

**VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF
TECHNOLOGY**

Department of Computer Engineering



Project Report on

Collaboratory tool in Android

In partial fulfillment of the Final Year, Bachelor of Engineering (B.E.) Degree in Computer Engineering at the University of Mumbai Academic Year 2020-2021.

Submitted by

Anish Adnani D17B 01

Nihal Bhandary D17B 10

Supriya Patil D17B 50

Project Mentor

Mr. Richard Joseph

(2020-2021)

**VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF
TECHNOLOGY**
Department of Computer Engineering



Certificate

This is to certify that **Anish Adnani, Nihal Bhandary, and Supriya Patil** of Final Year Computer Engineering studying under the University of Mumbai have satisfactorily completed the mini project on "**Collaboratory tool in android**" as a part of their coursework of Mini Project for Semester-VIII under the guidance of their mentor **Mr. Richard Joseph** in the year 2020-2021.

This mini project report entitled (*Collaboratory tool in android*) by (*Anish Adnani, Niha Bhandary, Supriya Patil*) is approved for the degree of **(Cloud Computing)**.

Program Outcomes

Grad
e

PO1, PO2, PO3, PO4, PO5, PO6, PO7,
PO8, PO9, PO10, PO11, PO12 PSO1,
PSO2

Date:

Project Guide: Mr. Richard Joseph

Mini Project Report Approval For B. E

(Computer Engineering)

This mini project report entitled ***Collaboratory tool in android*** by ***Anish Adnani, Nihal Bhandary, Supriya Patil*** is approved for the degree of B.E Computer Engg.

Internal Examiner

External Examiner

Head of the Department

Principal

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Name of student and Roll No.)

(Signature)

(Name of student and Roll No.)

(Signature)

(Name of student and Roll No.)

(Signature)

(Name of student and Roll No.)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to Assistant Professor **Mr. Richard Joseph** (Project Guide) for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr. (Mrs.) J.M. Nair** for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement at several times.

Computer Engineering

Department COURSE OUTCOMES FOR B.E Mini

Project Learners will be,

Course Outcome	Description of the Course Outcome
CO 1	Able to apply the relevant engineering concepts, knowledge and skills towards the project.
CO2	Able to identify, formulate and interpret the various relevant research papers and to determine the problem.
CO 3	Able to apply the engineering concepts towards designing solution for the problem.
CO 4	Able to interpret the data and datasets to be utilized.
CO 5	Able to create, select and apply appropriate technologies, techniques, resources and tools for the project.
CO 6	Able to apply ethical, professional policies and principles towards societal, environmental, safety and cultural benefit.
CO 7	Able to function effectively as an individual, and as a member of a team, allocating roles with clear lines of responsibility and accountability.
CO 8	Able to write effective reports, design documents and make effective presentations.
CO 9	Able to apply engineering and management principles to the project as a team member.
CO 10	Able to apply the project domain knowledge to sharpen one's competency.
CO 11	Able to develop professional, presentational, balanced and structured approach towards project development.
CO 12	Able to adopt skills, languages, environment and platforms for creating innovative solutions for the project.

Abstract

There are various advantages that collaborative tools offer to teams, as 75% of the workmen rate teamwork and collaboration as “very important” for their business.

Document Collaboration tools have a lot of benefits. Some of these include:

- Increased productivity
- Brainstorming ideas
- Faster problem solving
- Enhanced creativity
- Faster execution of ideas
- Builds a sense of community

The purpose of this project is to basically develop an android application to include functionalities that helps the user to login into the portal via any email id. The user is then assigned a unique id (used to store all the user information). Users can create new text documents or can upload multimedia files (stored in Firebase Database)and share them with other users on the portal. To improve the security of the system we have implemented Hybrid cryptography which basically divides the data into 3 different parts and all the parts are encoded using Addition Cipher with different keys, which reduces the chances of data breach and attacks. Further an in-app chatbot support is provided to the user for general queries that decreases the workload of the team.

Our system falls under the “PaaS” category of cloud computing as it includes Database management, security algorithms. This app provides a platform for other users to upload their file on the internet and share it with other users safely without worrying about the complexity of building and maintaining the infrastructure typically associated with developing and launching the application, which is the basic definition of Platform as a Service “PaaS”

INDEX

Content	Pg no.
Chapter 1: Introduction	
1.1 Introduction.....	12
1.2 Motivation.....	12
1.3 Problem Definition.....	13
1.4 Relevance of the Project.....	13
1.5 Methodology employed for development.....	13
 Chapter 2: Literature Survey	
2.1 Research Papers Referred.....	15
a. Abstract of the research paper	
b. Inference drawn	
 Chapter 3: Requirement Gathering	
3.1 Functional Requirements.....	17
3.2 Non-Functional Requirements.....	17
3.3 Constraints.....	18
3.4. Hardware, Software, Technology and tools available.....	18
3.5 System Block diagram.....	19
 Chapter 4: Proposed Design	
4.1 System Design.....	20
4.2 Detailed Designs.....	21
4.3 Project Scheduling & Tracking using Timeline / Gantt Chart.....	23
 Chapter 5: Implementation Details	
5.1. Methodology employed.....	24

5.2. Algorithms used.....	26
5.3 Screenshots (GUI) of the project.....	30

Chapter 6: Conclusion

6.1 Limitations.....	38
6.2 Conclusion.....	38
6.3 Future Scope.....	39

Chapter 7: References

7.1. Research Papers Referred.....	40
------------------------------------	----

LIST OF FIGURES

Figures	Pg. No
Fig 1 : Incremental model	<u>13</u>
Fig 2: System Block diagram	<u>19</u>
Fig 3 : System Design	<u>20</u>
Fig 4 : Flowchart	<u>21</u>
Fig 5 : Use Case Diagram	<u>22</u>
Fig 6 : Gantt Chart	<u>23</u>
Fig 7: Various features available on Firebase	<u>24</u>
Fig 8: Shows a row of user data stored in Authentication	<u>24</u>
Fig 9: Storage of Multimedia file	<u>25</u>
Fig 10: Storage of text files	<u>25</u>
Fig 11: Demonstration of Encoding	<u>26</u>
Fig 12: Time Taken by Polycipher	<u>27</u>
Fig 13: Time Taken by Base64 encoding	<u>27</u>
Fig 14: User Registration Screen	<u>30</u>
Fig 15: Verification link sent on email	<u>30</u>
Fig 16: Login Page	<u>31</u>
Fig 17: User Home Page	<u>31</u>
Fig 18: Shared Fragment	<u>32</u>

Fig 19: Profile Fragment	<u>32</u>
Fig 20: Available options	<u>33</u>
Fig 21: Sharing file to a particular user	<u>33</u>
Fig 22: Viewing a file	<u>34</u>
Fig 23: Viewing/ Editing a document	<u>34</u>
Fig 24: Chatbot support	<u>35</u>
Fig 25: Supports any online english assistance tool	<u>35</u>
Fig 26: Displaying a list of users registered in the app	<u>36</u>
Fig 27: Documents Table Structure	<u>36</u>
Fig 28: Showing structure of User Table	<u>37</u>
Fig 29: Firebase Database storage (saving multimedia files)	<u>37</u>

Chapter 1

Introduction

1.1 Introduction to the project

Reports are the most common form of communication between the company's hierarchy.

Reports are not only drafted by the job-oriented class but reports are also done as a part of various classwork of students, publishing a research paper, etc. Reports done in a group do require a systematic approach to avoid inconsistency.

We aim to develop a cloud-based android application that provides a free portal to all the users to register and use the services. Functionalities include uploading a new multimedia file (images, videos, gifs, etc), creating a new text document, sharing the files with other users, in-app chatbot support for general queries. Moreover, all the data is encrypted using hybrid cryptography which enhances the security over traditional ciphers used commonly. This system will allow the users to work efficiently and consistently at no cost.

1.2 Motivation

Sometimes combining multiple documents may get difficult as each user has to write their own document and then to combine them takes a lot of time. We intend to develop a system that shall solve this problem by allowing users to share the document so multiple users can edit a document at the same time, while seeing each other's changes instantaneously. It will allow users to access the information from anywhere. The document owner retains control of the file. Hence he can add or remove the users according to his preferences. For any organization security of data is always a concern so by taking this into consideration we have added a security aspect which will encrypt the data using hybrid cryptography hence the data will be safe and secure.

1.3 Problem Definition

To create a system that shall be available to all the people at free of cost. We have considered developing an android application for “a collaborative tool”. This easily accessible android application can be easily used by anyone by just registering onto the application and users can avail facilities to create/ upload new documents/files and share it with other users.

