

Visvesvaraya Technological University Belagavi, Karnataka-590 018



A MINI PROJECT REPORT On

NOTE DOWN APPLICATION

Submitted
in partial fulfilment requirements for the credit of the Course on
MOBILE APPLICATION DEVELOPMENT
18CSMP68
by

P PADMAPRASAD SHENOY 4CB19CS064
S SREENIVASA SHENOY 4CB19CS087

Under the Guidance of
Mr. Janardhana Bhat K
Assistant Professor



Department of Computer Science and Engineering
Canara Engineering College, Benjanapadavu 2021-2022

CANARA ENGINEERING COLLEGE

BENJANAPADAVU, BANTWAL- 574219 – KARNATAKA

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING



CERTIFICATE

This is to certify that **P PADMAPRASAD SHENOY(4CB19CS064)** and **S SREENIVASA SHENOY(4CB19CS087)** have successfully completed the mini project work on ‘**Note Down Application**’ and submitted in partial fulfillment of the requirements of the Course on **Mobile Application Development (18CSMP68)** prescribed by the **Visvesvaraya Technological University** during the academic year **2021-2022**. It is verified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The project report has been approved as it satisfies the academic requirements in respect of mini project work prescribed by Bachelor of Engineering Degree.

Course Instructor
Mr. Janardhana Bhat K
Assistant Professor

H.O.D
Dr. Demian Antony D’ Mello
Professor

Name of the Examiners

Signature with Date

1.

2.

ACKNOWLEDGEMENT

We are indebted to our **Principal, Dr. Ganesh V. Bhat** and management of Canara Engineering College for providing an environment with all facilities that helped us in completing our mini project.

We are extremely grateful to **Dr. Demian Antony D' Mello, Head of Computer Science & Engineering Department** for his moral support and encouragement.

We wish to express our sincere gratitude to our course instructor, **Mr. Janardhana Bhat K, Assistant Professor, Computer Science & Engineering Department**, for the guidance and suggestions.

We thank all the teaching and non-teaching staff of the department for their kind help in completing this mini project.

Last but not the least; we would like to add some personal note. If there is a driving force that kept us going, and what has not changed it is the constant support and blessing of our parents, family and friends. There is no doubt, in spite of my strenuous efforts error might remain in mini project. Naturally, we alone take full responsibility for any lack of clarity, occasional erratum or inexactness that may occur.

P PADMAPRASAD SHENOY 4CB19CS064

S SREENIVASA SHENOY 4CB19CS087

ABSTRACT

In a busy schedule, we tend to forget many important things easily, and to remember these things we need to note down things in a piece of paper. In such busy schedules people need some personal assistant or a reminder to remind them about the important work that needs to be done. This android application will help them to remind, to do such important things. These sticky notes android application can help us to note the daily task which needs to be done. Important meetings, events etc. can be recorded with great ease through the use of this application. This application will allow the users to organize the data in a simpler and easy way. In this application users can view all the Notes, likewise also can manage old and new notes. These sticky notes can also be edited and deleted.

TABLE OF CONTENTS

CONTENTS	PAGE NO.
1. INTRODUCTION	
1.1 Introduction to Android Application	01
1.2 Introduction to Android Studio	02
1.3 Mini Project Description	03
2. REQUIREMENT SPECIFICATION	
2.1 Hardware Requirements	04
2.2 Software Requirements	04
3. DESIGN	
3.1 Architectural Diagram/Flowchart	05
3.2 XML Code	06
3.3 Database Schema diagram	29
4. IMPLEMENTATION DETAILS	
4.1 List of classes imported.	30
4.2 List of important library functions used.	32
4.3 Java Code	34
4.4 Manifest file code	60
5. RESULTS	
5.1 Sample Output (Screenshots)	62
6. CONCLUSION & FUTURE WORK	71
REFERENCES	72

CHAPTER 1

INTRODUCTION

1.1 Introduction to Android Application:

The Android Application is the largest installed base among various mobile platforms across the globe. Hundreds of millions of mobile devices are powered by Android in more than 190 countries of the world. Android Application is a software designed to run on an Android device or emulator. The term also refers to an APK file which stands for Android package. Android apps can be written in Kotlin, Java, and C++ and are run inside Virtual Machine. The official development environment is Android Studio. The compiled code along with any data and resource files required by the application is bundled by the apt tool into an Android package, an archive file marked by an .apk suffix. This file is the vehicle for distributing the application and installing it on mobile devices; it's the file users download to their devices. All the code in a single .apk file is considered to be one application. By default, every application runs in its own Linux process, with a unique Linux user id. Each process has its own virtual machine (VM).

1.1.1 Android Application Components:

- **Activities:** The application's presentation layer. Every screen in your application will be an extension of the Activity class. Activities use Views to form graphical user interfaces that displays information and respond to user actions. In terms of desktop development, an Activity is equivalent to a Form.
- **Services:** Services are the invisible workers of your application. Service components run invisibly, updating your data sources and visible Activities and triggering Notifications.
- **Content Providers:** Are a shareable data store. Content Providers are used to manage and share application databases. Content Providers are the preferred way of sharing data across application boundaries
- **Intents:** They are simple message-passing framework. Using Intents, you can broadcast messages system-wide or to a target Activity or Service, stating your intention to have an action performed.
- **Broadcast Receivers:** By creating and registering a Broadcast Receiver, your application can listen for broadcast Intents that match specific filter criteria.

- **Notifications:** They are user notification framework. Notifications let you signal users without stealing focus or interrupting their current Activities. They are the preferred technique for getting a user's attention from within a Service or Broadcast Receiver

1.2 Introduction to Android Studio:

Android Studio is the official integrated development environment (IDE) for Android application development. It is based on the IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools. To support application development within the Android operating system, Android Studio uses a Gradle- based build system, emulator, code templates, and GitHub integration. Every project in Android Studio has one or more modalities with source code and resource files. These modalities include Android app modules, Library modules, and Google App Engine modules.

1.2.1 Features:

Android Studio is having many cool features which can foster and could help you to increase development productivity. Here are the cool features:

- Powerful code editing (smart editing, code re-factoring)
- Rich layout Editor (As you soon as you drag and drop views on the layout, it shows you preview in all the screens including Nexus 4, Nexus 7, Nexus 10 and many other resolutions. Layout designing can be done much faster way as compared to eclipse.)
- Gradle-based build support
- Maven Support
- Templated based wizards
- Lint tool analysis (The Android lint tool is a static code analysis tool that checks your Android project source files for potential bugs and optimization improvements for correctness, security, performance, usability, accessibility, and internationalization).
- It provides ability to capture directly generate a screenshot of your application. Yes it was already included in the SDK but Android Studio provides something more:
- Device frame (As frames for many Nexus devices are available, you can capture screenshot in whichever frame you like most)

- Drop shadow
- Screen glare

1.3 Miniproject Description:

Notes are a great way to keep track of information for yourself. You can quickly capture what's on your mind and get a reminder later at the right place or time. Note taking forces you to pay attention and helps you focus in class (or while reading a textbook). It helps you learn. Studies on learning have shown that actively summarizing what you note helps you understand and remember the information later. Your notes serve as your guide to doing your job better, too; you can easily refer to the important information you need to succeed whenever you need it, without delay. And a secret bonus, taking notes actually makes you smarter.

