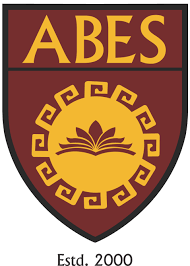
**MINI PRTOJECT REPORT**

**NOTE MAKING APPLICATION**



Name: VISHAL SINGH CHAUDHARY

University Roll No. 1900320100199

**SUBMITTED TO:**

**Department of Computer Science & Engineering (16 size)**

ABES engineering college

GHAZIABAD

Table of Contents

[ACKNOWLEDGEMENT iv](#_Toc56522468)

[DECLARATION v](#_Toc56522469)

[LIST OF FIGURES vi](#_Toc56522470)

[Chapter 1 1](#_Toc56522471)

[INTRODUCTION 1](#_Toc56522472)

[1.1 BACKGROUND 3](#_Toc56522473)

[1.2 SIMILAR APPLICATION 3](#_Toc56522474)

[1.2.1 EVERNOTE 4](#_Toc56522475)

[1.2.2 ONENOTE 5](#_Toc56522476)

[1.3 PROJECT GOALS 6](#_Toc56522477)

[Chapter 2 7](#_Toc56522478)

[SURVEY 7](#_Toc56522479)

[2.1 CRYPTOGRAPHY 7](#_Toc56522480)

[2.2 PASSWORD BASED ENCRYPTION/DECRYPTION 8](#_Toc56522481)

[Chapter 3 9](#_Toc56522482)

[TOOLS AND TECHNOLOGY USED 9](#_Toc56522483)

[3.1 PLATFORM CHOICE 9](#_Toc56522484)

[3.2 THE ANDROID ™ LIFE CYCLE 11](#_Toc56522485)

[3.3 CRYPTOGRAPHY 13](#_Toc56522486)

[3.3.1 TECHNIQUES USED FOR CRYPTOGRAPHY 13](#_Toc56522487)

[3.3.2 TYPES OF CRYPTOGRAPHY 13](#_Toc56522488)

[3.4 FIREBASE 14](#_Toc56522489)

[3.5 CLOUD FIRESTORE 15](#_Toc56522490)

[Chapter 4 16](#_Toc56522491)

[SNAPSHOT 16](#_Toc56522492)

[4.1 SPLASH SCREEN 16](#_Toc56522493)

[4.2 REGISTER ACTIVITY 17](#_Toc56522494)

[4.3 LOG IN ACTIVITY 18](#_Toc56522495)

[4.4 MAIN ACTIVITY 19](#_Toc56522496)

[4.5 ADD NOTE 20](#_Toc56522497)

[4.6 CLOUD FIRESTORE 21](#_Toc56522498)

[4.7 EDIT NOTE 22](#_Toc56522499)

[4.8 NOTE CONTENT AND NAVIGATION 23](#_Toc56522500)

[4.9 SHARE NOTE 24](#_Toc56522501)

[Chapter 5 26](#_Toc56522502)

[RESULTS AND DISCUSSIONS 26](#_Toc56522503)

[Chapter 6 27](#_Toc56522504)

[CONCLUSION AND FUTURE WORK 27](#_Toc56522505)

[Chapter 7 28](#_Toc56522506)

[REFRENCES 28](#_Toc56522507)

# ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my Parents as well as our teacher (Dr. Santosh Kumar ) who gave me the golden opportunity to do this wonderful project on Note Making Application , which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them.  
Secondly I would also like to thank my friends who helped me a lot in finalizing this project within the limited time frame.

# DECLARATION

I hereby declare that the Report entitled ("NOTE MAKING APPLICATION") is an authentic record of my own work as requirements of Mini Project during the period from 15/05/2020 to 16/11/2020 for the award of degree of B.Tech. (Computer Science & Engineering), ABES Engineering College, Ghaziabad, under the guidance of (Name of Project Guide).

**VISHAL SINGH CHAUDHARY**

**1900320100199**

**Date: 18/11/2020**

# LIST OF FIGURES

Figure 1.1 It shows total worldwide cellular consumption

Figure 2.1 Evernote Phone app

Figure 2.2 OneNote Phone app

Figure 3.1 No of Android User in the world

Figure 3.2 it shows the android activity lifecycle

# Chapter 1

# INTRODUCTION

There are 5.20 billion unique mobilephoneusersin the world today, according to the latest data from GSMA Intelligence. The total number of unique mobile users around the world grew by 102 million in the past 12 months. Unique mobile users are currently growing at a rate of 2 percent per year. People now more than ever are looking to access the Internet, take photographs, record videos, write notes, and much more while on-the-go, making mobile applications for smartphones increasingly prevalent in today’s society. Many popular websites today either have mobile-friendly websites and/or specifically have created applications for mobile users, further adding use to mobile devices.

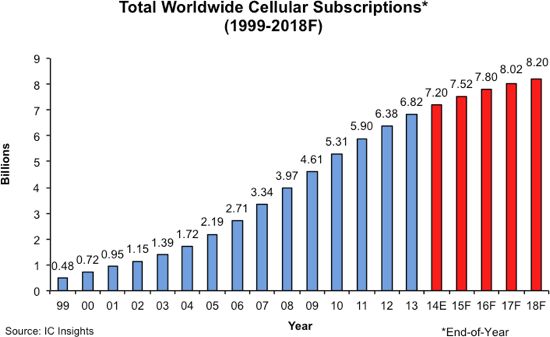


Figure 1.1: This graphic shows the total worldwide cellular subscriptions in every year

Smartphones serve as an ‘all-in-one’ device, offering the functionality of a camera, journal, notebook, and audio recorder all in one unit. Individuals who travel can use smartphones to make notes, take photographs, record videos, and record sound clips of the world around them. This additional functionality provided by smartphones is convenient for travellers, as it eliminates the need to carry around several different devices.

With the increase in the no of mobile device and the mobile user, our Note Making Application will Simplify the life’s of many mobile users, with the help of our application user can store small notes , with title and description , with the current date , which help the user to remember what all work they have to complete and what all work is pending , after completing the work user can delete note and then are UI will show the remaining note , our app use the cloud based service, so user can access their notes from anywhere or from any phone.

