clicked on a submit button then it will proceed to the One Time Password window for verification. The user has to enter the correct OTP to proceed further and if the entered OTP is not right the user will have to follow again the same previous process. If the entered password is correct then the user will proceed to the profile section, in this user can add his profile image if he wants and he has to enter his name. A user must enter his name in the profile section otherwise his profile will not be created. After this, he has to click on the setup profile button and the user will navigate to the main window and will be able to communicate with other users.

C. User Application Interaction

The application will provide capabilities such as one-to-one communication, community group messaging, information sharing, extraction of text from an image, picture conversion, language translation, and altering profile name and image once the user has completed the authentication process. The app will have tabs with names like profile, group chat, and users, as well as search features that will aid in data sorting. The application can be used according to the needs of the user; there are additional menus that provide options such as logging out and help. The chat will be preserved in the database, and the user will always have access to it. The chat activity page would allow users to transmit text messages as well as data such as PDFs, photos, and documents.

D. Chat Module

This is the system's module, which will handle the system's messaging activity. When a user wants to communicate with other users on the network, this module is activated. Users can send or receive messages in the form of text, images, or other files in this module. A Firebase database reference is created along with the other UI elements when this module is launched. We load any prior messages with a method that gets previously transmitted messages when we launch. If the users send a message after initializing all elements, a check is made to see if the sent message is a text, image, or other files. After determining the type of message, a function is called to upload the image, text message, or any other file to Firebase Storage and create a record in the Firebase Real-time Database, which allows us to track how well our application is operating.

E. Language Translation

Firebase also includes a Machine Learning Kit, which includes a variety of APIs. We can retrieve the string's most probable language or confidence scores for all of the string's available languages by using the kit to determine the language of a text string. In addition, Romanized text for English, Hindi, Japanese, Spanish, and French can be recognized. On-device Firebase offline translation models are used to implement the Translation model.

A language's translation model can be up to 30MB in size. As a result, just three languages are available in this manner, that can be downloaded and deleted as needed. This model is used to transform languages from one source language to another. After the text input is given, the translation model checks to see if the translation model has already been downloaded, the model must be downloaded before translation, and afterward, the model converts the input text to the chosen target language. To begin the translation procedure, we have to call the translate method and also give the text to be translated this approach converts text from one language to another. we need to use a callback listener to obtain the successfully translated text. The success and failure listeners were implemented. When our text is correctly translated, the onSuccess() function is called, and if there is an error, the onFailure() method is called. When the models have successfully downloaded, we must construct a callback listener to obtain the download done event.

F. Image to Text Extraction

We utilized the Machine Learning Kit to recognize text in photographs. On-device and cloud-based models are available for the general-purpose API. Run the text recognizer to recognize text in an image using an on-device or cloud-based model. We have used the FirebaseVisionImage object to extract text from a Bitmap in this scenario. Then, using the Vision Text, create a text section made up of Blocks, Lines, and Elements.

A block is a group of text lines that are connected together, such as a paragraph or a column. A line is a group of words that are all on the same vertical axis.

A contiguous group of alphanumeric characters on the same vertical axis is called an element. We obtain a FirebaseVisionText object inside the success listener if the text recognition worked successfully. This FirebaseVisionText object provides all of the image's textual information.

The user-input text is extracted using the ML Kit's on-device Image Text Extraction (Recognize text) Dependency, which can then be utilized for language identification and translation. In the current version of this Kit, the Cloud Recognize text API can extract user-input text from documents, photos, and pictures. This can also recognize a variety of different Latin texts, including Romanized, Hindi, Japanese, and Chinese, among others. In our app, it can recognize English as well as Devanagari script from the image.

G. Save Our Souls

In case of an emergency, this feature is very useful. The user's current location is sent to the users which are added by the current user in this application. A User can click on add relative option to add the contacts. In this, the user needs to add the name and phone number of others to whom a user wants to send an alert message. An emergency button is there in which when the user clicks on that button then a message in text form is sent to the contacts added by the user with the current location of the user. Certain helpline numbers are there to which users can directly call with a single click when the user is in trouble.