CHAPTER 2

REQUIREMENT SPECIFICATION

2.1 Hardware Requirements

- ✦ Processor: Intel core i3 and i5
- ✦ RAM: 8GB (Minimum 4GB)

2.2 Software Requirements

- ✦ Operating System: Windows 10
- ✦ Tool/IDE (with version) used: Android Studio 4.1.3
- ✦ Emulator Used: Pixel 4a API 30
- ✦ Language used: Java
- ✦ Database used: Firebase

CHAPTER 3

DESIGN

3.1 Architectural Diagram of the Application

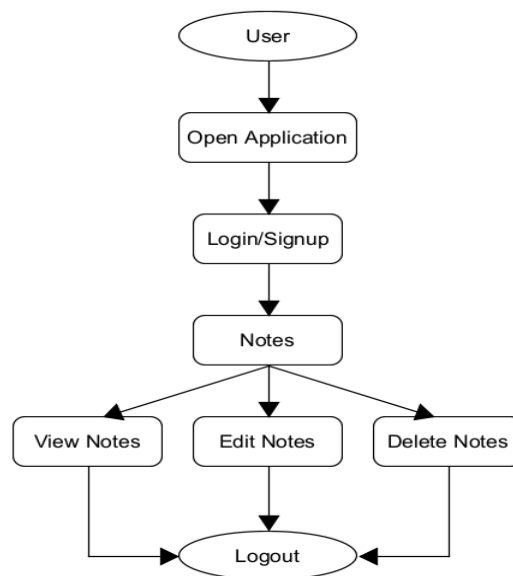


Fig (3.1): Architectural diagram

Explanation:

- In our application, the user needs to register himself first by entering his/her email and password. If he/she has an account, can directly log in.
- Then the user has to login by entering the email and password.
- The user can opt to either view available jobs or promote jobs available.
- If user chooses to promote job(s), then he/she can post available jobs under post a job button.
- If user chooses to view available jobs the he can do so under view all posts button.
- Then the user can logout otherwise.

3.2 XML Code

3.2.1 activity_splash_screen.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#06283D"
    tools:context=".splash_screen">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="100dp"
        android:layout_height="100dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:src="@drawable/notedown_logo"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.503">

    </ImageView>

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="275dp"
        android:text="NOTE DOWN APP"
        android:textAllCaps="true"
        android:textColor="#DFF6FF"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent">

    </TextView>

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="40dp"
        android:text="- Powered by team sp -"
        android:textAllCaps="true"
        android:textColor="#47B5FF"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent">

    </TextView>

    <ProgressBar
        android:id="@+id/progressbar"
        style="?android:attr/progressBarStyle"
        android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"
        android:indeterminateTint="#DFF6FF"
        android:visibility="invisible"
        android:layout_marginBottom="160dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

3.2.2 activity_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#06283D"
    tools:context=".MainActivity">

    <ProgressBar
        android:id="@+id/progressbarofmainactivity"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/gotoforgotpassword"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:indeterminateTint="#DFF6FF"
        android:visibility="invisible">

    </ProgressBar>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/centerline"
        android:layout_centerInParent="true">
    </RelativeLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginRight="30dp"
        android:layout_above="@id/centerline"
        android:layout_marginBottom="150dp"
        android:orientation="vertical">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Welcome Back"
            android:textSize="42sp"
            android:textAlignment="center"
            android:textColor="#47B5FF"
            android:textStyle="bold"
            android:gravity="center_horizontal">
        </TextView>
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"

```

```

        android:text="Login to continue"
        android:textAlignment="center"
        android:textSize="23sp"
        android:textColor="#DFF6FF"
        android:gravity="center_horizontal">
    </TextView>
</LinearLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_above="@id/centerline"
    android:layout_marginBottom="10dp"
    android:id="@+id/email"
    android:layout_marginRight="30dp"
    android:layout_marginLeft="30dp">
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/loginemail"
        android:hint="Enter Your Email"
        android:inputType="textEmailAddress">
    </EditText>
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/centerline"
    android:layout_marginTop="10dp"
    android:id="@+id/password"
    android:layout_marginRight="30dp"
    android:layout_marginLeft="30dp"
    app:passwordToggleEnabled="true">
    <EditText
        android:id="@+id/loginpassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Your Passworrd">
    </EditText>
</com.google.android.material.textfield.TextInputLayout>

<RelativeLayout
    android:id="@+id/login"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_below="@id/password"
    android:layout_marginLeft="30dp"
    android:layout_marginTop="30dp"
    android:layout_marginRight="30dp"
    android:background="@drawable/button_design">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:text="Login"
        android:textColor="@color/white"
        android:textSize="16sp"

```

```

        android:textStyle="bold" />
    </RelativeLayout>

    <TextView
        android:id="@+id/gotoforgotpassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/login"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="10dp"
        android:layout_marginRight="20dp"
        android:gravity="center_horizontal"
        android:text="Forgot Password ?"
        android:textAlignment="center"
        android:textColor="#DFF6FF" />

    <RelativeLayout
        android:id="@+id/gotosignup"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="30dp"
        android:layout_marginLeft="30dp"
        android:layout_marginEnd="30dp"
        android:layout_marginRight="30dp"
        android:layout_marginBottom="30dp"
        android:background="@drawable/button_design">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:layout_marginStart="8dp"
            android:layout_marginLeft="8dp"
            android:text="Sign Up For New Account"
            android:textColor="@color/white"
            android:textSize="16sp"
            android:textStyle="bold" />
    </RelativeLayout>

</RelativeLayout>

```

3.2.3 activity_signup.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#06283D"
    tools:context=".signup">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/centerline2"
        android:layout_centerInParent="true">

    </RelativeLayout>

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30dp"
    android:layout_marginRight="30dp"
    android:layout_above="@id/centerline2"
    android:layout_marginBottom="150dp"
    android:orientation="vertical">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Create Account"
        android:textSize="42sp"
        android:textAlignment="center"
        android:textColor="#47B5FF"
        android:textStyle="bold"
        android:gravity="center_horizontal">
    </TextView>
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Sign Up To Get Started"
        android:textAlignment="center"
        android:textSize="23sp"
        android:textColor="#DFF6FF"
        android:gravity="center_horizontal">
    </TextView>
</LinearLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_above="@id/centerline2"
    android:layout_marginBottom="10dp"
    android:id="@+id/email"
    android:layout_marginRight="30dp"
    android:layout_marginLeft="30dp">
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/signupemail"
        android:hint="Enter Your Email"
        android:inputType="textEmailAddress">
    </EditText>
</com.google.android.material.textfield.TextInputLayout>
<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/centerline2"
    android:layout_marginTop="10dp"
    android:id="@+id/password"
    android:layout_marginRight="30dp"
    android:layout_marginLeft="30dp"
    app:passwordToggleEnabled="true">

    <EditText
        android:id="@+id/signuppassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Your Passworrd">
    </EditText>

```

```

</com.google.android.material.textfield.TextInputLayout>

<RelativeLayout
    android:id="@+id/signup"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_below="@id/password"
    android:layout_marginLeft="30dp"
    android:layout_marginTop="30dp"
    android:layout_marginRight="30dp"
    android:background="@drawable/button_design">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:text="Sign Up"
        android:textColor="@color/white"
        android:textSize="16sp"
        android:textStyle="bold" />
</RelativeLayout>

<TextView
    android:id="@+id/gotologin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/signup"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="70dp"
    android:layout_marginRight="20dp"
    android:gravity="center_horizontal"
    android:text="Go Back To Login !"
    android:textAlignment="center"
    android:textColor="#DFF6FF"
    android:textSize="18sp" />
</RelativeLayout>

```

3.2.4 activity_forgotpassword.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#06283D"
    tools:context=".forgotpassword">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/centerline3"
        android:layout_centerInParent="true">
    </RelativeLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```



```

        android:layout_marginLeft="30dp"
        android:layout_marginRight="30dp"
        android:layout_above="@id/centerline3"
        android:layout_marginBottom="120dp"
        android:orientation="vertical">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Forgot Password"
            android:textSize="35sp"
            android:textAlignment="center"
            android:textColor="#47B5FF"
            android:textStyle="bold"
            android:gravity="center_horizontal">
        </TextView>
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Enter Registered Email"
            android:textAlignment="center"
            android:textSize="23sp"
            android:textColor="#DFF6FF"
            android:gravity="center_horizontal">
        </TextView>
    </LinearLayout>

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginRight="30dp"
        android:layout_above="@id/centerline3"
        android:hint="Enter Registered Email"
        android:id="@+id/forgotpassword"
        android:background="@color/white"
        android:padding="10dp"
        android:layout_marginBottom="10dp">
    </EditText>
    <Button
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:text="Recover Passsword"
        android:textAllCaps="false"
        android:textSize="16sp"
        android:id="@+id/passwordrecoverbutton"
        android:layout_below="@id/centerline3"
        android:layout_marginTop="10dp"
        android:layout_marginLeft="30dp"
        android:layout_marginRight="30dp"
        android:backgroundTint="#1363DF">
    </Button>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:id="@+id/gobacktologin"
        android:text="Go Back To Login !"
        android:layout_marginBottom="40dp"
        android:textColor="#DFF6FF"
        android:textAlignment="center"
        android:textSize="18sp"