For the security purpose of the user notes , we encrypt the user notes with the best encryption algorithm , i.e. AES(Advanced Encryption Standard), so nobody other than user can read his notes

## 1.1 BACKGROUND

Prior to beginning development, we researched existing note making oriented applications and the processes and standards of Android™ programming. This section of our paper outlines our findings regarding similar applications and major competitors, and then continues on to explain our choice of the Android™ platform. After choosing the Android™ platform, I conducted more research into the Android™ lifecycle. Then I research various cryptography algorithm available, and how these algorithm work and then I also research which algorithm will be best with respect to my application

## 1.2 SIMILAR APPLICATION

Even though there are many solutions that exist in the current mobile market that try to address the aforementioned problems, none of them are tailored specifically for use in academia, rather they are more general-purpose applications that target a broader market, trying to encapsulate as many customers as possible. This section will describe existing technologies that have been researched, tried and tested and continue to illustrate the gap in the market for academic note taking.

### 1.2.1 EVERNOTE

Evernote is a multi-platform note taking system which supports a wide variety of platforms, ranging from iOS for iPhones and iPads to Android, Windows Phone and Web, with support even for PCs on both Mac and Windows [1].

For the setting of a meeting, tablets are most appropriate, so in my research of this piece of software, I focused on the iOS version through the iPad. Evernote provides an eloquent way of organising notes in the form of ‘notebooks’[1] which is very similar to storing notes in physical folders, separating the notes by category and purpose. Checklists can be created and notes can be shared with peers. Although checklists do indirectly provide a way to check progress they do not allow for action to be delegated to individuals whilst seamlessly making alerting them through the system. Additionally, although sharing of notes may be required in certain places it is not always desirable to share notes in their entirety as there isn’t always a need for complete transparency with all involved parties.

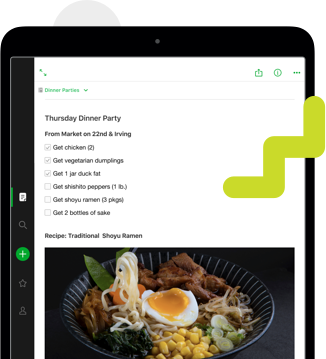


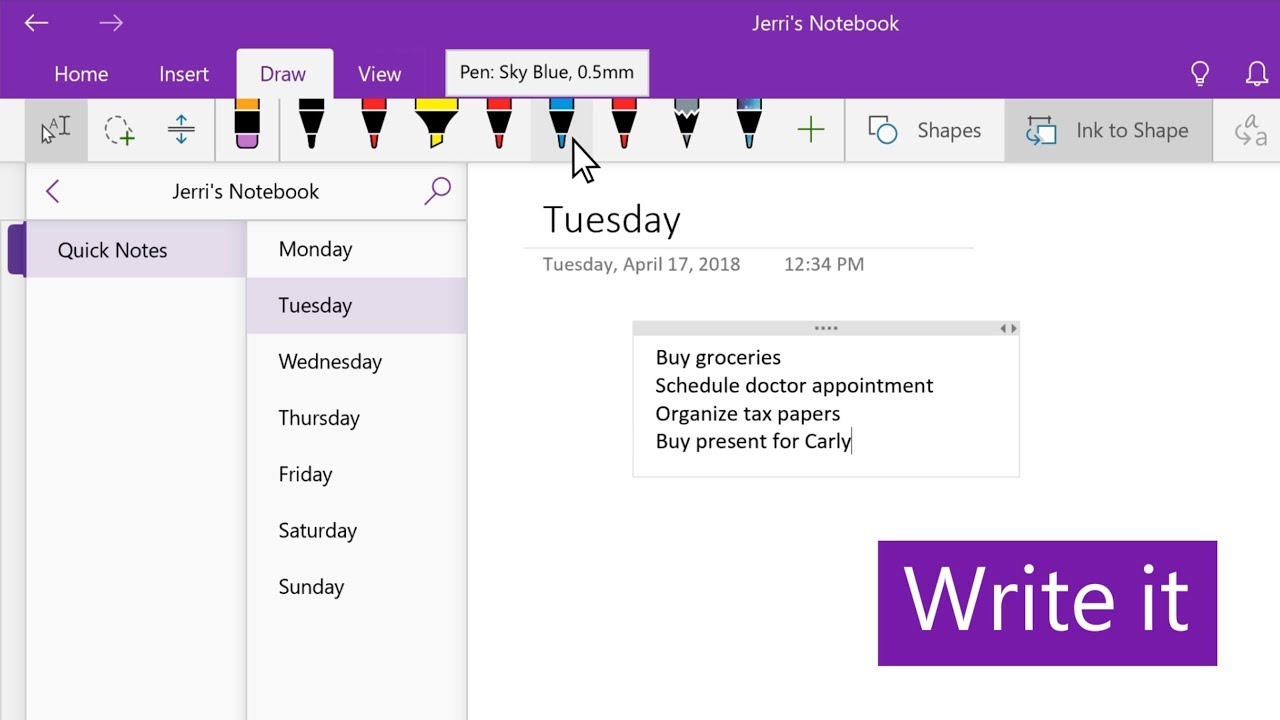
Figure 1.2: Evernote Phone app Example

### 1.2.2 ONENOTE

Similar to Evernote, Microsoft’s OneNote is available on a wide variety of platforms, but for continuity we will focus here on their iOS version of the application [2].

OneNote provides a slightly different method of organising notes, through the introduction of sections in addition to having notebooks. This is very useful for a continual documentation where everything is still relevant each time that the notes are accessed, however this is not the case with academic meetings, where we usually are only interested in the latest, most recent updates and notes on previous meetings become a distraction more than anything when it comes to viewing then notes again.

.



**Figure 1.3 : OneNote App Example**

## 1.3 PROJECT GOALS

The main objectives of the project is to create an application that is suitable for use by all the user to take some important notes anywhere from any device . This must be reflected in the design of the system so that different types of users have different privileges. It is of the upmost importance that real value is produced and delivered as a means of driving the project and is essential to develop a deeper understand of the mobile application development process. Security of users notes is also the major priority of author

The whole concept of mobile development and cryptography was completely alien to the author from the beginning of this project; therefore, a lot of focus and time has been exerted into learning these foreign concepts. This was another core objective of this project and these concepts were all self-taught through various different resources.