1.4 Relevance of the Project

Our system adheres to those people who want to minimize their workload by accessing only one document where multiple users can edit the same instantaneously. It even allows users to see who else is making changes to it as they work. Users can also add images, videos, gifs in the document to make it interactive. It is very easy to handle. It also allows the owner to add or remove the users when working on specific documents. That way users can control exactly who can make changes to the document. And to make the system more secure we have added encryption technology which will make users data more secure.

1.5 Methodology

Considering the fact that any system can be best developed when using incremental model, we too have also followed the process of incremental model.

This model will allow us to improve the system as well as increase the size in terms of its functionality, configurations, accessibility, security, etc.

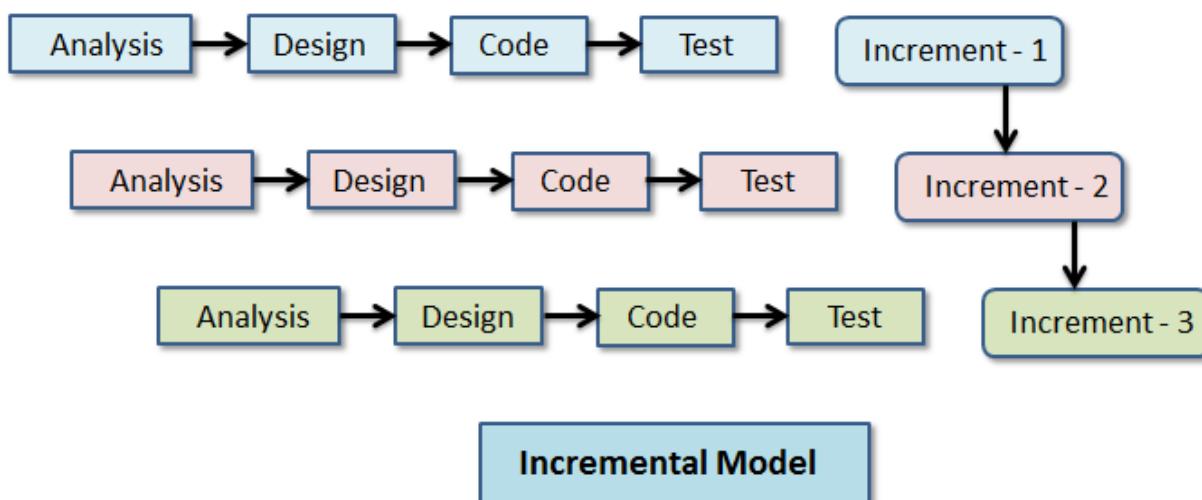


Fig 1 : Incremental model

For each increment we decided a set of functionalities to be included in the application.

Further iteration are made based on improvements that can be done to the existing system and adding new functionalities to the application.

We decided to use Firebase Database, to store the user data and provide a real time secure medium for the users. Firebase Database “Spark” plan is a free to use plan which provides developers with most of the features to build a complete full fledged application.

“Firebase Storage” is used to store multimedia objects uploaded by the user.

“Firebase Realtime Database” is used to store the data with security rules.

Chapter 2

Literature Survey

2.1 Research Papers Referred

- Firebase is considered a major web application platform. Particularly known for its NoSQL structured real time database and various services attached to it. The Firebase supports development of Khawas, Chunnu & Shah, Pritam. (2018). Application of Firebase in Android App Development-A Study. International Journal of Computer Applications. 179. 49-53. 10.5120/ijca2018917200. applications across various platforms including web-based applications, mobile-based applications, desktop-based applications. The paper describes the particular use of firebase in android applications. Firebase offers authentication services, database services.
- S. Wen and W. Dang, "Research on Base64 Encoding Algorithm and PHP Implementation," 2018 26th International Conference on Geoinformatics, 2018, pp. 1-5, doi: 10.1109/GEOINFORMATICS.2018.8557068. Base64 is one of the most widely used encoding methods for transmitting 8-bit byte codes on the internet, it can be used to transmit long identification information in HTTP environment. Its advantage is that the efficiency of the algorithm is high, the coded results are short, also unreadable.
- M. A. Mokar, S. O. Fageeri and S. E. Fattoh, "Using Firebase Cloud Messaging to Control Mobile Applications," 2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), 2019, pp. 1-5, doi: 10.1109/ICCCEEE46830.2019.9071008. The system proposed in this paper is a new framework that can be utilized by any application programmer to use it for controlling mobile Application by using FCM technology Stand for (Firebase Cloud Messaging), which is supported by Google. The system is designed to deal with more than one Firebase application at the same time and send data messages that are designed by the programmer either by creating these data manually or to get this data from another environment.

- “Data StructureComparisonBetween MySql Relational Database andFirebase Database NoSqlon Mobile Based Tourist Tracking Application”, K G Sudiarta, N E Indrayana , W Suasnawa , S A Asri , Putu Wijaya Sunu, et al 2020 J. Phys.: Conf. Ser.1569 032092 . The Firebase Database is a NoSqlcloud-based database that synchronizes data across all clients in real time .Data from the Firebase realtime database is stored as JSON and synchronized in realtime to each connected client. This data is stored cloud hosted. When developers create cross-platform applications with Android, iOS, and JavaScript SDK, all clients will share a Realtime Database instance and receive the latest data updates automatically. Firebase is not like an SQL database, firebase databases do not have tables or records. When you add data to a JSON tree, the data will become a node in the existing JSON structure with the associated key.

Chapter 3

Requirement Gathering

3.1 Functional Requirements

- The user should be able to register onto the application by using any email-id
- The user should be able to access their account from multiple devices
- The user should be able to create new text documents and upload any multimedia files through the portal
- Functionally, user should also be able to share files with other users present onto the portal
- Current editors of the document should also be shown
- In-case any user forgets his/her password, password reset link should be sent to their registered email-id
- All the text documents are encrypted to securely store the data onto Firebase Database
- Users should have access to the list of peers that the file is shared with, so that access can be removed at any point of time
- In-app chatbot support for general queries

3.2 Non Functional Requirements

- Users on registering, receive an email to verify their account, without which the user cannot access the account
- The application should be compatible with most of the android versions, with little to no change in the functionality implemented
- It should be robust, i.e secure from privacy threats
- All documents and files should be available all day long, as availability is one of the key factors
- The system should be easy to use and accessible
- There must be recoverability, i.e data once deleted should be available for backup
- It should be user friendly, i.e there should be guidance provided as how to use the system that might be helpful for the new users.

3.3 Constraints

- There should be an active internet connection to create new files or to upload files from the device
- Documents can only contain text, images cannot be included in documents due to lack of universal links
- Files can only be shared with people registered onto the application
- People do not receive an app notification if there is some file shared with them, which can make it difficult to identify that a new file is shared

3.4 Hardware and Software Requirements

Hardware Requirements

- As we are using Firebase Database, we do not have any specific hardware requirements to setup the cloud
- Recommended RAM to develop an application on android studio is 6GB, for smooth functioning of the mentioned windows application

Software Requirements

- JAVA
- Android Studio
- Firebase Database
- Paper DB
- DialogFlow console
- Kommunicate
- Google gms services
- Androidx appcompat library (so that app layout is same in various phones with varied dimensions)

Students requirements

Hardware Requirements

- An android phone with android 7+ to use the application

Software Requirements

- Chrome to access multimedia files uploaded onto the app
- Gmail to access mails sent from server

3.5 System Block Diagram

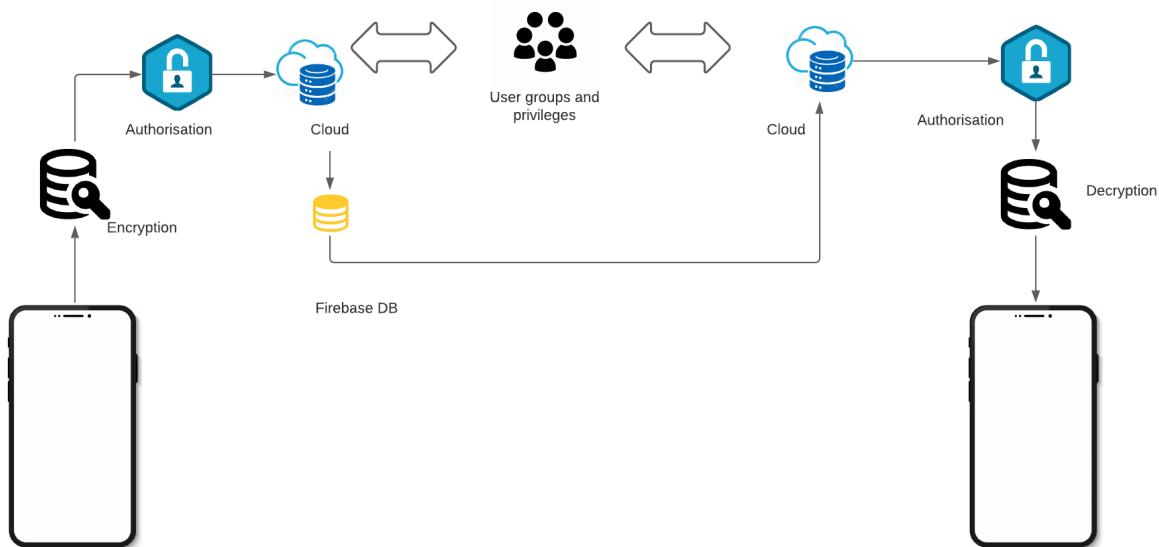


Fig 2: System Block diagram

As seen from the above diagram all the data stored by the users is encrypted.