```

```
        android:gravity="center_horizontal" />
</RelativeLayout>
```

3.2.5 notes_layout.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="15dp"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:elevation="5dp"
        app:cardCornerRadius="5dp"
        app:cardMaxElevation="5dp"
        android:id="@+id/notecard">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/note"
            android:orientation="vertical">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal">

                <TextView
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:id="@+id/notetitle"
                    android:paddingTop="10dp"
                    android:paddingRight="10dp"
                    android:paddingLeft="10dp"
                    android:paddingBottom="10dp"
                    android:text="Title"
                    android:textSize="20sp"
                    android:textColor="#06283D"
                    android:layout_weight="1">

                </TextView>

                <ImageView
                    android:layout_width="wrap_content"
                    android:layout_height="match_parent"
                    android:id="@+id/menupopbutton"
                    android:layout_weight="0"
                    android:src="@drawable/ic_baseline_more_vert_24">

                </ImageView>

            </LinearLayout>

            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:id="@+id/notecontent"
                android:paddingLeft="10dp"
                android:paddingTop="10dp"
                android:paddingRight="10dp"
                android:text="content"
                android:textSize="15sp"
```

```

        android:paddingBottom="20dp"
        android:textColor="#06283D">
    </TextView>

</LinearLayout>

</androidx.cardview.widget.CardView>
</RelativeLayout>

```

3.2.6 activity_createnote.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".createnote">

    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbarofcreatenote"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        android:background="#47B5FF">

        <EditText
            android:id="@+id/createtitleofnote"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:hint="Enter Title"
            android:paddingTop="12dp"
            android:textColor="#06283D"
            android:textSize="20sp"
            android:textStyle="bold">

            </EditText>
        </androidx.appcompat.widget.Toolbar>

        <EditText
            android:id="@+id/createcontentofnote"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_below="@id/toolbarofcreatenote"
            android:background="#06283D"
            android:gravity="top|left"
            android:hint="NoteDown"
            android:padding="15dp"
            android:textColor="#DFF6FF"
            android:textSize="19sp">

            </EditText>

        <com.google.android.material.floatingactionbutton.FloatingActionButton
            android:id="@+id/savenote"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentRight="true"
            android:layout_alignParentBottom="true"
            android:layout_gravity="bottom|end"
            android:layout_marginRight="30dp"

```

```

        android:layout_marginBottom="30dp"
        android:backgroundTint="#1363DF"
        android:src="@drawable/ic_baseline_save_24"
        app:maxImageSize="40dp">

</com.google.android.material.floatingactionbutton.FloatingActionButton>

<ProgressBar
    android:id="@+id/progressbarofcreatenote"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:layout_marginTop="20dp"
    android:indeterminateTint="#DFF6FF"
    android:visibility="invisible">

</ProgressBar>
</RelativeLayout>

```

3.2.7 activity_editnoteactivity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".editnoteactivity">

    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbarofeditnote"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        android:background="#47B5FF">

        <EditText
            android:id="@+id/edittitleofnote"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:paddingTop="12dp"
            android:textColor="#06283D"
            android:textSize="20sp"
            android:textStyle="bold">

            </EditText>
        </androidx.appcompat.widget.Toolbar>

        <EditText
            android:id="@+id/editcontentofnote"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_below="@id/toolbarofeditnote"
            android:background="#06283D"
            android:gravity="top|left"
            android:padding="15dp"
            android:textColor="#DFF6FF"
            android:textSize="15sp">

            </EditText>

        <com.google.android.material.floatingactionbutton.FloatingActionButton

```

```

        android:id="@+id/saveeditnote"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:layout_gravity="bottom|end"
        android:layout_marginRight="30dp"
        android:layout_marginBottom="30dp"
        android:backgroundTint="#1363DF"
        android:src="@drawable/ic_baseline_save_24"
        app:maxImageSize="40dp">

</com.google.android.material.floatingactionbutton.FloatingActionButton>

</RelativeLayout>

```

3.2.8 activity_notesactivity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#06283D"
    tools:context=".notesactivity">

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_marginBottom="30dp"
        android:layout_marginRight="30dp"
        android:layout_gravity="bottom|end"
        android:id="@+id/createnotefab"
        android:src="@drawable/ic_baseline_add"
        android:backgroundTint="#1363DF"
        app:maxImageSize="50dp"
        android:layout_marginEnd="30dp"
        android:layout_alignParentEnd="true">
    </com.google.android.material.floatingactionbutton.FloatingActionButton>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerview"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scrollbars="vertical">

    </androidx.recyclerview.widget.RecyclerView>

</RelativeLayout>

```

3.2.9 activity_notesdetails.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"

```

```
android:layout_height="match_parent"
tools:context=".notedetails">

<androidx.appcompat.widget.Toolbar
    android:id="@+id/toolbarofnotedetail"
    android:layout_width="match_parent"
    android:layout_height="?attr/actionBarSize"
    android:background="#47B5FF">

    <TextView
        android:id="@+id/titleofnotedetail"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingTop="12dp"
        android:text="Title displayed"
        android:textColor="#06283D"
        android:textSize="20sp"
        android:textStyle="bold">

    </TextView>
</androidx.appcompat.widget.Toolbar>

<TextView
    android:id="@+id/contentofnotedetail"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@id/toolbarofnotedetail"
    android:background="#06283D"
    android:gravity="top|left"
    android:hint="Notes....."
    android:padding="15dp"
    android:textColor="#DFF6FF"
    android:textSize="17sp">

</TextView>

<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/gotoeditnote"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    android:layout_gravity="bottom|end"
    android:layout_marginRight="30dp"
    android:layout_marginBottom="30dp"
    android:backgroundTint="#1363DF"
    android:src="@drawable/ic_baseline_edit_24"
    app:maxImageSize="40dp">