The end product was aimed to provide an efficient, alternative solution to the problem domain and was intended to improve the user experience with the security of their notes with best cryptography algorithm [3]

# 

# Chapter 2

# SURVEY

# 2.1 CRYPTOGRAPHY

DNA cryptography is a new born cryptographic field emerged with the research of DNA computing, in which DNA is used as information carrier and the modern biological technology is used as implementation tool. The vast parallelism and extraordinary information density inherent in DNA molecules are explored for cryptographic purposes such as encryption, authentication, signature, and so on. In this paper, we briefly introduce the biological background of DNA cryptography and the principle of DNA computing, summarize the progress of DNA cryptographic research and several key problems, discuss the trend of DNA cryptography, and compare the status, security and application fields of DNA cryptography with those of traditional cryptography and quantum cryptography. It is pointed out that all the three kinds of cryptography have their own advantages and disadvantages and complement each other in future practical application. The current main difficulties of DNA cryptography are the absence of effective secure theory and simple realizable method. The main goal of the research of DNA cryptography is exploring characteristics of DNA molecule and reaction, establishing corresponding theories, discovering possible development directions, searching for simple methods of realizing DNA cryptography, and laying the basis for future development.[4][5][6][7][8][9]

## 2.2 PASSWORD BASED ENCRYPTION/DECRYPTION

This paper develops a theory of multi-instance (mi) security and applies it to provide the first proof-based support for the classical practice of salting in password-based cryptography. Mi-security comes into play in settings (like password-based cryptography) where it is computationally feasible to compromise a single instance, and provides a second line of defence, aiming to ensure (in the case of passwords, via salting) that the effort to compromise all of some large number *m* of instances grows linearly with *m*. The first challenge is definitions, where we suggest LORX-security as a good metric for mi security of encryption and support this claim by showing it implies other natural metrics, illustrating in the process that even lifting simple results from the si setting to the mi one calls for new techniques. Next, we provide a composition-based framework to transfer standard single-instance (si) security to mi-security with the aid of a key-derivation function. Analysing password-based KDFs from the PKCS#5 standard to show that they meet our in differentiability-style mi-security definition for KDFs, we are able to conclude with the first proof that per password salts amplify mi-security as hoped in practice. We believe that mi-security is of interest in other domains and that this work provides the foundation for its further theoretical development and practical application.[10][11][12][13][14]

# Chapter 3

# TOOLS AND TECHNOLOGY USED

## 3.1 PLATFORM CHOICE

There is a wide variety of different tablets in the market, each of which run a specific platform. The main platforms used in this market are Android, iOS, Windows. Two main ways exist to create applications on any of these platforms. Web based applications, and native applications. Native applications bring the principal advantage that the user experience is enhanced and optimised as the application will operate quickly and seamlessly with the platform as it has been designed specifically for it. Not going for a web based application does however mean that if the application needed cross platform operability we would need to design an application (relatively) from scratch for each platform. However, cross platform operability is outside the scope of this project so it was decided that creating a native application was the best course of action for this project.[15]

Although Android devices hold the bigger market share in the tablet sector, there are a lot of different Android devices which create a lack if consistency between device usability. An analysis that attempts at gauging the user experience of tablets compared different tablets across all platforms on a number of different criteria, it was found that the Apple iPad which runs on the iOS platform had the least cognitive load, meaning users could adopt the device whilst having little or no knowledge of the works or usage patterns of the device. This in addition to the iPad being the clients preferred device of usage gave the iOS platform a boost in the race for platform choice. Figure 2.1: Cognitive Load Index of different tablets in the market.[28] Windows was eliminated as the deployment platform at this stage as the resources to deploy onto a physical device were not available at the time. Since this project is based around interacting with the user through an interface, an integrated development environment was a must. Both Android and iOS provide development studios, Anroid Studio and XCode respectively. After experimenting with both IDEs, I found that XCode offered easier integration with the front-end design and back-end than Android Studio, a feature that especially stood out for me was ‘Auto Layout’ which allows for constraints to be set on all views in a view

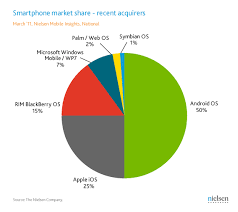


Figure 3.1 data shows android phone user

hierarchy and dynamically calculates size and positioning of objects in accordance with the orientation, state and screen size of the iOS device. This is done for both internal changes within the application and external changes that can affect the device view.

## 3.2 THE ANDROID ™ LIFE CYCLE

The Android™ life cycle is a series of stages that all Android™ activities go through at different points when being using on a device. An Android™ activity is essentially a window that the user can interact with. The stages an activity goes through are accompanied by a series of methods that are called at each stage in the life cycle. The 11 different stages and methods that are called to transition from one stage to another are displayed in the figure below

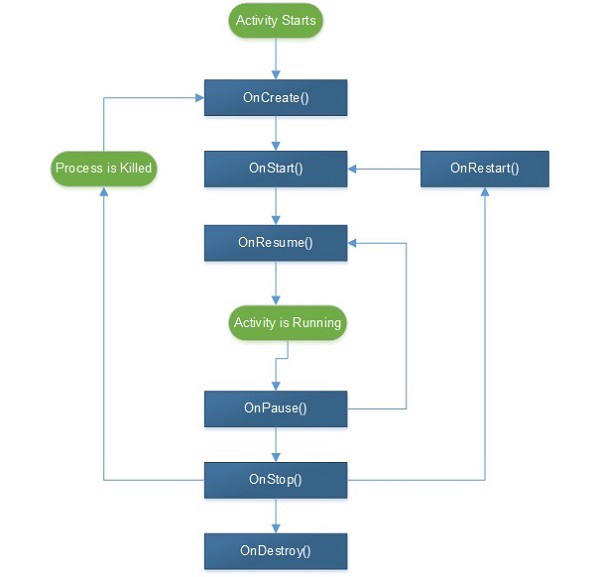


Figure 3.2 this figure shows android Activity lifecycle

At the very beginning of an app’s life cycle, the onCreate() method is called when the application is being opened. The body of this method is where some of the initial onetime things get done like creating or linking to a database. After onCreate() is the onStart() method. The onStart() method is called before the application becomes visible, so a user interface should be created during this method call. After onStart() is the onResume() method which mainly recreates any saved states that the user may have been in beforehand. At this point in the application lifecycle, the application is fully visible and the user can freely interact with it.

While the activity is visible to the user, there are actions the user can take to make the activity not be visible anymore. If the user does something that causes another activity to be displayed in front of the current activity, then the onPause() method will be called on the activity that is moving into the background. Since the activity is losing focus, pieces of information necessary to save state should be saved, like form entries or unsaved draft emails. If the user goes back into the application, the onResume() method would get called to restore the saved state. On the other hand, if the user decides to do something like switch applications or go to the home screen, then the onStop() method will be called on the current activity. At this point, the application may never come back into the foreground, so all states should be saved here, and all resources that are not necessary should be released.