The user needs to Register on the app via any email id which will be verified by sending a verification email. Users can then use the registered email id to use the services provided in the application.

Users Data is stored in Firebase Database for real time and quick access. Users can add privileges for other users accessing the information. Data will only be accessible to shared users. All the data is decrypted before displaying it in the user application.

Chapter 4:

Proposed Design

4.1 System Design

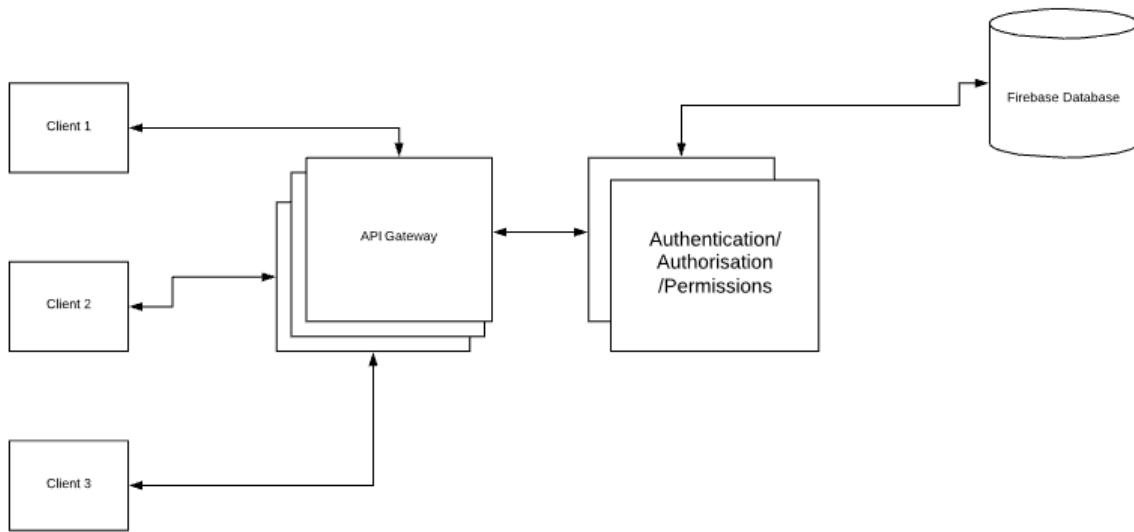


Fig 3 : System Design

As shown in the figure, clients interact through Firebase SDK in Android to access contents from the database while maintaining privileges. API gateway allows Firebase to be used in Android application. API uses the rules written in Firebase rule to check the user for authentication, document permission rules, providing the authorisation privileges to the owner of the document,etc. NoSQL database is the Firebase Realtime Database which contains the data in JSON like format in form of key-value pairs. The data is encoded at the client side, this encoded data is then stored under a particular key. When a user shares access to another key, the permission for the said user to access the document is written by writing this particular key under “accessible documents” node in the user id.

4.2 Detailed Design

Flowchart Diagram:

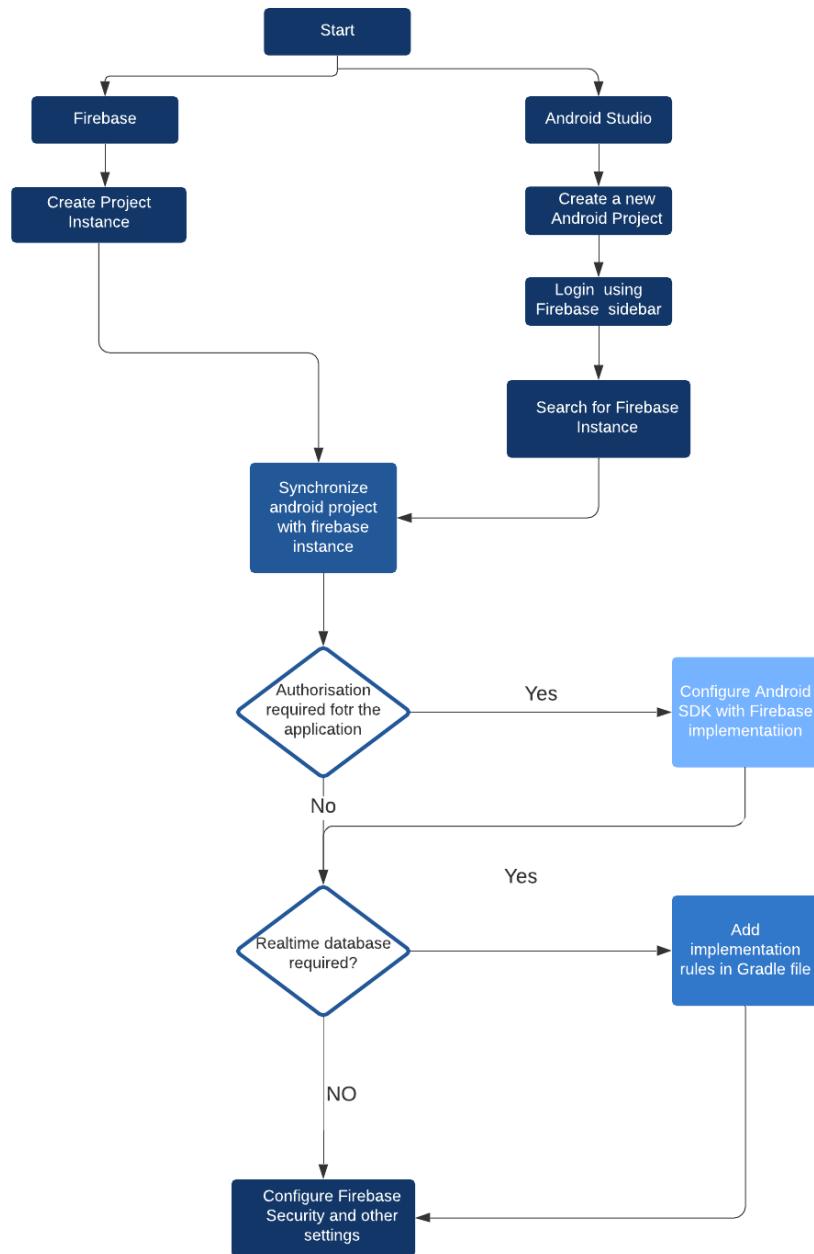


Fig 4 : Flowchart

Use Case Diagram:

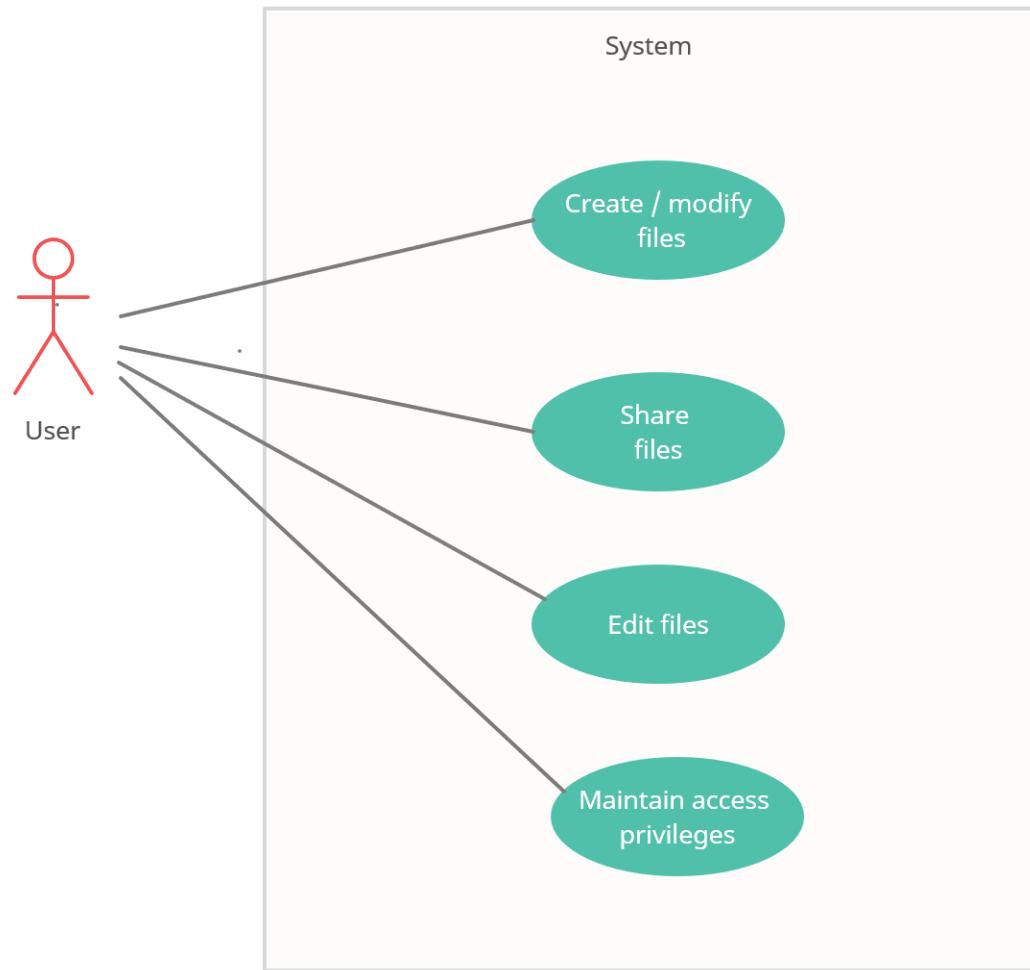


Fig 5 : Use Case Diagram

4.3 Project Scheduling and Tracking using Gantt Chart

	January			February			March			April		
	1-10	10-20	20-31	1-10	10-20	20-28	1-10	10-20	20-31	1-10	10-20	20-30
Requirements gathering												
Research												
Guidelines												
Project Initiation												
Planning												
Planning of resources												
Planning of tools to be used												
Modelling												
Prepare System Design												
Prepare Detailed Designs												
Building and Deployment												
Coding												
Testing												
Deployment												
Review												
Implementing changes												
Testing												
Deployment												
Preparing mini project Report and presentation												

Fig 6 : Gantt Chart

Chapter 5

Implementation Details

5.1 Methodology Applied

Firebase is a platform developed by google for creating android and web applications. Main advantage of using a Firebase database for storing data is that it allows real time storage of data and real time retrieval of data without any delay.

When any app is integrated with Firebase Database all the user analytics are maintained by the company without us having to worry about the implementation of the logs.

There are various features available in Firebase

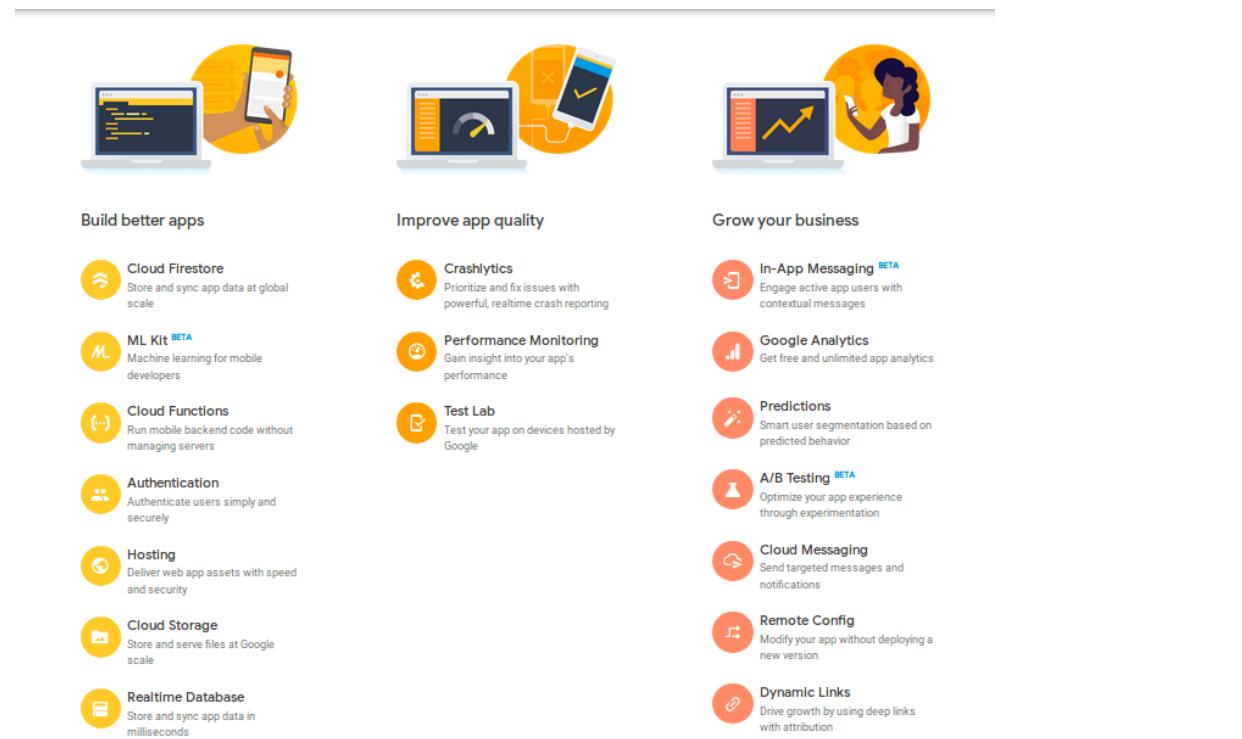


Fig 7: Various features available on Firebase

Registration

Users will register themselves, the data of which will be stored in the Authentication page of Firebase and the passwords are managed by Firebase, hence reducing the chance of security breach.

Fig 8: Shows a row of user data stored in Authentication

File Creation/ Uploading

Users can upload and create new files which will be stored in the “Documents” table. Document stores the following information file_id, filename, owner id, shared node (which contains a list of users the file is shared with) and url in case if the file is multimedia. Files are stored in Firebase storage and their access link is copied into the url of “Documents” table. If the file is a text file it also contains the text stored in encrypted format.