</com.google.android.material.floatingactionbutton.FloatingActionButton>
</RelativeLayout>
```

3.3 Database Schema diagram

NOTES

<u>User ID</u>	Notes Title	Notes Content
----------------	-------------	---------------

CHAPTER 4

IMPLEMENTATION

4.1 List of classes imported

- **NonNull :**
Denotes that a parameter, field or method return value can never be null
- **AppCompatActivity :**
Base class for activities that wish to use some of the newer platform features on older Android devices.
- **ProgressDialog :**
A dialog showing a progress indicator and an optional text message or view. Only a text message or a view can be used at the same time
- **Intent :**
An intent is an abstract description of an operation to be performed.
- **Bundle :**
A mapping from String keys to various Parcelable values.
- **View :**
This class represents the basic building block for user interface components. A View occupies a rectangular area on the screen and is responsible for drawing and event handling
- **Button :**
A user interface element the user can tap or click to perform an action.
- **EditText :**
A user interface element for entering and modifying text.
- **ProgressBar :**
A user interface element that indicates the progress of an operation. Progress bar supports two modes to represent progress: determinate, and indeterminate.
- **Toast :**
A toast is a view containing a quick little message for the user. The toast class helps you create and show those.
- **OnCompleteListener :**
Listener called when Task completes.
- **Task :**
The task class represents a single operation that doesnot return a value and that usually executes asynchronously.

- **AuthResult :**
Result object obtained from operations that can affect the authentication state. Contains a method that returns the currently signed-in user after the operation has completed
- **FirebaseAuth :**
The entry point of the Firebase Authentication SDK.
- **SupressLint :**
Indicates that Lint should ignore the specified warnings for the annotated element.
- **LayoutInflater :**
Instantiates a layout XML file into its corresponding View objects.
- **ViewGroup :**
A ViewGroup is a special view that can contain other views (called children.) The view group is the base class for layouts and views containers.
- **TintTypedArray :**
A class that wraps a TypedArray and provides the same public API surface. The purpose of this class is so that we can intercept calls to new APIs.
- **Toolbar :**
A standard toolbar for use within application content.
- **LinearLayoutManager :**
A RecyclerView.LayoutManager implementation which provides similar functionality to android.widget.ListView.
- **RecyclerView :**
A flexible view for providing a limited window into a large data set.
- **FirebaseRecyclerAdapter :**
This class is a generic way of backing a RecyclerView with a Firebase location. It handles all of the child events at the given Firebase location and marshals received data into the given class type
- **FirebaseRecyclerOptions :**
Options to configure a FirebaseRecyclerAdapter.
- **DatabaseReference :**
A Firebase reference represents a particular location in your Database and can be used for reading or writing data to that Database location.
- **FirebaseDatabase :**
The entry point for accessing a Firebase Database. You can get an instance by calling getInstance() . To access a location in the database and read or write data, use getReference() .

- **Menu :**
Interface for managing the items in a menu. By default, every Activity supports an options menu of actions or options.
- **MenuItem :**
Interface for direct access to a previously created menu item.
- **FloatingActionButton :**
Floating action buttons are used for a special type of promoted action. They are distinguished by a circled icon floating above the UI and have special motion behaviors related to morphing, launching, and the transferring anchor point.

4.2 List of imported Library Functions

- **onCreate () :**
Called when the activity is starting.
- **getSupportActionBar() :**
Set a Toolbar to act as the ActionBar for this Activity window.
- **setTitle() :**
Set the action bar's title. This will only be displayed if `DISPLAY_SHOW_TITLE` is set.
- **setHomeButtonEnabled() :**
Enable or disable the "home" button in the corner of the action bar.
- **getInstance() :**
Gets the default FirebaseDatabase instance.
- **getReference() :**
Gets a DatabaseReference for the database root node.
- **setHasFixedSize() :**
RecyclerView can perform several optimizations if it can know in advance that RecyclerView's size is not affected by the adapter contents.
- **setLayoutManager() :**
Set the RecyclerView.LayoutManager that this RecyclerView will use.
- **onBindViewHolder ():**
The new ViewHolder will be used to display items of the adapter using this.
- **setText() :**
Sets the text to be displayed.
- **onCreateViewHolder() :**

Creates new ViewHolder object whenever the RecyclerView needs a new one.

- **startListening() :**
Start listening for database changes and populate the adapter.
- **setAdapter() :**
Set a new adapter to provide child views on demand.
- **onCreateOptionsMenu() :**
Initialize the contents of the Activity's standard options menu. You should place your menu items in to menu.
- **onOptionsItemSelected() :**
This hook is called whenever an item in your options menu is selected. The default implementation simply returns false to have the normal processing happen
- **toString() :**
Returns a string representation of the object. In general, the toString method returns a string that "textually represents" this object.
- **trim() :**
Returns a string whose value is this string, with any leading and trailing whitespace removed.
- **setError() :**
Sets the right-hand compound drawable of the TextView to the "error" icon and sets an error message that will be displayed in a popup when the TextView has focus.
- **getDateInstance() :**
Gets the date formatter with the default formatting style for the default FORMAT locale.
- **makeText() :**
Make a standard toast that just contains text.
- **getStringExtra() :**
Retrieve extended data from the intent.
- **setStackFromEnd() :**
Compatibility support for android.widget.AbsListView.setStackFromBottom(boolean)
- **LinearLayoutManager() :**
Creates a vertical LinearLayoutManager.

4.3 Java Code

4.3.1 splash_screen

```
package com.example.notedownapp;

import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.ProgressBar;

import androidx.appcompat.app.AppCompatActivity;

public class splash_screen extends AppCompatActivity {
    ProgressBar mprogressbar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash_screen);

        mprogressbar=findViewById(R.id.progressbar);
        mprogressbar.setVisibility(View.VISIBLE);
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {

                startActivity(new Intent(splash_screen.this, MainActivity.class));
                finish();
            }
        }, 5000);
    }
}
```

4.3.2 Mainactivity

```
package com.example.notedownapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.RelativeLayout;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

public class MainActivity extends AppCompatActivity {

    private EditText mloginemail,mloginpassword;
```

```

private RelativeLayout mlogin,mgotosignup;
private TextView mgotoforgotpassword;

private FirebaseAuth firebaseAuth;

ProgressBar mprogressbarofmainactivity;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    getSupportActionBar().hide();

    mloginemail=findViewById(R.id.loginemail);
    mloginpassword=findViewById(R.id.loginpassword);
    mlogin=findViewById(R.id.login);
    mgotosignup=findViewById(R.id.gotosignup);
    mgotoforgotpassword=findViewById(R.id.gotoforgotpassword);

    mprogressbarofmainactivity=findViewById(R.id.progressbarofmainactivity);

    firebaseAuth=FirebaseAuth.getInstance();
    FirebaseUser firebaseUser=firebaseAuth.getCurrentUser();

    if(firebaseUser!=null){
        finish();
        startActivity(new Intent(MainActivity.this,notesactivity.class));
    }

    mgotosignup.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent(MainActivity.this,signup.class));
        }
    });

    mgotoforgotpassword.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent(MainActivity.this,forgotpassword.class));
        }
    });

    mlogin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String mail=mloginemail.getText().toString().trim();
            String password=mloginpassword.getText().toString().trim();

            if(mail.isEmpty()||password.isEmpty()){
                Toast.makeText(getApplicationContext(),"All fields are
required",Toast.LENGTH_SHORT).show();
            }
            else{
                mprogressbarofmainactivity.setVisibility(View.VISIBLE);

                firebaseAuth.signInWithEmailAndPassword(mail,password).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {

```

```

        if(task.isSuccessful()){
            checkmailverification();
        }
        else{
            Toast.makeText(getApplicationContext(),"Account does
not exist",Toast.LENGTH_SHORT).show();
mprogressbarofmainactivity.setVisibility(View.INVISIBLE);
        }
    }
    });
}
});
}

private void checkmailverification(){
    FirebaseUser firebaseUser=firebaseAuth.getCurrentUser();
    if(firebaseUser.isEmailVerified()==true){
        Toast.makeText(getApplicationContext(),"Logged
In",Toast.LENGTH_SHORT).show();
        finish();
        startActivity(new Intent(MainActivity.this,notesactivity.class));
    }
    else{
        mprogressbarofmainactivity.setVisibility(View.INVISIBLE);
        Toast.makeText(getApplicationContext(),"Verify your Email
first",Toast.LENGTH_SHORT).show();
        firebaseAuth.signOut();
    }
}
}
}

```

4.3.3 signup

```

package com.example.notedownapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.RelativeLayout;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

public class signup extends AppCompatActivity {
    private EditText msignupemail,msignuppassword;
    private RelativeLayout msignup;
    private TextView mgotologin;

    private FirebaseAuth firebaseAuth;

    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_signup);

    getSupportActionBar().hide();

    msignupemail=findViewById(R.id.signupemail);
    msignuppassword=findViewById(R.id.signuppassword);
    msignup=findViewById(R.id.signup);
    mgotologin=findViewById(R.id.gotologin);

    firebaseAuth=FirebaseAuth.getInstance();

    mgotologin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent=new Intent(signup.this,MainActivity.class);
            startActivity(intent);
        }
    });

    msignup.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String mail=msignupemail.getText().toString().trim();
            String password=msignuppassword.getText().toString().trim();

            if(mail.isEmpty()||password.isEmpty()){
                Toast.makeText(getApplicationContext(),"All fields are
required",Toast.LENGTH_SHORT).show();
            }
            else if(password.length()<7){
                Toast.makeText(getApplicationContext(),"Password should contain 8
characters",Toast.LENGTH_SHORT).show();
            }
            else{

firebaseAuth.createUserWithEmailAndPassword(mail,password).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if(task.isSuccessful()){
                        Toast.makeText(getApplicationContext(),"Registration
Successful",Toast.LENGTH_SHORT).show();
                        sendEmailVerification();
                    }
                    else{
                        Toast.makeText(getApplicationContext(),"Failed To
Register",Toast.LENGTH_SHORT).show();
                    }
                }
            });
        }
    });
}

private void sendEmailVerification(){
    FirebaseUser firebaseUser=firebaseAuth.getCurrentUser();
    if(firebaseUser!=null){
        firebaseUser.sendEmailVerification().addOnCompleteListener(new
OnCompleteListener<Void>() {

```