In addition to the user, the Android™ OS can also affect the activity life cycle. One situation where this is the case is if the OS ever runs low on memory and needs to free some up. it may call onDestroy() on the application which would completely close the application and its activities. Everything should be closed and released during the onDestroy() method call. If the onDestroy() method is not called on the application, and the user goes back into the application, the onRestart() method will get called followed by the onStart() method, then the onResume() method. The only new step here is the onResume() method which will handle things that may need to be restored from a previous state as a result of calling onStop()

## 3.3 CRYPTOGRAPHY

cryptography is technique of securing information and communications through use of codes so that only those person for whom the information is intended can understand it and process it. Thus preventing unauthorized access to information. The prefix “crypt” means “hidden” and suffix graphy means “writing”.In Cryptography the techniques which are use to protect information are obtained from mathematical concepts and a set of rule based calculations known as algorithms to convert messages in ways that make it hard to decode it. These algorithms are used for cryptographic key generation, digital signing, verification to protect data privacy, web browsing on internet and to protect confidential transactions such as credit card and debit card transactions

### 3.3.1 TECHNIQUES USED FOR CRYPTOGRAPHY

In today’s age of computers cryptography is often associated with the process where an ordinary plain text is converted to cipher text which is the text made such that intended receiver of the text can only decode it and hence this process is known as encryption. The process of conversion of cipher text to plain text this is known as decryption.

### 3.3.2 TYPES OF CRYPTOGRAPHY

1. **Symmetric Key Cryptography:**

It is an encryption system where the sender and receiver of message use a single common key to encrypt and decrypt messages. Symmetric Key Systems are faster and simpler but the problem is that sender and receiver have to somehow exchange key in a secure manner. The most popular symmetric key cryptography system is Data Encryption System(DES).

1. **Hash Functions:**

There is no usage of any key in this algorithm. A hash value with fixed length is calculated as per the plain text which makes it impossible for contents of plain text to be recovered. Many operating systems use hash functions to encrypt passwords.

1. **Asymmetric Key Cryptography:**

Under this system a pair of keys is used to encrypt and decrypt information. A public key is used for encryption and a private key is used for decryption. Public key and Private Key are different. Even if the public key is known by everyone the intended receiver can only decode it because he alone knows the private key.

## 3.4 FIREBASE



1. Firebase is a BAaS from Google.
2. It can be treated as a Balck-Box of code which takes input and gives a desired output after some computation.
3. The catch is the fact that we don’t know how these computations are done .
4. That’s the role of a BAaS to provide various Backend related services .
5. Firebase provides alot of services including

* Authentication
* Hosting
* Real time Database
* Cloud FireStore
* Machine learning
* Data Analysis

## 3.5 CLOUD FIRESTORE



1. Cloud Fire store is in built database of Firebase.
2. It is a NoSQL database
3. It stores data in collections and document
4. The main format in which it stores data includes arrays and JSON
5. A JSON document stores data in key-value pairs where both the key and value are to be enclosed in double quotes all the time.
6. Unlike SQL databases there are no joins but they do have ***populate*** feature which helps to create a reference to a document in some other collection using its object ids

.

# Chapter 4

# SNAPSHOT

## 4.1 SPLASH SCREEN

When the user open the app , splash screen will be shown for 2sec, then Login

Activity is opened , if user is not Logged In, Otherwise MainActivity is

Closed

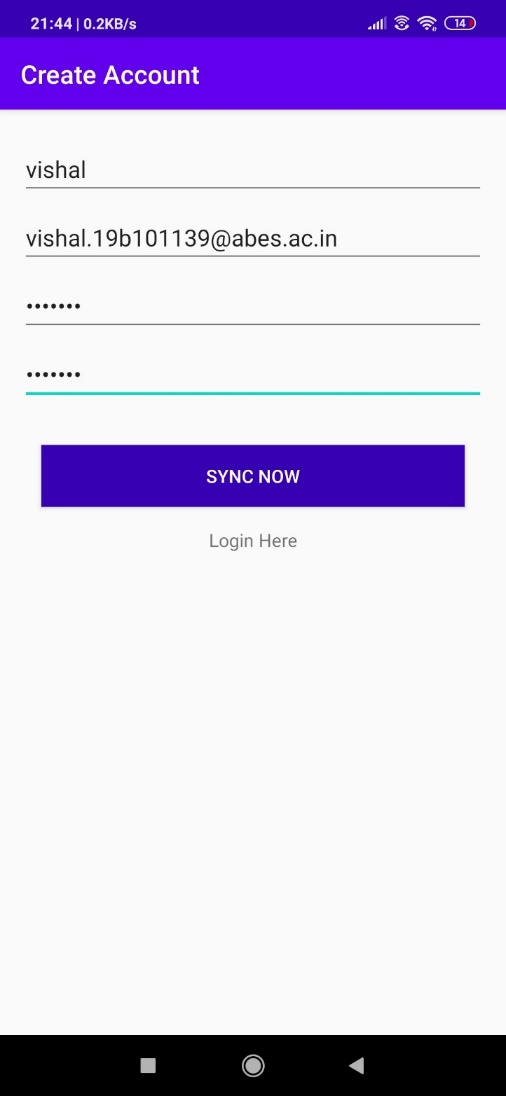


## 4.2 REGISTER ACTIVITY

If user is not Registered in our app, then through this activity user will register