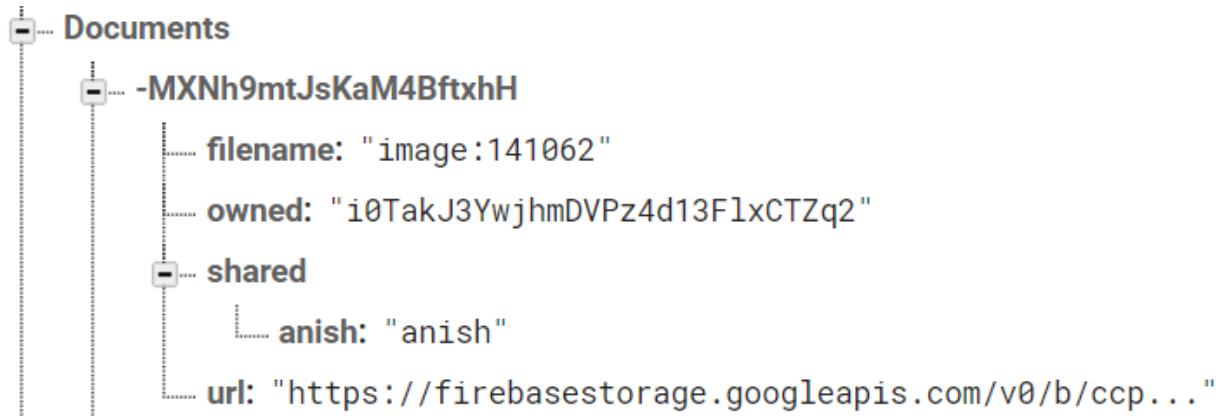


Fig 9: Storage of Multimedia file

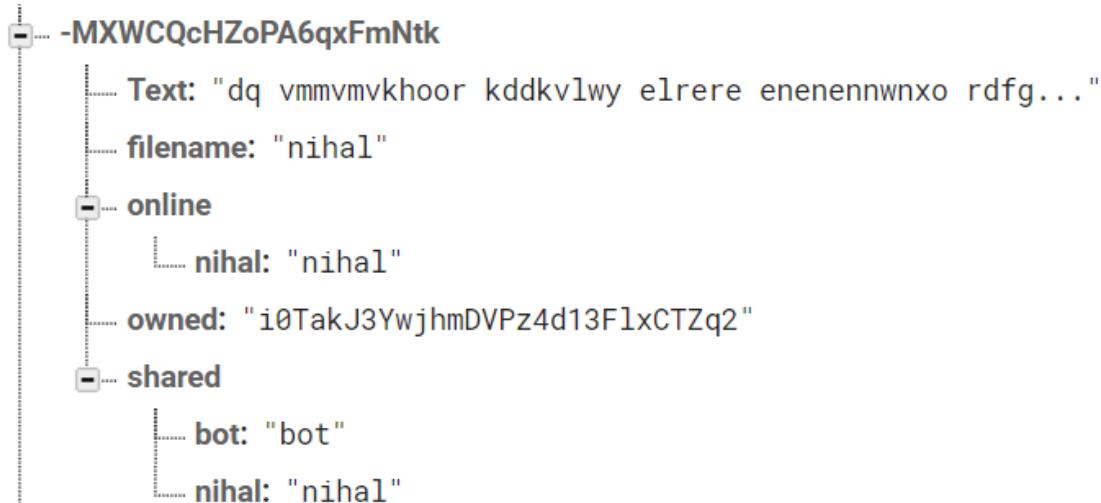


Fig 10: Storage of text files

Security (Encoding)

All the files are encoded using hybrid cryptography ,which provides a secure medium for the text files to be stored.

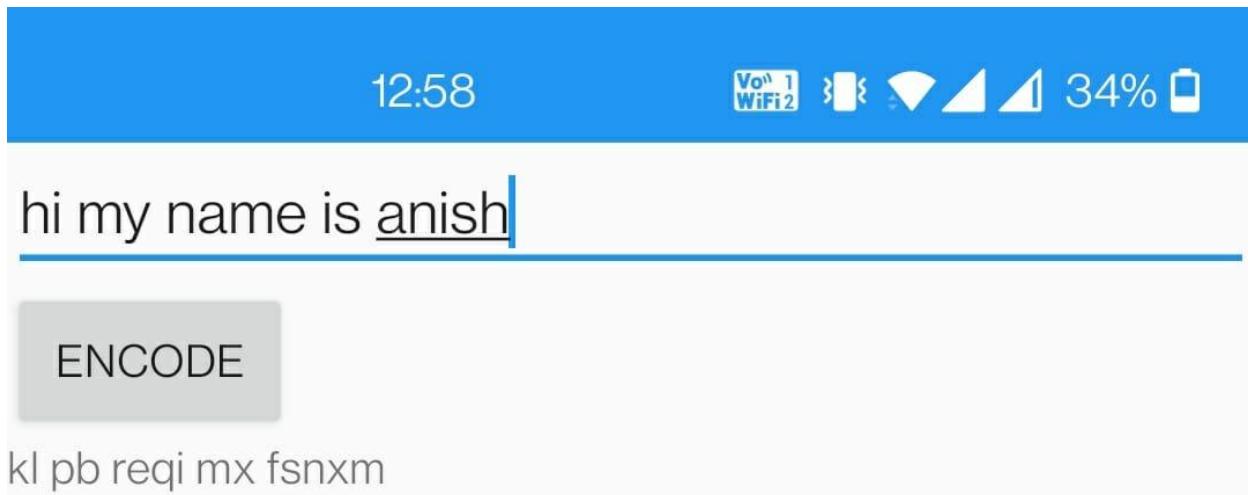


Fig 11: Demonstration of Encoding

5.2 Algorithms Used

There are various algorithms used in the development of such a kind of collaborative tool
Comparison of Base-64 encoding vs Hybrid Cryptography

	Base 64	Poly - Cipher
Time required for encoding	5 ms	13 ms
Time required for decoding	1 ms	10 ms
Key	none	3 keys used for encoding and decoding
Space after encoding	Increases after encoding	Remains the same
Security aspect	Easily breakable.	It takes considerable time and effort to break the cipher. As it takes multiple keys for decoding it. In case of a key leak, most part of the text will be undecipherable as all keys are required to decode the text.
Implementation	Easy to implement	Comparatively requires more lines of code
Security	Security is higher as it does not involve keys to encrypt and decrypt	Security can be compromised if the encryptions keys are not stored securely

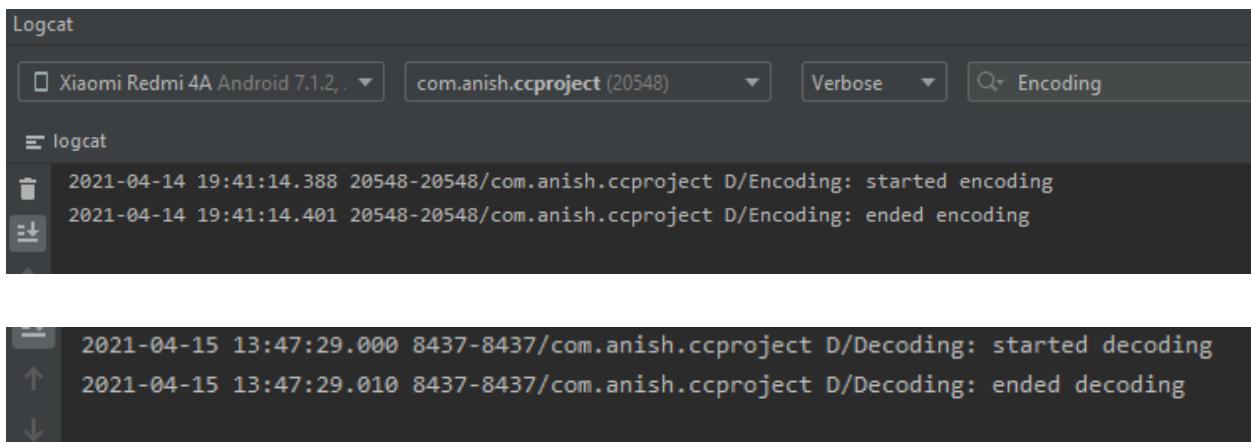


Fig 12: Time Taken by Polycipher

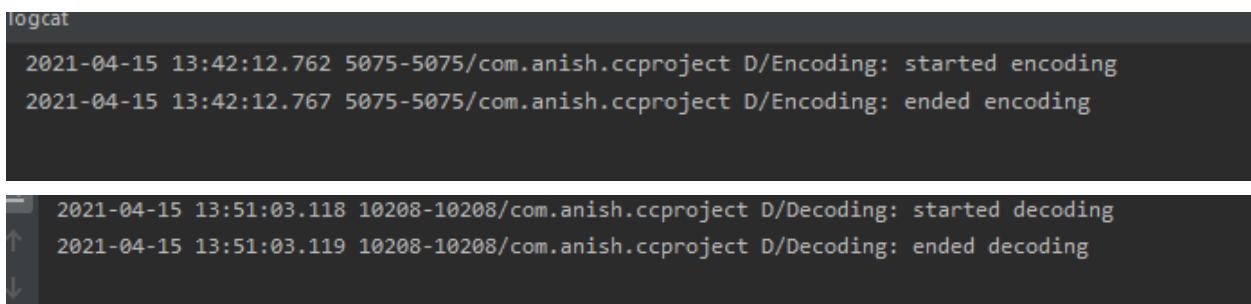


Fig 13: Time Taken by Base64 encoding

Hybrid cryptography

Step 1 includes dividing the strings into various parts so that each part can be dealt with individually. For consideration we have divided the string into 3 parts

```
int onethird = input.length()/3;
int twothird = 2*onethird;
```

Step 2 includes encoding the string using any cipher (for consideration we are using addition cipher) with different keys.

```
String a = encrypter(input.substring(0,onethird),3);
String b = encrypter(input.substring(onethird,twothird),4);
String c = encrypter(input.substring(twothird),5);
return a+b+c;
```

Where the encrypter function is as follows

```
protected String encrypter(String inputString, int key){
    StringBuffer output;
    Character charac;
```

```

int previousAscii,newAscii;
output = new StringBuffer();

for(int i=0;i<inputString.length();i++){
    charac = inputString.charAt(i);
    if(charac.equals(' ')){
        output.append(Character.toString(charac));
        continue;
    }
    if(!Character.isLetter(charac)){
        output.append(Character.toString(charac));
        continue;
    }
    previousAscii = (int)charac;
    newAscii = previousAscii + key;
    if(newAscii > 90 && Character.isUpperCase(charac) || newAscii > 122){
        newAscii -= 26;
    }
    output.append(Character.toString((char)newAscii));
}
return String.valueOf(output);
}