```

        @Override
        public void onComplete(@NonNull Task<Void> task) {
            Toast.makeText(getApplicationContext(),"Verification Mail
Sent!,Verify and Login",Toast.LENGTH_SHORT).show();
            firebaseAuth.signOut();
            finish();
            startActivity(new Intent(signup.this,MainActivity.class));
        }
    });
}
else{
    Toast.makeText(getApplicationContext(),"Failed to send Verification
Email",Toast.LENGTH_SHORT).show();
}
}
}
}

```

4.3.4 forgotpassword

```

package com.example.notedownapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;

public class forgotpassword extends AppCompatActivity {
    private EditText mforgotpassword;
    private Button mpasswordrecoverbutton;
    private TextView mgobacktologin;

    FirebaseAuth firebaseAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_forgotpassword);

        getSupportActionBar().hide();

        mforgotpassword=findViewById(R.id.forgotpassword);
        mpasswordrecoverbutton=findViewById(R.id.passwordrecoverbutton);
        mgobacktologin=findViewById(R.id.gobacktologin);

        firebaseAuth=FirebaseAuth.getInstance();

        mgobacktologin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(forgotpassword.this,MainActivity.class);
                startActivity(intent);
            }
        });
    }
}

```



```

    }
});

mpasswordrecoverbutton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String mail=mforgotpassword.getText().toString().trim();
        if(mail.isEmpty()){
            Toast.makeText(getApplicationContext(),"Enter Your
Email",Toast.LENGTH_SHORT).show();
        }
        else{
firebaseAuth.sendPasswordResetEmail(mail).addOnCompleteListener(new
OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task) {
                if(task.isSuccessful()){
                    Toast.makeText(getApplicationContext(),"Mail sent,You
can recover password using mail",Toast.LENGTH_SHORT).show();
                    finish();
                    startActivity(new
Intent(forgotpassword.this,MainActivity.class));
                }
                else{
                    Toast.makeText(getApplicationContext(),"Email is
wrong/Account does not exist",Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}

});
}
}
}
}

```

4.3.5 createnote

```
package com.example.notedownapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.Toast;

import androidx.appcompat.widget.Toolbar;

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;
```

```

import java.util.HashMap;
import java.util.Map;

public class createnote extends AppCompatActivity {

    EditText mcreatetitleofnote,mcreatecontentofnote;
    FloatingActionButton msavenote;

    FirebaseAuth firebaseAuth;
    FirebaseUser firebaseUser;
    FirebaseFirestore firebaseFirestore;

    ProgressBar mprogressbarofcreatenote;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_createnote);

        mcreatecontentofnote=findViewById(R.id.createcontentofnote);
        mcreatetitleofnote=findViewById(R.id.createtitleofnote);
        msavenote=findViewById(R.id.savenote);
        mprogressbarofcreatenote=findViewById(R.id.progressbarofcreatenote);

        Toolbar toolbar=findViewById(R.id.toolbarofcreatenote);
        setSupportActionBar(toolbar);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        firebaseAuth=FirebaseAuth.getInstance();
        firebaseFirestore=FirebaseFirestore.getInstance();
        firebaseUser=FirebaseAuth.getInstance().getCurrentUser();

        msavenote.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String title=mcreatetitleofnote.getText().toString();
                String content=mcreatecontentofnote.getText().toString();
                if(title.isEmpty()||content.isEmpty()){
                    Toast.makeText(getApplicationContext(),"Both fields
required",Toast.LENGTH_SHORT).show();
                }
                else{
                    mprogressbarofcreatenote.setVisibility(View.VISIBLE);
                    DocumentReference
documentReference=firebaseFirestore.collection("notes").document(firebaseUser.getId(
)).collection("myNotes").document();
                    Map<String,Object> note=new HashMap<>();
                    note.put("title",title);
                    note.put("content",content);
                    documentReference.set(note).addOnSuccessListener(new
OnSuccessListener<Void>() {
                        @Override
                        public void onSuccess(Void aVoid) {
                            Toast.makeText(getApplicationContext(),"Note created
successfully",Toast.LENGTH_SHORT).show();
                            startActivity(new
Intent(createnote.this,notesactivity.class));
                        }
                    }).addOnFailureListener(new OnFailureListener() {
                        @Override
                        public void onFailure(@NonNull Exception e) {

```

```

        Toast.makeText(getApplicationContext(),"Failed to create
note",Toast.LENGTH_SHORT).show();
        mprogressbarofcreatenote.setVisibility(View.INVISIBLE);
//        startActivity(new
Intent(createnote.this,notesactivity.class));
    }
    });
}
}
});
}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {

    switch (item.getItemId()){
        case android.R.id.home:
            finish();
            startActivity(new Intent(createnote.this,notesactivity.class));
    }
    if(item.getItemId()== android.R.id.home) {
        onBackPressed();
    }
    return super.onOptionsItemSelected(item);
}
}

```

4.3.6 editnoteactivity

```

package com.example.notedownapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.widget.Toolbar;
import androidx.core.app.NavUtils;

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;

import java.util.HashMap;
import java.util.Map;

public class editnoteactivity extends AppCompatActivity {

    Intent data;
    EditText medittitleofnote,meditcontentofnote;

```

```

FloatingActionButton msaveeditnote;

FirebaseAuth firebaseAuth;
Firestore firebaseFirestore;
FirebaseUser firebaseUser;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_editnoteactivity);

    medittitleofnote=findViewById(R.id.edittitleofnote);
    meditcontentofnote=findViewById(R.id.editcontentofnote);
    msaveeditnote=findViewById(R.id.saveeditnote);
    data=getIntent();

    firebaseFirestore=Firestore.getInstance();
    firebaseUser=FirebaseAuth.getInstance().getCurrentUser();

    Toolbar toolbar=findViewById(R.id.toolbarofeditnote);
    setSupportActionBar(toolbar);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);

    msaveeditnote.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            //Toast.makeText(getApplicationContext(),"Save button",
            Toast.LENGTH_SHORT).show();
            String newtitle=medittitleofnote.getText().toString();
            String newcontent=meditcontentofnote.getText().toString();
            if(newtitle.isEmpty()||newcontent.isEmpty()){
                Toast.makeText(getApplicationContext(),"Something is empty",
                Toast.LENGTH_SHORT).show();
            }
            else{
                DocumentReference
                documentReference=firebaseFirestore.collection("notes").document(firebaseUser.getId()
                ).collection("myNotes").document(data.getStringExtra("noteId"));
                Map<String,Object> note=new HashMap<>();
                note.put("title",newtitle);
                note.put("content",newcontent);
                documentReference.set(note).addOnSuccessListener(new
                OnSuccessListener<Void>() {
                    @Override
                    public void onSuccess(Void aVoid) {
                        Toast.makeText(getApplicationContext(),"Note is updated",
                        Toast.LENGTH_SHORT).show();
                        startActivity(new
                        Intent(editnoteactivity.this,notesactivity.class));
                    }
                }).addOnFailureListener(new OnFailureListener() {
                    @Override
                    public void onFailure(@NonNull Exception e) {
                        Toast.makeText(getApplicationContext(),"Failed to
                        update", Toast.LENGTH_SHORT).show();
                    }
                });
            }
        }
    });
}

```

```

String notetitle=data.getStringExtra("title");
String notecontent=data.getStringExtra("content");
meditcontentofnote.setText(notecontent);
meditttitleofnote.setText(notetitle);

}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {

    switch (item.getItemId()) {
        case android.R.id.home:
            finish();
            startActivity(new Intent(editnoteactivity.this, notesactivity.class));
    }
    // if(item.getItemId()== android.R.id.home){
    //     onBackPressed();
    // }
    return super.onOptionsItemSelected(item);
}
}

```

4.3.7 notedetails

```

package com.example.notedownapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.TextView;
import androidx.appcompat.widget.Toolbar;
import androidx.core.app.NavUtils;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

public class notedetails extends AppCompatActivity {

    private TextView mtitleofnotedetail,mcontentofnotedetail;
    FloatingActionButton mgotoeditnote;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_notedetails);
        mtitleofnotedetail=findViewById(R.id.titleofnotedetail);
        mcontentofnotedetail=findViewById(R.id.contentofnotedetail);
        mgotoeditnote=findViewById(R.id.gotoeditnote);