IV. RESULT

Following the completion of this system, we are left with a completely functional messaging application capable of exchanging real-time messages and photographs. To begin, the user must complete the one-time password authentication process so that his phone number could be correctly validated.

The prior messages, if any, would be displayed on the application's main page. Users can respond to messages with an emoji, such as a joyful face, to make communication more visual and entertaining. A language translator has been added which supports more than 50 languages and an Image to text extraction feature is there in which the text from an image will be extracted which is recognized in English as well as Devanagari script.

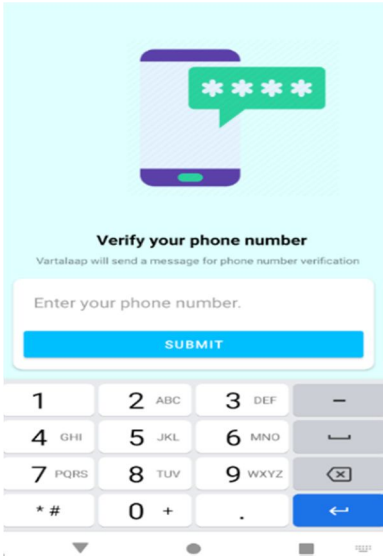


Fig. 2 Login page

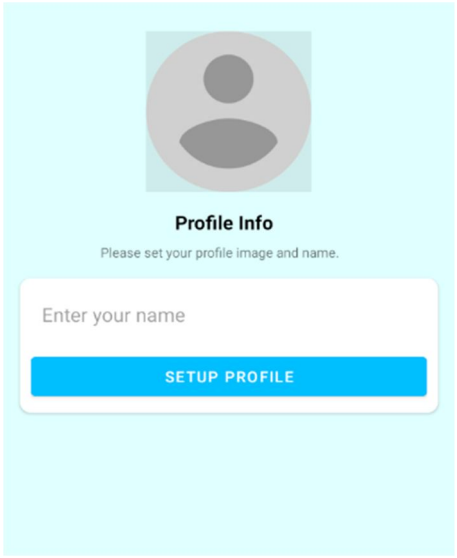


Fig. 3 Profile setup