```

To decrypt the string while displaying it to the user reverse procedure is followed

```

int onethird = input.length()/3;
int twothird = 2*onethird;
String a = decrypter(input.substring(0,onethird),3);
String b = decrypter(input.substring(onethird,twothird),4);
String c = decrypter(input.substring(twothird),5);
Log.d("Decoding","ended decoding");

return a+b+c;

```

Where decrypter function is as follows

```
protected String decrypter(String inputString, int key){  
    StringBuffer output;  
    Character charac;  
    int previousAscii,newAscii;  
    output = new StringBuffer();  
  
    for(int i=0;i<inputString.length();i++){  
        charac = inputString.charAt(i);  
        if(charac.equals(' ')){  
            output.append(Character.toString(charac));  
            continue;  
        }  
        if(!Character.isLetter(charac)){  
            output.append(Character.toString(charac));  
            continue;  
        }  
        previousAscii = (int)charac;  
        newAscii = previousAscii - key;  
        if(newAscii < 65 && Character.isUpperCase(charac) || newAscii < 97){  
            newAscii += 26;  
        }  
        output.append(Character.toString((char)newAscii));  
    }  
    return String.valueOf(output);  
}
```

5.3 Screenshots of the project

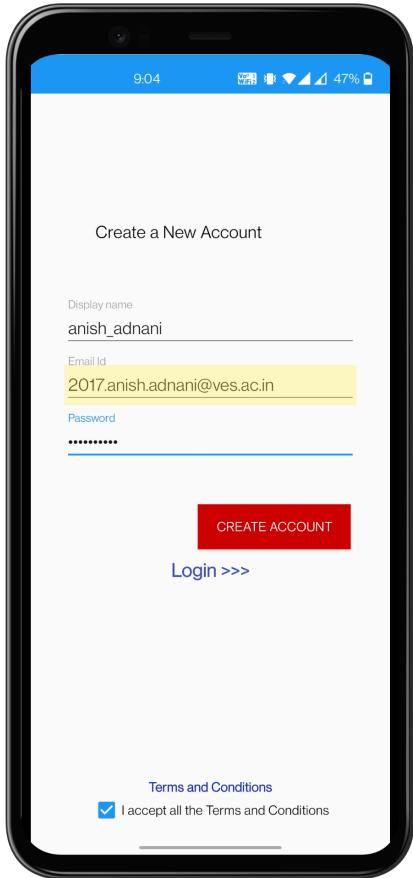


Fig 14: User Registration Screen

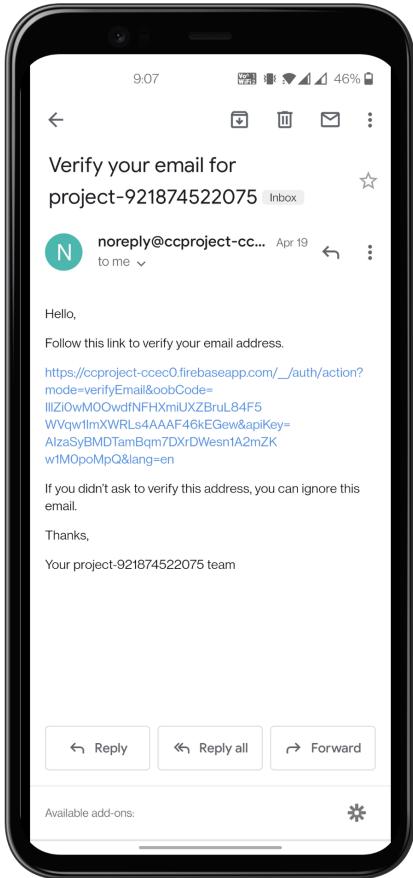


Fig 15: Verification link sent on email

The user initially needs to register on the app as shown in Fig , the second step includes the user account authentication which is done by sending a verification link to the users registered email id

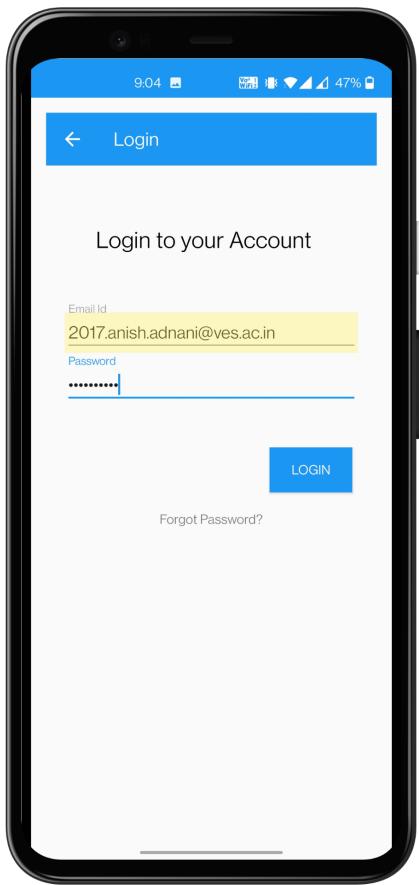


Fig 16: Login Page

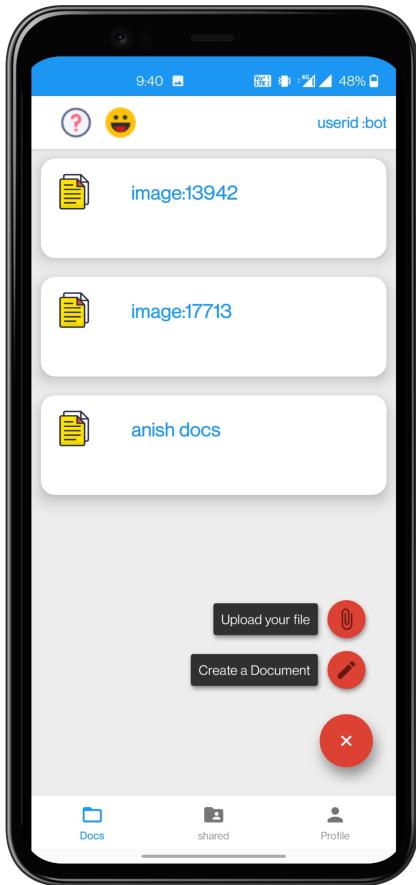


Fig 17: User Home Page

Users can then login into their accounts from the same credentials they used to register on the app. The email id must be verified before they login the app.

The next figure (Fig) shows the homepage of the app.

Where all the user created documents, uploaded files are visible.

Also users have an option to create new documents and to upload multimedia files

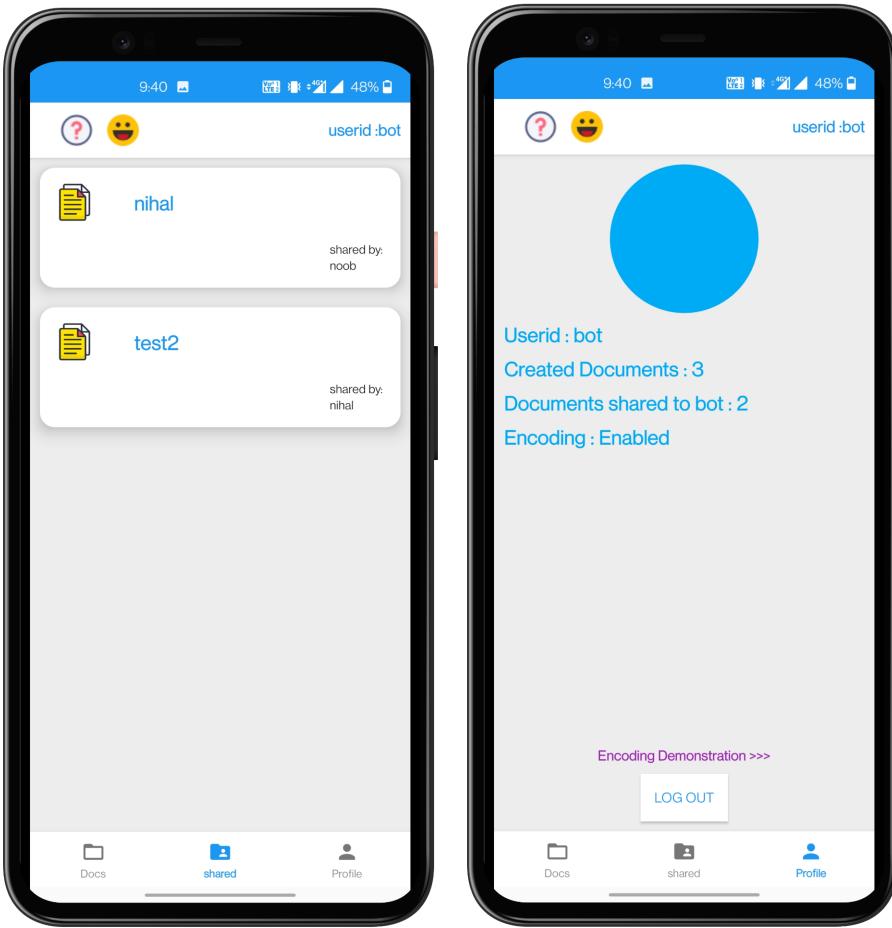


Fig 18: Shared Fragment

Fig 19: Profile Fragment

As seen from the Fig : the shared fragment displays all the files that have been shared to the user. It also includes details such as who created the file and the original file name. Users can view the shared file by clicking onto any particular file

Profile Fragment displays user details such as the user name, count of user created documents, count of documents uploaded by the user, and also a logout button in case the user needs to logout of the application.