        Toolbar toolbar=findViewById(R.id.toolbarofnotedetail);
        setSupportActionBar(toolbar);

        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        Intent data=getIntent();

        mgotoeditnote.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

```

```

        Intent intent=new Intent(v.getContext(),editnoteactivity.class);
        intent.putExtra("title",data.getStringExtra("title"));
        intent.putExtra("content",data.getStringExtra("content"));
        intent.putExtra("noteId",data.getStringExtra("noteId"));
        v.getContext().startActivity(intent);
    }
});

mcontentofnotedetail.setText(data.getStringExtra("content"));
mtitleofnotedetail.setText(data.getStringExtra("title"));
}
@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {

    switch (item.getItemId()) {
        case android.R.id.home:
            finish();
            startActivity(new Intent(notedetails.this,notesactivity.class));
    }
    // if(item.getItemId()== android.R.id.home) {
    //     onBackPressed();
    // }
    return super.onOptionsItemSelected(item);
}
}

```

4.3.8 notesactivity

```

package com.example.notedownapp;

import androidx.annotation.NonNull;
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.RecyclerView;
import androidx.recyclerview.widget.StaggeredGridLayoutManager;

import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.view.Gravity;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.PopupMenu;
import android.widget.TextView;
import android.widget.Toast;

import com.firebase.ui.firestore.FirestoreRecyclerAdapter;
import com.firebase.ui.firestore.FirestoreRecyclerOptions;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;

```

```

import com.google.firebase.firestore.Query;

import org.w3c.dom.Document;

import java.util.ArrayList;
import java.util.List;
import java.util.Random;

public class notesactivity extends AppCompatActivity {

    FloatingActionButton mcreatenotefab;
    private FirebaseAuth firebaseAuth;

    RecyclerView mrecyclerview;
    StaggeredGridLayoutManager staggeredGridLayoutManager;

    FirebaseUser firebaseUser;
    FirebaseFirestore firebaseFirestore;
    FirestoreRecyclerAdapter<firebasemodel, NoteViewHolder> noteAdapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_notesactivity);

        mcreatenotefab=findViewById(R.id.createnotefab);
        firebaseAuth=FirebaseAuth.getInstance();

        firebaseUser=FirebaseAuth.getInstance().getCurrentUser();
        firebaseFirestore=FirebaseFirestore.getInstance();

        getSupportActionBar().setTitle("Notes");
        mcreatenotefab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(notesactivity.this, createnote.class));
            }
        });

        Query
        query=firebaseFirestore.collection("notes").document(firebaseUser.getUid()).collection("myNotes").orderBy("title", Query.Direction.ASCENDING);

        FirestoreRecyclerOptions<firebasemodel>allusernotes=new
        FirestoreRecyclerOptions.Builder<firebasemodel>().setQuery(query, firebasemodel.class)
        .build();

        noteAdapter=new FirestoreRecyclerAdapter<firebasemodel,
        NoteViewHolder>(allusernotes) {
            @RequiresApi(api = Build.VERSION_CODES.M)
            @Override
            protected void onBindViewHolder(@NonNull NoteViewHolder noteViewHolder,
            int i, @NonNull firebasemodel firebasemodel) {

                ImageView
                popupbutton=noteViewHolder.itemView.findViewById(R.id.menupopbutton);

                int colorcode=getRandomColor();

                noteViewHolder.mnote.setBackgroundColor(noteViewHolder.itemView.getResources().getColor(colorcode, null));
            }
        };
    }
}

```

```

noteViewHolder.noteTitle.setText(firebaseModel.getTitle());
noteViewHolder.noteContent.setText(firebaseModel.getContent());

String docId=noteAdapter.getSnapshots().getSnapshot(i).getId();

noteViewHolder.itemView.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View v) {
        Intent intent=new Intent(v.getContext(),notedetails.class);
        intent.putExtra("title",firebaseModel.getTitle());
        intent.putExtra("content",firebaseModel.getContent());
        intent.putExtra("noteId",docId);
        v.getContext().startActivity(intent);

        //Toast.makeText(getApplicationContext(),"This is
clicked",Toast.LENGTH_SHORT).show();
    }
});

popupbutton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        PopupMenu popupMenu=new PopupMenu(v.getContext(),v);
        popupMenu.setGravity(Gravity.END);

popupMenu.getMenu().add("Edit").setOnMenuItemClickListener(new
MenuItem.OnMenuItemClickListener() {
        @Override
        public boolean onMenuItemClick(MenuItem item) {

            Intent intent=new
Intent(v.getContext(),editnoteactivity.class);
            intent.putExtra("title",firebaseModel.getTitle());

intent.putExtra("content",firebaseModel.getContent());
            intent.putExtra("noteId",docId);
            v.getContext().startActivity(intent);
            return false;
        }
    });

popupMenu.getMenu().add("Delete").setOnMenuItemClickListener(new
MenuItem.OnMenuItemClickListener() {
        @Override
        public boolean onMenuItemClick(MenuItem item) {
            //Toast.makeText(v.getContext(), "This note is
deleted", Toast.LENGTH_SHORT).show();
            DocumentReference
documentReference=firebaseFirestore.collection("notes").document(firebaseUser.getUid(
)).collection("myNotes").document(docId);
            documentReference.delete().addOnSuccessListener(new
OnSuccessListener<Void>() {
                @Override
                public void onSuccess(Void aVoid) {
                    Toast.makeText(v.getContext(), "This note is
deleted", Toast.LENGTH_SHORT).show();
                }
            }).addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {

```



```

Toast.makeText(v.getContext(), "Failed to
delete", Toast.LENGTH_SHORT).show();
    }
    });
    return false;
    }
    });
    popupMenu.show();
    }
    });

    }

    @NonNull
    @Override
    public NoteViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.notes_layout,parent,false);
        return new NoteViewHolder(view);
    }
    };

    mrecyclerview=findViewById(R.id.recyclerview);
    mrecyclerview.setHasFixedSize(true);
    staggeredGridLayoutManager=new
StaggeredGridLayoutManager(2,StaggeredGridLayoutManager.VERTICAL);
    mrecyclerview.setLayoutManager(staggeredGridLayoutManager);
    mrecyclerview.setAdapter(noteAdapter);
}

public class NoteViewHolder extends RecyclerView.ViewHolder{

    private TextView notetitle;
    private TextView notecontent;
    LinearLayout mnote;
    public NoteViewHolder(@NonNull View itemView) {
        super(itemView);
        notetitle=itemView.findViewById(R.id.notetitle);
        notecontent=itemView.findViewById(R.id.notecontent);
        mnote=itemView.findViewById(R.id.note);
    }

}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu,menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()){
        case R.id.logout:
            firebaseAuth.signOut();
            finish();
            startActivity(new Intent(notesactivity.this,MainActivity.class));
        }
    return super.onOptionsItemSelected(item);
}
}

```

```

@Override
protected void onStart() {
    super.onStart();
    noteAdapter.startListening();
}

@Override
protected void onStop() {
    super.onStop();
    if(noteAdapter!=null){
        noteAdapter.stopListening();
    }
}

private int getRandomColor(){
    List<Integer> colorcode=new ArrayList<>();
    colorcode.add(R.color.c1);
    colorcode.add(R.color.c2);
    colorcode.add(R.color.c3);
    colorcode.add(R.color.c4);
    colorcode.add(R.color.c5);
    colorcode.add(R.color.c6);
    colorcode.add(R.color.c7);
    colorcode.add(R.color.c8);
    colorcode.add(R.color.c9);
    colorcode.add(R.color.c10);
    colorcode.add(R.color.c11);
    colorcode.add(R.color.c12);
    colorcode.add(R.color.c13);
    colorcode.add(R.color.c14);
    colorcode.add(R.color.c15);

    Random random=new Random();
    int number=random.nextInt(colorcode.size());
    return colorcode.get(number);
}
}

```

4.3.9 firebase model

```

package com.example.notedownapp;

public class firebase model {

    private String title;
    private String content;

    public firebase model(){
    }

    public firebase model(String title,String content){
        this.title=title;
        this.content=content;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {

```