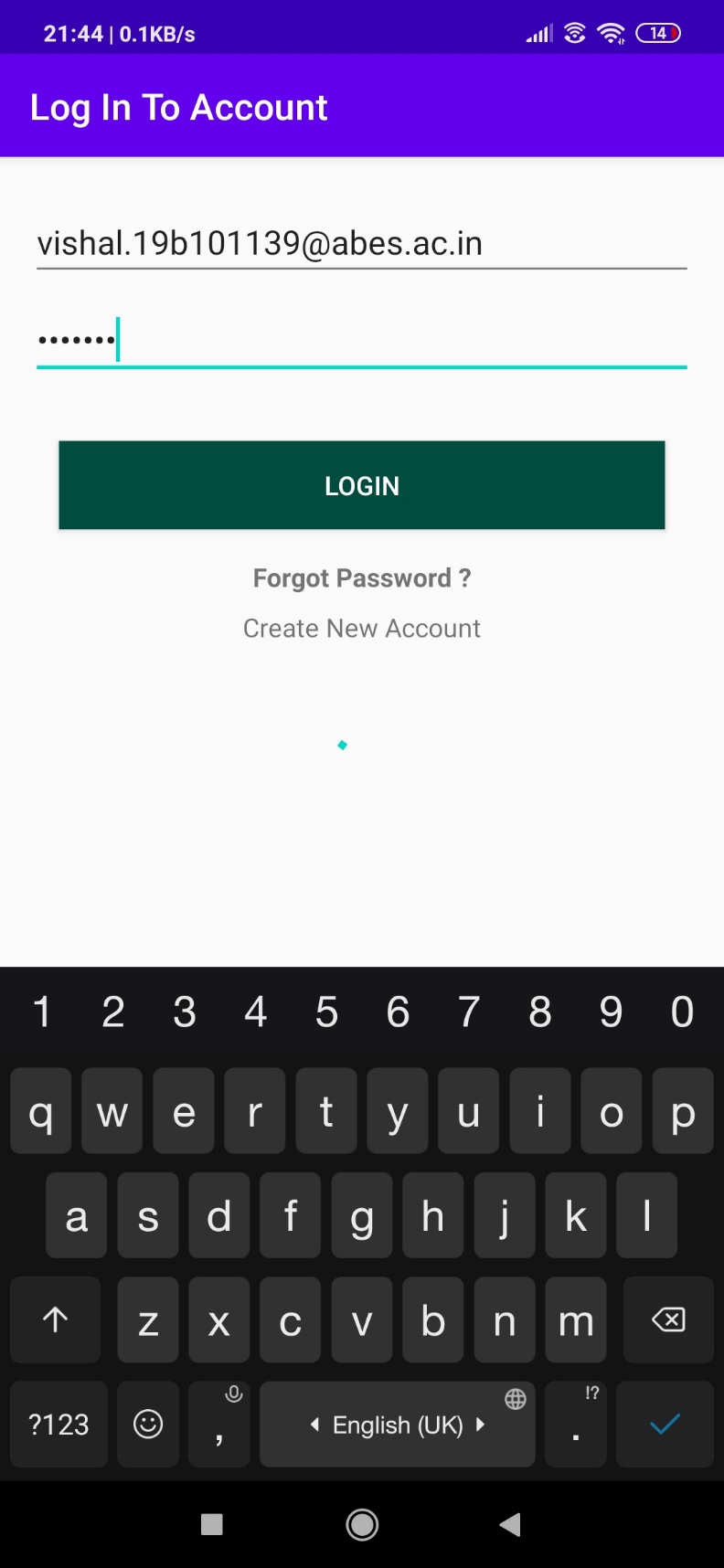
In our app, and after registering in our app , Log In Activity is Opened to Log

In in our App



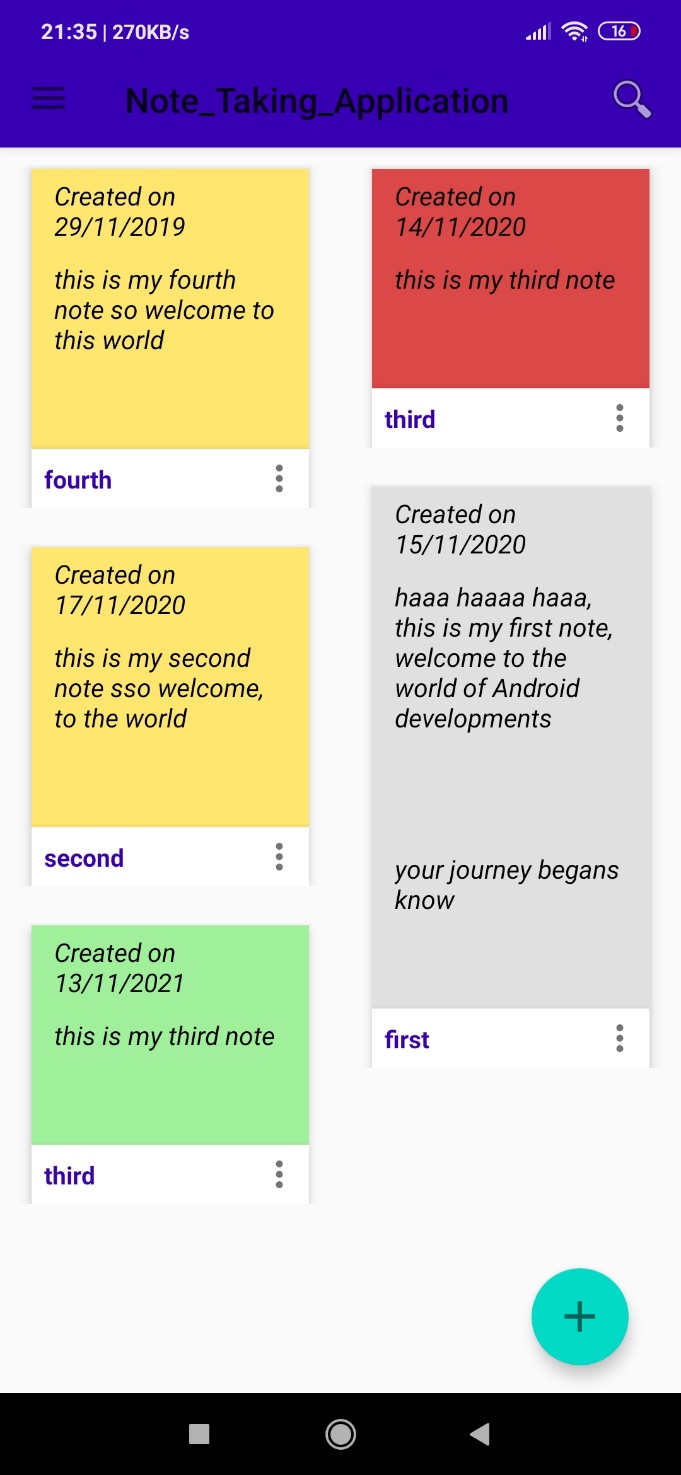
## 4.3 LOG IN ACTIVITY

After User is Registered , user will move to Login Activity and Log In in our app to save and edit note.