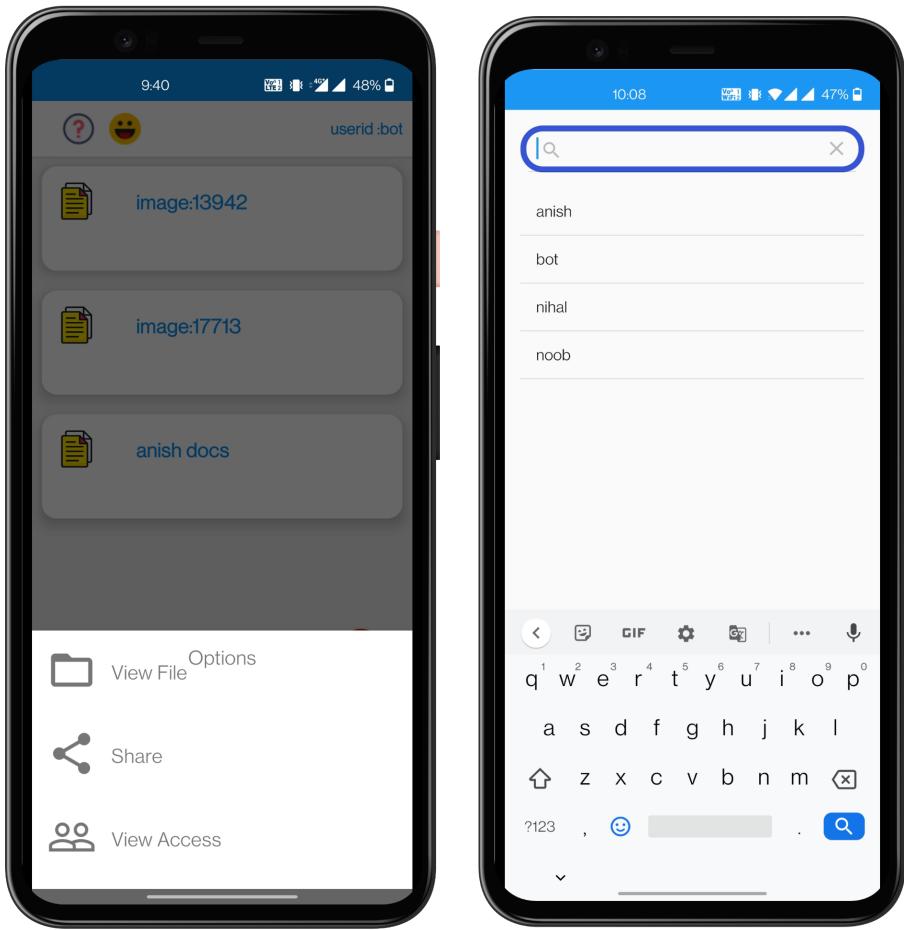


Fig 20: Available options

Fig 21: Sharing file to a particular user

There are various options available to user for a particular file namely View the file, Share file with other users, View who all has the file been shared with

Fig : shows that when a user wants to share a file, he needs to search the user by his user_name and add him in the access list.

Eg : “Anish” shares a file to “supriya” then “Anish” will search for “supriya” in the list and the file will appear in the shared section of “supriya”

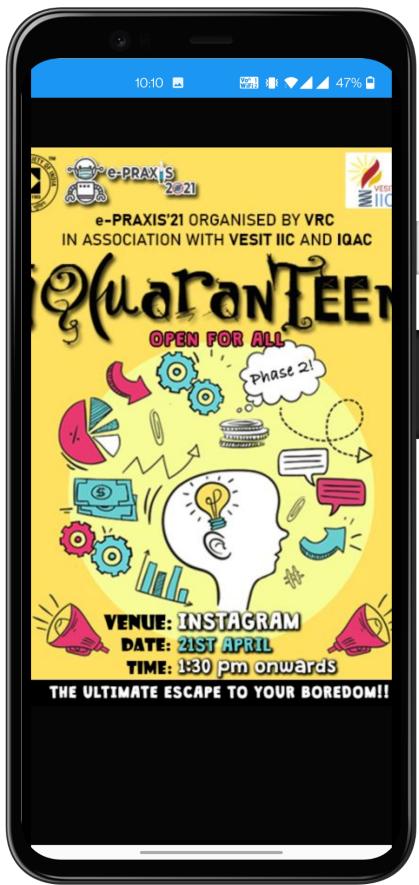


Fig 22: Viewing a file

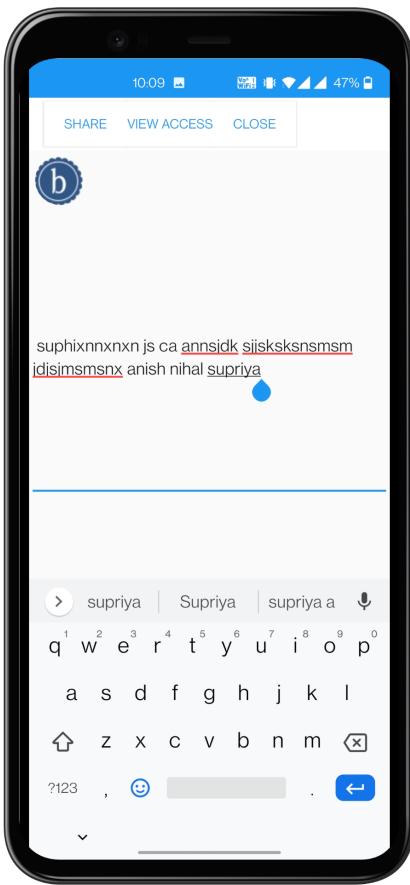


Fig 23: Viewing/ Editing a document

Fig : shows how a multimedia file is being viewed in the application

Fig : shows how a text document is being viewed and edited by the user. The top corner of the page also displays the logo of active users on the current viewed document.

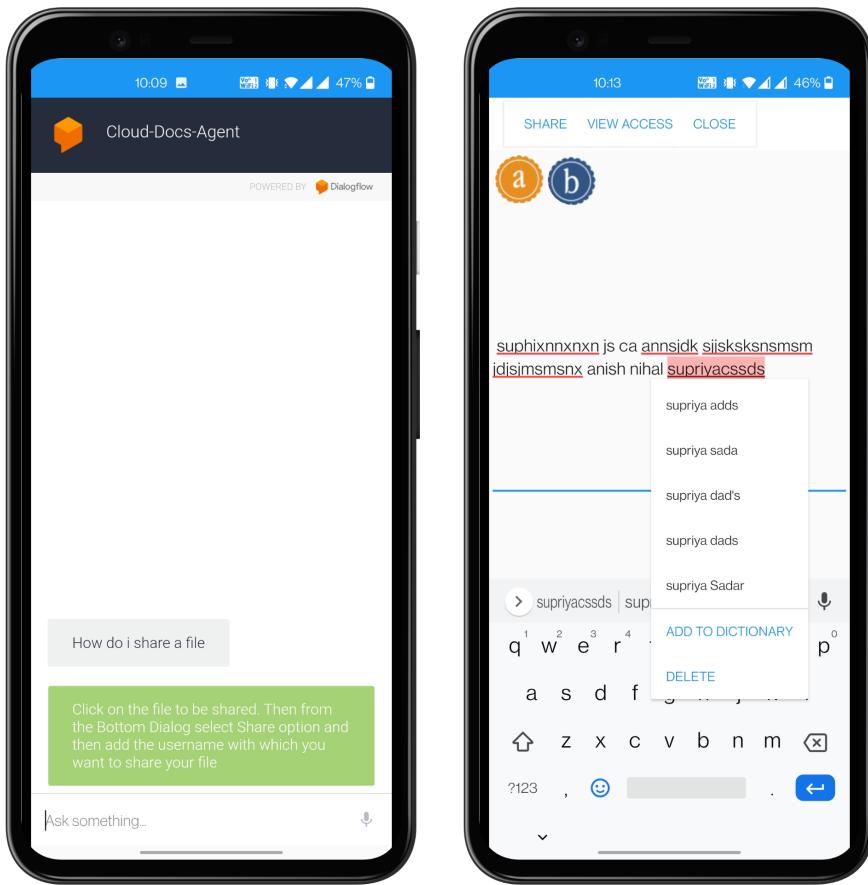


Fig 24: Chatbot support

The app is provided with an in-app chatbot support for general queries. Chatbot is developed using DialogFlow console and can be integrated in the app by hosting a website containing the iframe of the chatbot and embedding the website in the app using Webview. An alternative approach is to use Kommunicate SDK for dialog flow chatbot integration

Fig 25: Supports any online english assistance tool

Also the text documents can use any assistance tool like “Grammarly” to check for spelling and grammar errors.

The screenshot shows the Firebase Authentication section of the console. On the left, there's a sidebar with 'Project Overview' and various services like Build, Authentication, Firestore Database, etc. The main area is titled 'Authentication' and has tabs for 'Users', 'Sign-in method', 'Templates', and 'Usage'. Below is a table of registered users:

Identifier	Providers	Created	Signed in	User UID
n2bhy@protonmail.com	✉️	1 Apr 2021	11 Apr 2021	i0TakJ3YwjhmDVPz4d13FlxCTZq2
criston81782@gmail.com	✉️	31 Mar 2021	28 Apr 2021	osjwxXOV35aBBrpR9je27aiIMNz1
anishadnani00@gmail.com	✉️	24 Mar 2021	24 Mar 2021	qHUEmy0NkYQWuWL87hlaWPd1...
nihalbandari2010@gmail.c...	✉️	24 Mar 2021	19 Apr 2021	uRz9SfoZK8eXh4cZ8lHSoOAbJFG2
2017.anish.adnani@yes.ac.in	✉️	19 Apr 2021	28 Apr 2021	uVTvk1UxbRgYu3PEYgrdHiQ3XLb2

Fig 26: Displaying a list of users registered in the app

The above figure shows Firebase console with a list of registered users in the application

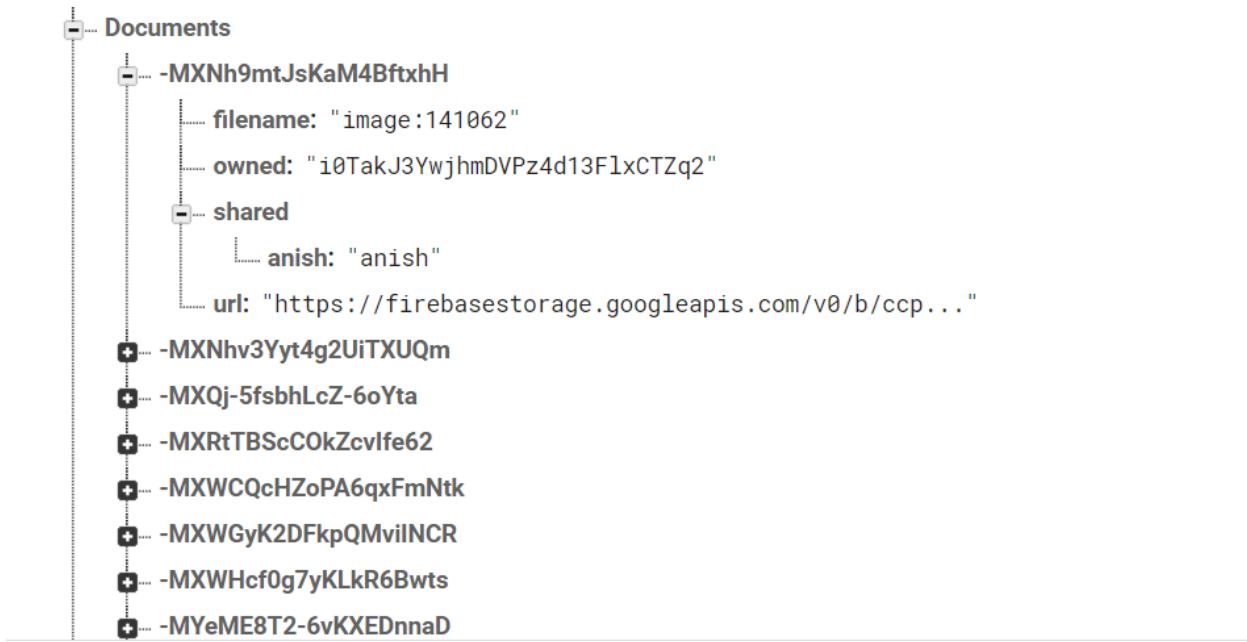


Fig 27: Documents Table Structure

As seen from the above figure each text document has a unique id.

Text documents have a specific filename, and owner and can be shared with multiple users



Fig 28: Showing structure of User Table

The User Table contains the user_name , search_name = user_name, device token, and list of owned files id, and a list of shared files id.

These file ids are then accessed from the documents table to get file information.

Name	Size	Type	Last modified
image:13942	52.28 KB	image/jpeg	4 Apr 2021
image:140512	134 KB	image/jpeg	3 Apr 2021
image:140632	141.18 KB	image/jpeg	3 Apr 2021
image:141062	57.55 KB	image/jpeg	3 Apr 2021
image:17713	60.3 KB	image/jpeg	19 Apr 2021
image:338725	163.34 KB	image/jpeg	4 Apr 2021

Fig 29: Firebase Database storage (saving multimedia files)

Above figure shows the multimedia files storage in Firebase Database storage.

Chapter 6

Conclusion

6.1 Limitations

- The app has been developed using android studio, hence is only compatible with android devices with no support to IOS users
- As we do not have a website developed, documents can only include texts, as documents that contain images do not have any specific positioning of the images (have random layout), that can be created on a website and universal links are then created to move between apps.

Eg: youtube.com link in a phone redirects to the youtube app rather than the website

- If a user is shared with a file with cannot provide app notifications as we are currently using free Spark plan for Firebase database, that does provide broadcasting notifications for free of cost

6.2 Conclusion

Our System works perfectly fine without having any downtime. Firebase (Realtime) Database used to store the data has no inconsistency and data is updated within 200 ms of the push operation. All multimedia files (images, videos, gifs, etc) are stored in Firebase Storage. This system can definitely aid the user in creating collaborative reports which are encrypted using hybrid cryptography. Files can be easily shared over the medium without the user having to set up any architecture for such an application. Hence our application “Collaboration tool in android” was successfully developed

6.3 Future Scope

Building a website for the same product will definitely help as a hosted website would allow us to create universal links, which would indirectly be useful to create documents that can contain images as well. Currently, with just an android application such a scenario is not possible.

Also providing open shareable links can help, currently, a user can only share data with other users, but cannot create shareable links through which anyone can view the data, such links can be easily created.

A notification would also be provided if a user is shared with a file, but this can only be done by upgrading the current Firebase plan.

Unique key encryption can be done, i.e, each user will have a specific key through which the data will be encrypted, hence further minimizing the success chances of data attacks.

Further, to enhance the user experience an in-app tutorial attached can prove to be great in supporting the new users.

Chapter 7

References

7.1 Research Papers Referred

- [1] Khawas, Chunnu & Shah, Pritam. (2018). Application of Firebase in Android App Development-A Study. International Journal of Computer Applications. 179. 49-53. 10.5120/ijca2018917200.
- [2] S. Wen and W. Dang, "Research on Base64 Encoding Algorithm and PHP Implementation," 2018 26th International Conference on Geoinformatics, 2018, pp. 1-5, doi: 10.1109/GEOINFORMATICS.2018.8557068.
- [3] M. A. Mokar, S. O. Fageeri and S. E. Fattoh, "Using Firebase Cloud Messaging to Control Mobile Applications," 2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), 2019, pp. 1-5, doi: 10.1109/ICCCEEE46830.2019.9071008.
- [4] "Data StructureComparisonBetween MySql Relational Database andFirebase Database NoSqlon Mobile Based Tourist Tracking Application", K G Sudiarta, N E Indrayana , W Suasnawa , S A Asri , Putu Wijaya Sunu, et al 2020 J. Phys.: Conf. Ser.1569 032092 .