```

        this.title = title;
    }

    public String getContent() {
        return content;
    }

    public void setContent(String content) {
        this.content = content;
    }
}

```

4.3 Manifest file code

AndroidManifest.xml

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.notedownapp">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.NoteDownApp">
        <activity
            android:name=".notedetails"
            android:exported="false"
            android:theme="@style/Theme.AppCompat.DayNight.NoActionBar" />
        <activity
            android:name=".editnoteactivity"
            android:exported="false"
            android:theme="@style/Theme.AppCompat.DayNight.NoActionBar" />
        <activity
            android:name=".createnote"
            android:exported="false"
            android:theme="@style/Theme.AppCompat.DayNight.NoActionBar" />
        <activity
            android:name=".notesactivity"
            android:exported="false" />
        <activity
            android:name=".forgotpassword"
            android:exported="false" />
        <activity
            android:name=".signup"
            android:exported="false" />

        <!--
        <activity-->
        <!--
            android:name=".MainActivity"-->
        <!--
            android:exported="true"-->
        <!--
            <intent-filter>-->
        <!--
            <action android:name="android.intent.action.MAIN" />-->

        <!--
            <category android:name="android.intent.category.LAUNCHER" />-->
        <!--
            </intent-filter>-->
        <!--
            </activity>-->

        <activity
            android:name=".MainActivity"

```

```
        android:exported="true">
    </activity>

    <activity
        android:name=".splash_screen"
        android:exported="true"
        android:theme="@style/SplashScreen">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</application>

</manifest>
```

CHAPTER 5

RESULTS

5.1 SCREENSHOTS

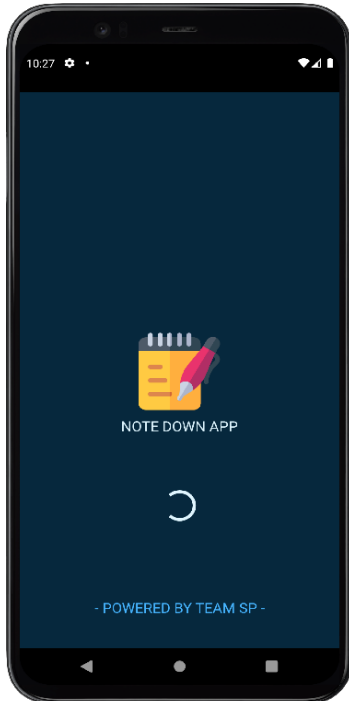


Figure 5.1.1: Splash Screen

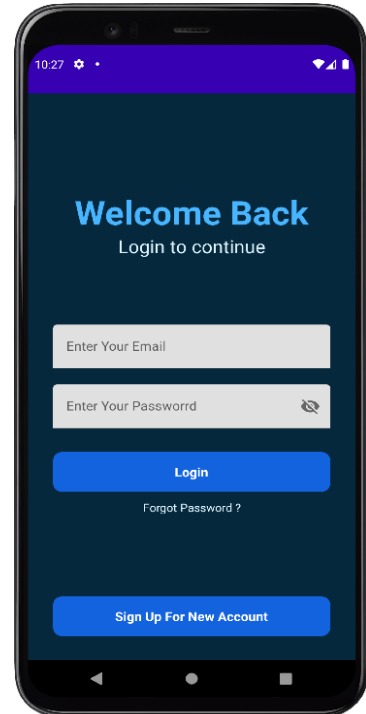


Figure 5.1.2: Login Page

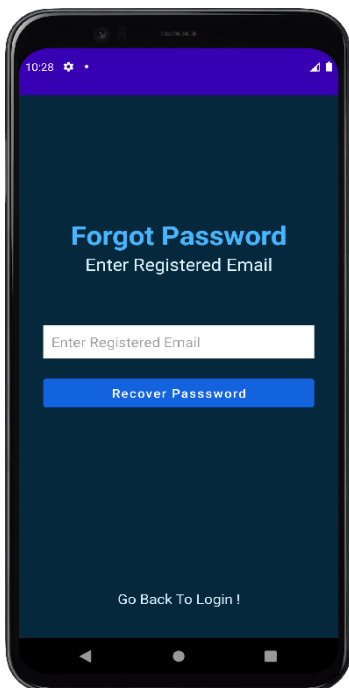


Figure 5.1.3: Forgot Password

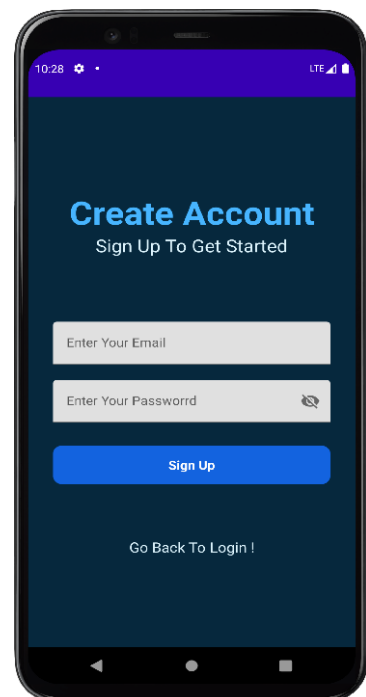


Figure 5.1.4: Sign Up

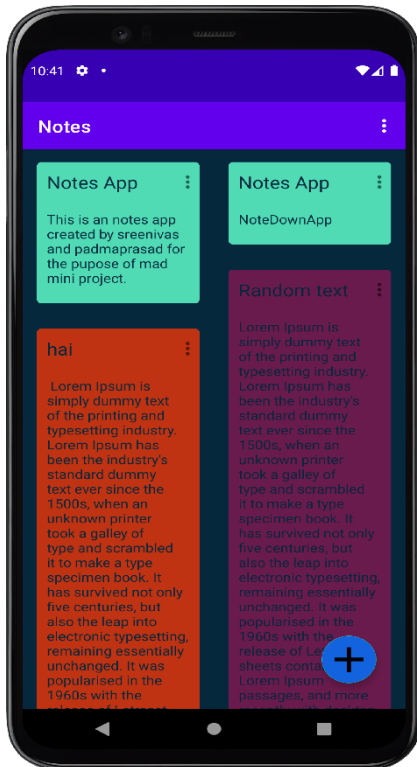


Figure 5.1.5: Notes Screen

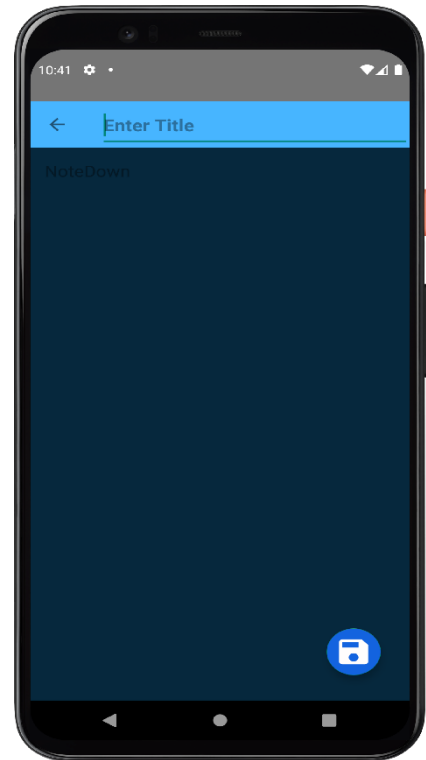


Figure 5.1.6: Create Notes

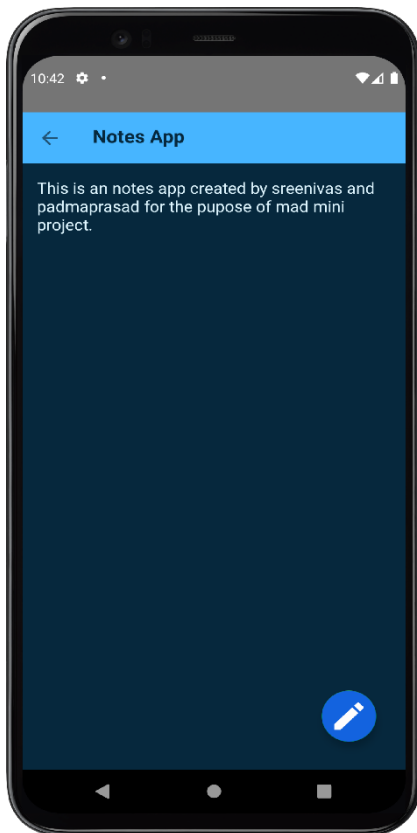


Figure 5.1.7: View Notes

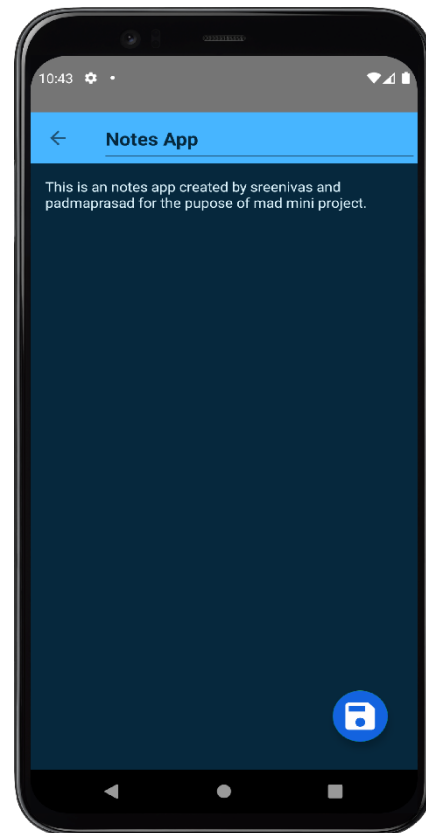


Figure 5.1.8: Edit Notes

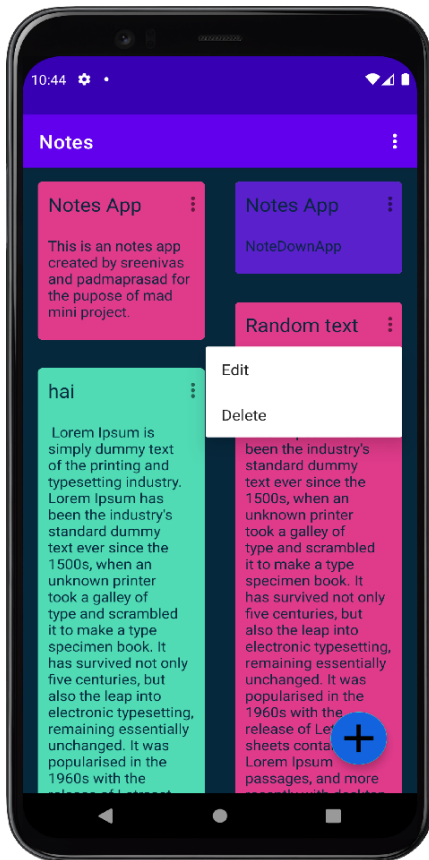


Figure 5.1.9: Options

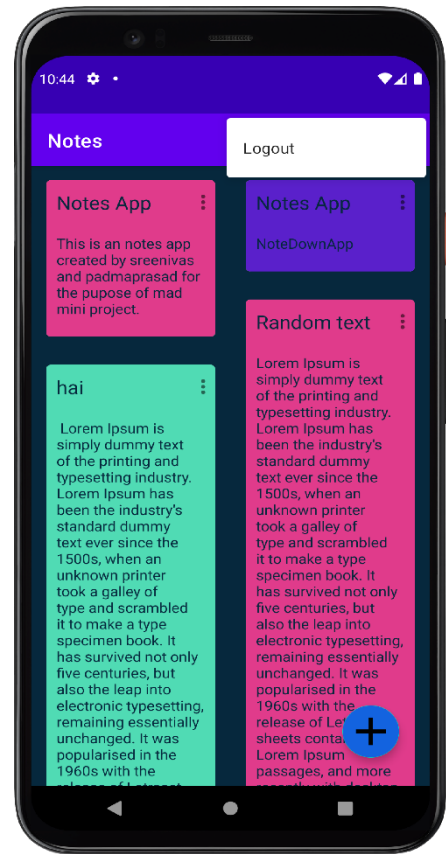


Figure 5.1.10: Logout

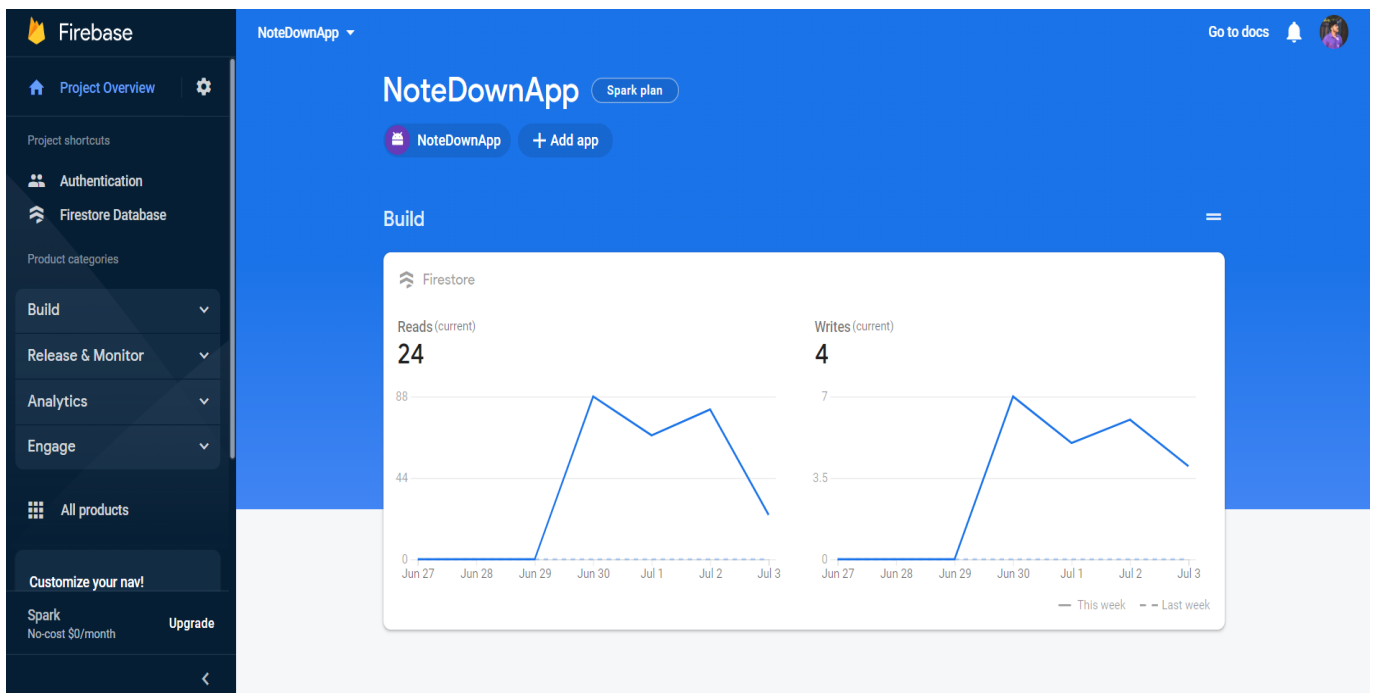


Figure 5.1.11:Firebase Console

CHAPTER 6

CONCLUSION & FUTURE ENHANCEMENTS

- The Note Down App is developed by using the Android studio to fully meet the objective of people.
- This is especially important as smartphones have become a common device for use by businesses as well as general customers.
- Some are focused more on just taking simple notes, while others come with more features and functionality that can turn your smartphone into a digital notebook.
- You can also use your smartphones for putting down ideas and other inspirations minute to minute.
- You can create and discuss content with your colleagues as well.

REFERENCES

The books and the other sources we referred while doing this project are as follows:

Referred Books

- Google Developer Training **Android Developer Fundamentals Course – Concept Reference** Google Developer Training Team, 2017

Websites Referred

- <https://www.youtube.com>
- developer.android.com