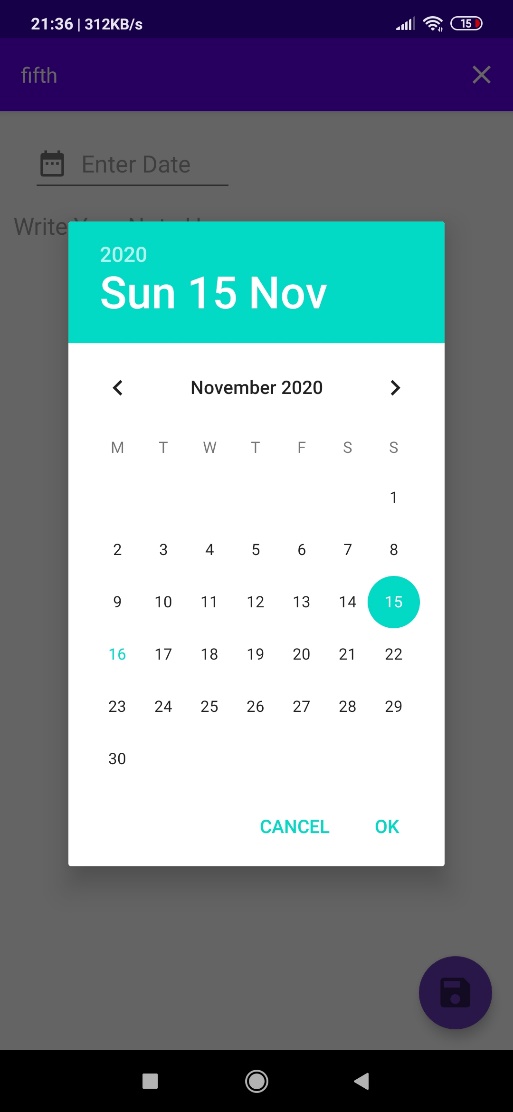
## 4.4 MAIN ACTIVITY

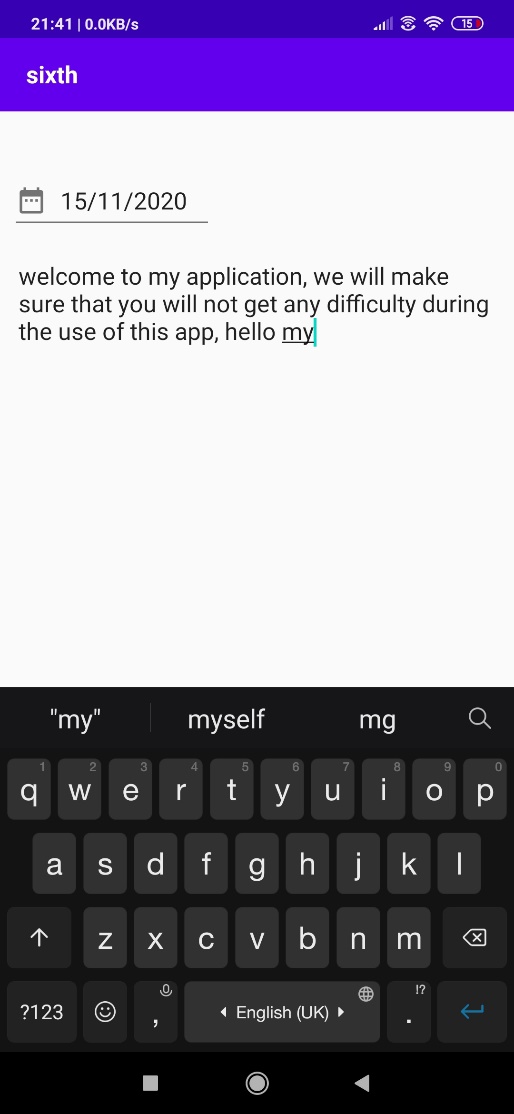
This is the main Activity in our where all the notes of the user is shown , from this activity user can navigate to every part of the app, i.e user can add, edit, read and delete the notes



## 4.5 ADD NOTE

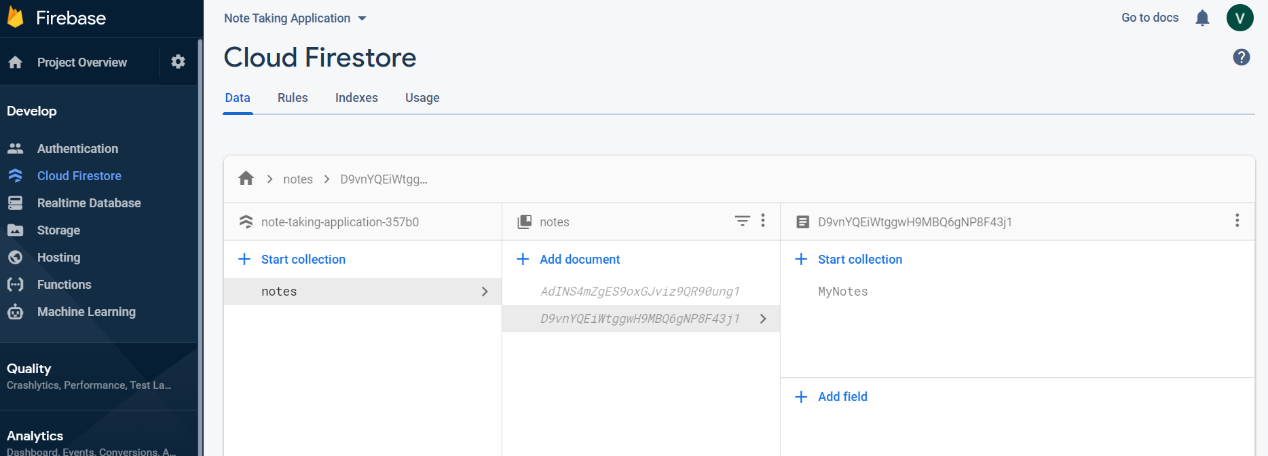
In this , Activity user can add note , with content , title and date, notes which are stored in cloud firestore are encrypted.



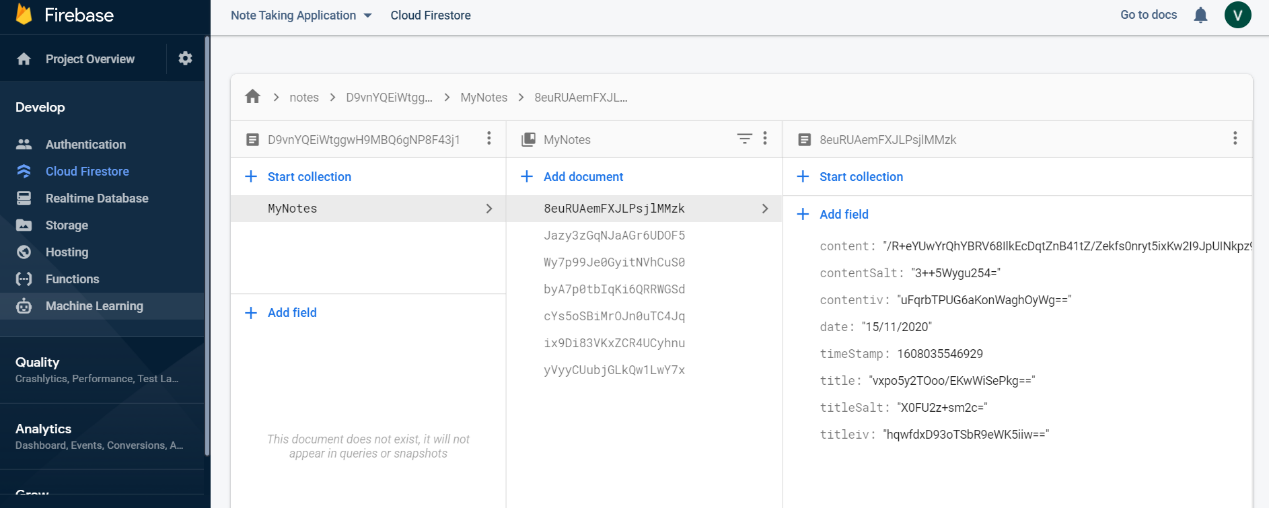


## 4.6 CLOUD FIRESTORE

The note added by user is saved in cloud firestore in Encrypted form so that no one other than user can understand what is written in notes, i.e Notes saved in cloud firestore are in encrypted form.

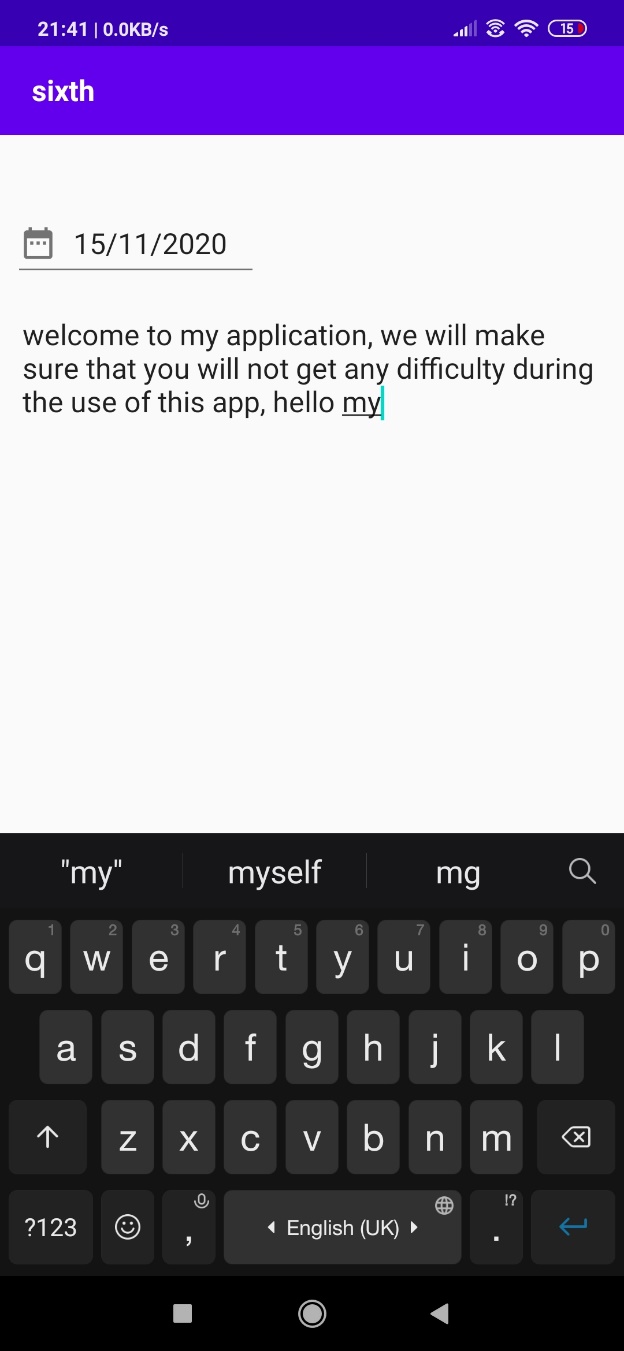


Data of user is encrypted



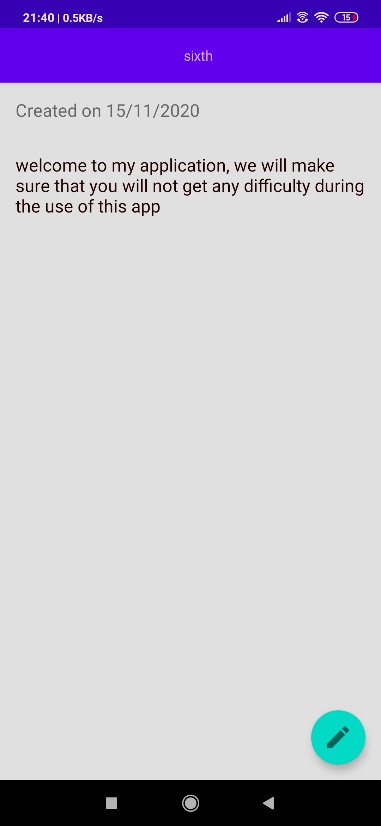
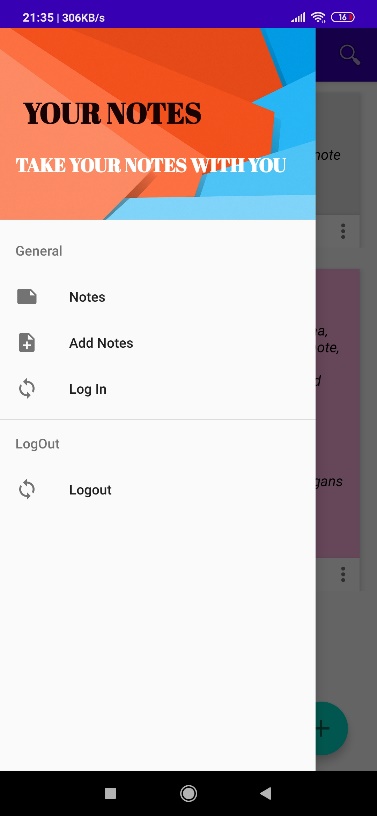
## 4.7 EDIT NOTE

In this Activity user can edit note, i.e user can change the content of note an d save the note in its account in an encrypted form



## 4.8 NOTE CONTENT AND NAVIGATION

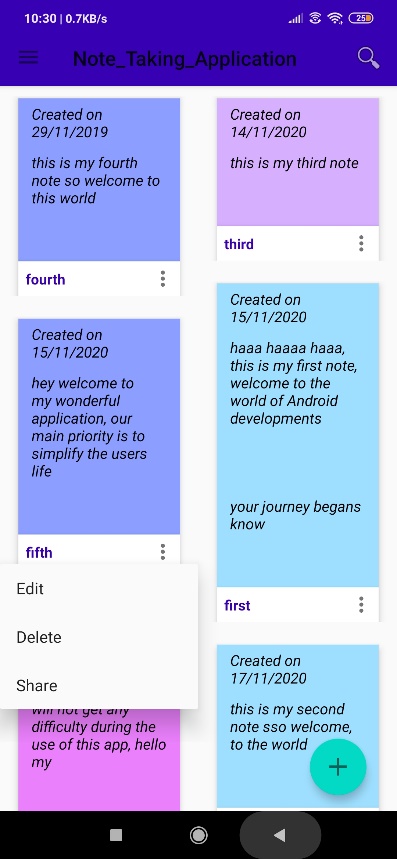
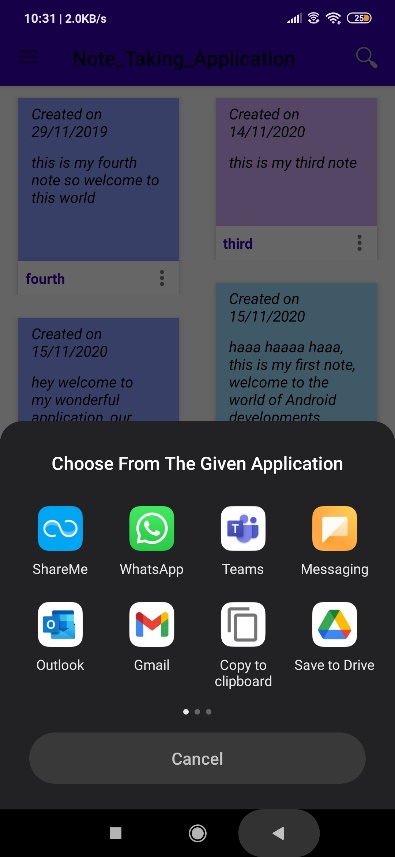
In note Content Activity user can see its notes in a better manner and through navigation view user can Log Out and Add Notes

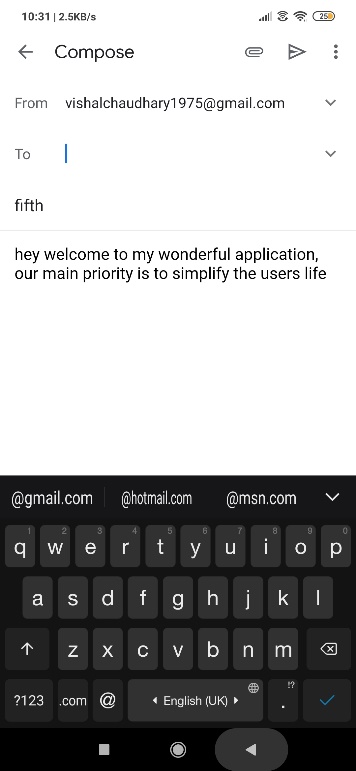
 

## 

## 4.9 SHARE NOTE

With the help of this feature user can share its note with anyone through any social interacting sites



# Chapter 5

# RESULTS AND DISCUSSIONS

The Result of this project is that user can store their notes safely with the help of our application, user can also edit and delete the notes safely, user can access its notes from any part of the world and from any android phone

# Chapter 6

# CONCLUSION AND FUTURE WORK

Future development for this project might involve further improving the application itself and creating a website that would work alongside the application.

There are many feature I can add in this app such as allowing user to add audio, video, and images to notes, it will increase the flexibility of the app , there is a huge scope of improving the UI of the app for improving the experience of user will using my app

I can also make website that work along with this app, so that user can access the notes through website also

# Chapter 7

# REFRENCES

1. Evernote [Internet]. 2020 [cited 15 November 2020]. Available from: <https://evernote.com>
2. Microsoft OneNote | The digital note-taking app for your devices [Internet]. Onenote.com. 2020 [cited 15 November 2020]. Available from: <https://www.microsoft.com/en-in/microsoft-365/onenote/digital-note-taking-app?ms.url=onenotecom&rtc=1>
3. Various Encryption/Decryption Algorithm: [best Algorithm] <https://acodez.in/data-encryption-algorithms/>
4. Watson J D, Hopkins N H, Roberts J W, et al. Molecular Biology of the Gene. 4th ed. Menlo Park, CA: The Benjamin/Cummings Publishing Co., Inc., 1987
5. Seeman N C. Nanotechnology and the double helix. Scientific American, 2004, 290: 34–43
6. Li Debao, Xu Ping. Theory and Methods of Recombinant DNA. Hangzhou: Zhejiang Science and Technology Publishing Co., 1994
7. Fodor S P, Read J L, Pirrung M C, et al. Light-directed, spatially addressable parallel chemical synthesis. Science, 1991, 251: 767–773
8. Pease A C, Solas D, Sullivan E J, et al. Light-generated oligonucleotide arrays for rapid DNA sequence analysis. Proc Natl Acad Sci USA, 1994, 91: 5022–5026
9. Schena M, Shalon D, Ronald W, et al. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. Science, 1995, 270: 467–470
10. Abadi, M., Warinschi, B.: Password-Based Encryption Analyzed. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 664–676. Springer, Heidelberg (2005)
11. Baudron, O., Pointcheval, D., Stern, J.: Extended Notions of Security for Multicast Public Key Cryptosystems. In: Montanari, U., Rolim, J.D.P., Welzl, E. (eds.) ICALP 2000. LNCS, vol. 1853, pp. 499–511. Springer, Heidelberg (2000)
12. Bellare, M., Boldyreva, A., Micali, S.: Public-Key Encryption in a Multi-user Setting: Security Proofs and Improvements. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 259–274. Springer, Heidelberg (2000)
13. Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A concrete security treatment of symmetric encryption. In: 38th FOCS, pp. 394–403. IEEE Computer Society Press (October 1997)
14. Bellare, M., Pointcheval, D., Rogaway, P.: Authenticated Key Exchange Secure against Dictionary Attacks. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 139–155. Springer, Heidelberg (2000)
15. Bolton D, Hall S, Nwabuoku A. A Scoring Comparison of Android and iOS Development - The New Stack [Internet]. The New Stack. 2015 [cited 23 March 2016]. Available from: http://thenewstack.io/scoring-comparison-android-ios-development/