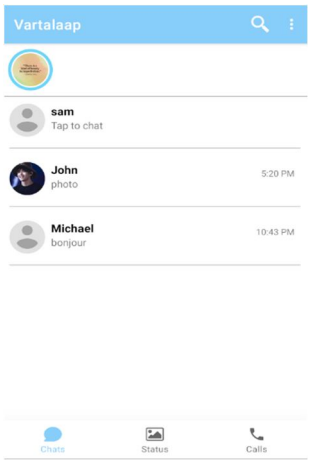


Fig. 4 The interface of main window

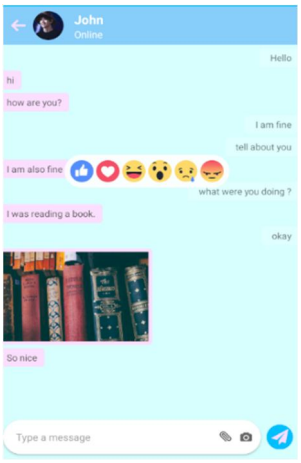


Fig. 5 A chatting screen

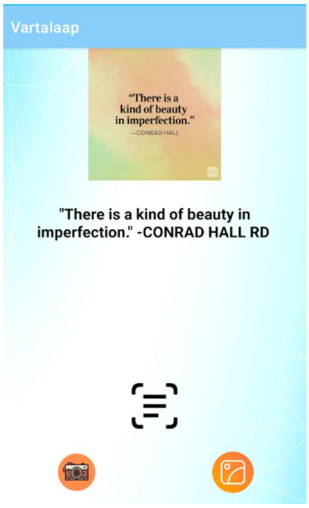


Fig. 6 Image OCR

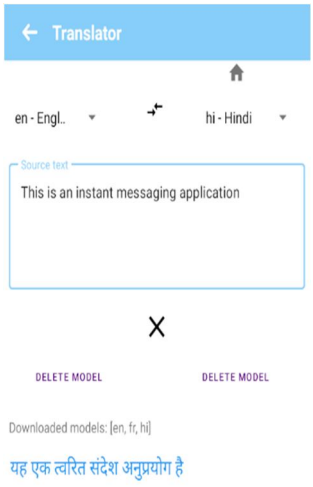


Fig. 7 Text Translation

V. CONCLUSION AND FUTURE SCOPE

We created an instant messaging application that is capable of exchanging messages along with an image to text extraction and language translation that will require improvements in the future, and we will work on them concurrently. There is always an opportunity for improvement, no matter how efficient an application is. As long as technology advances, the application will continue to improve. As a result, the software is incredibly future-proof.

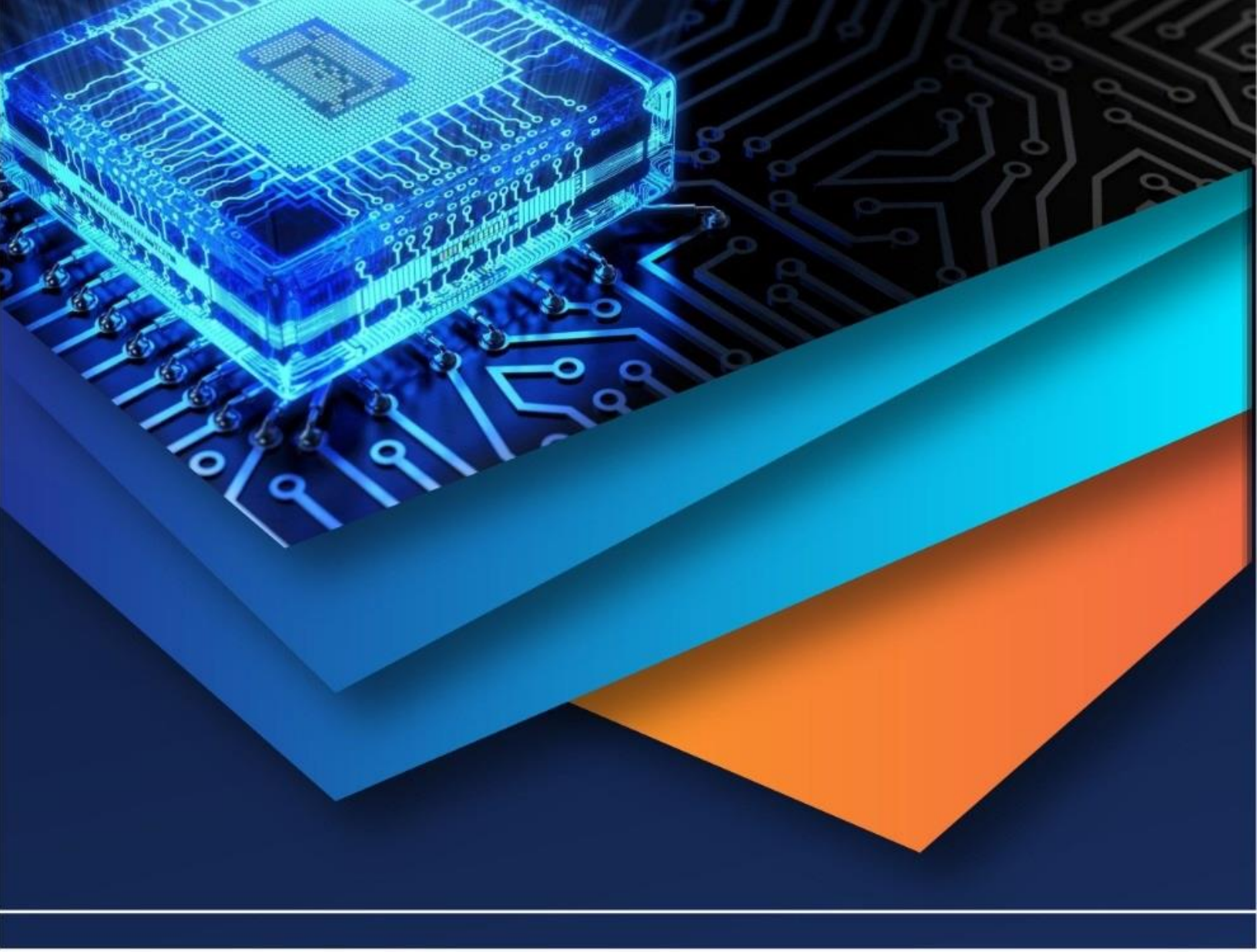
Our application can be expanded to include additional functions in the future, such as:

1) Audio and video calling
2) Voice messaging
3) Auto delete messages
4) Group calling

REFERENCES

[1] Ovadia, S. (2002). Real-Time Reference and the Importance of Text-Chat. The Reference Librarian, 38(79-80), 157-161.
[2] Soules, A., 2008. I Google, You Google, We Google... Against the Grain, 20(2).
[3] Li Ma et al, Research and Development of Mobile Application for Android Platform, International Journal of Multimedia and Ubiquitous Engineering 9(4):187-198 • April 2014
[4] Omidi, M. and Alipour, M., 2016. Why NOSQL And The Necessity of Movement Toward The NOSQL Data Base. IOSR Journal of Computer Engineering, 18(05), pp.116-118
[5] Gerber, A., & Craig, C. (2015). Programming in Android Studio. Learn Android Studio, 45–67.

[6] Albertengo, G., Debele, F. G., Hassan, W., & Stramandino, D. (2020). On the performance of web services, google cloud messaging and firebase cloud messaging. Digital Communication and Networks, 6(1), 31-37
[7] Ekata M. Lambture and Prof. Z.M. Shaikh "ANDROID CHATTING APPLICATION BASED ON WI-FI TECHNOLOGY" International Journal of Engineering Sciences & Research Technology, May 2016
[8] Nikhil Chaudhari, Sushma Shinkar and Priyanka Pagare "Chatting Application with Real Time Translation" International Research Journal of Engineering and Technology Volume: 05 Issue: 05, May 2018
[9] Sai Spandhana Reddy Emmadi and 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
[10] Ashita, Vasudha Bahl, Dr.Amita Goel and Nidhi Sengar, "Android Based Instant Messaging Tool Using Firebase as Backend", International Journal for Modern Trends in Science and Technology, 6(12): 198-201, 2020.
[11] Noor Sabah, Jamal Mohamad and Ban N. Dhannoon "Developing an End-to-End Secure Chat Application" IJCSNS International Journal of Computer Science and Network Security, VOL.17 No.11, November 2017
[12] Prabhat Kumar Patel, Shaina Gupta "Android Based Chat Messaging Application Using Firebase" International Journal of Creative Research Thoughts (IJCRT) Volume: 9 Issue: 5, May 2021
[13] S. Nayak, S. Das, S. Das, S. Sarker, P. Sarker, A. Dey, A. Sinha, J. Saha, A. Banerjee, N. Saha, S. Chowdhury, D. Chowdhury, P. Pradhan, S. A. Ali, A. Saha, R. Dey, S. Dey, "An application for end to end secure messaging service on Android supported device", 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2017, Vancouver, BC, Canada.
[14] Chatterjee, Nilanjan & Chakraborty, Souvik & Decosta, Aakash & Nath, Asoke. "Real-time Communication Application Based on Android Using Google Firebase" IJARCSMS, Volume 6, Issue 6, April 2018
[15] Rajsbaum, S. and Viso, E., 2005. Object-oriented algorithm analysis and design with Java. Science of Computer Programming, 54(1), pp.25-47.
[16] Yevick, David, A First Course in Computational Physics and Object-Oriented Programming, Cambridge University Press, 2005. ISBN 0-521- 82778-7
[17] Javed Ahmad Shaheen et al, Android OS with its Architecture and Android Application with Dalvik Virtual Machine Review, International Journal of Multimedia and Ubiquitous Engineering Vol. 12, No. 7 (2017), pp. 19-30
[18] Khawas, C. and Shah, P., 2018. Application of Firebase in Android App Development-A Study. International Journal of Computer Applications, 179(46), pp.49-53



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY