VISVESVARAYA TECHNOLOGIC AL UNIVERSITY

JNANA SANGAMA, BELAGAVI-590018, KARNATAKA

A Mini Project Report
on

OFFICE TASK MANAGEMENT SYSTEM

Submitted in partial fulfilment of the requirements for the VI Semester of degree of ***Bachelor of Engineering in Information Science and Engineering*** *of Visvesvaraya Technological University, Belagavi*

by

Aditya D	1RN19IS011
Akshay M L	1RN19IS017
Akshay P	1RN19IS018

Under the Guidance of

Mrs. Shwetha G N
Asst. Professor
Department of ISE

Department of Information Science and Engineering

RNS Institute of Technology

Dr. Vishnuvaradhana Road, Rajarajeshwari Nagar post,
Channasandra, Bengaluru-560098

2021-2022

RNS INSTITUTE OF TECHNOLOGY

Dr. Vishnuvaradhana Road, Rajarajeshwari Nagar post,
Channasandra,Bengaluru - 560098

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the project work entitled “Office Task Management System” has been successfully completed by **Aditya D (1RN19IS011), Akshay ML (1RN19IS017) and Akshay P (1RN19IS018),** bonafide students of **RNS Institute of Technology, Bengaluru** in partial fulfilment of the requirements for the award of degree in **Bachelor of Engineering in Information Science and Engineering** of **Visvesvaraya Technological University, Belgavi** during the academic year **2021-2022**. The mini project report has been approved as it satisfies the academic requirements with respect to the Mobile Applications Development laboratory.

Mrs.Shwetha G N
Mini Project Guide
Assistant Professor
Department of ISE

Dr. Suresh L
Professor and HOD
Department of ISE
RNSIT

Dr. M K Venkatesha
Principal
RNSIT

External Viva

Name of the Examiners **Signature with Date**

1. ______________________________ ______________________________

2. ______________________________ ______________________________

DECLARATION

We, **Aditya D [USN: 1RN19IS011], Akshay ML [USN: 1RN19IS017], Akshay P [USN: 1RN19IS018],** students of VI Semester BE, in Information Science and Engineering, RNS Institute of Technology hereby declare that the Project entitled "Office Task Management System" hasbeen carried out by us and submitted in partial fulfilment of the requirements for the *VI Semester of degree of* ***Bachelor of Engineering in Information Science and Engineering*** *of Visvesvaraya Technological University*, *Belgavi* during the academic year 2021-2022.

Place : Bengaluru

Date :

Aditya D	**(1RN19IS011)**
Akshay ML	**(1RN19IS017)**
Akshay P	**(1RN19IS018)**

ABSTRACT

Task management forms the foundation of the collaborative project management discipline. Done right, it provides organizations with an efficient workflow process that becomes the core vehicle for delivering all of their collaborative projects. Managing tasks is not easy, but, with a proper system in place and the right task sharing apps and tracking tools, professionals can ensure their projects run smoothly from beginning to end

The main aim of this project is to design an android-based mobile task management applicationfor the management of tasks in the organization. The system developed will reduce manual entry and avoid redundant data. It can maintain efficient daily, important tasks. Instead, the software can keep long and retrieve the information when needed.

ACKNOWLEDGMENT

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

We would like to profoundly thank **Management** of **RNS Institute of Technology** for providing such a healthy and vibrant environment to carry out the project work.

We would like to express our sincere thanks to our beloved Principal **Dr. M K Venkatesha** for his support and inspired me towards the attainment of knowledge.

We wish to place on record our words of gratitude to **Dr. Suresh L,** Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind our project work.

We would also like to thank our Mini Project guide Mrs Shwetha G N**,** Assistant Professor, Department of ISE, RNSIT, Bengaluru, for his valuable inputs, suggestions, time and guidance.

We would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped me to carry out the project work.

Place: Bengaluru

Date:

Aditya D
Akshay ML
Akshay P

Table of Contents

Certificate i

Declaration ii

Abstract iii

Acknowledgment iv

Table Of Contents v

List Of Figures vi

1. Introduction To Android 1

1.1 History 1

1.2 Android Versions 1

1.3 Architecture Of Android 2

1.4 Android Studio Installation 3

2. Introduction To Project 5

2.1 Overview Of The Project 5

2.1 Aim Of The Project 6

3. System Design 7

4. Implementation 13

5. Testing 21

6. Results 23

7. Conclusion And Future Enhancements 25

References 26

LIST OF FIGURES

Fig 3.1	System Design
Fig 5.1	Fill Register/Login Details
Fig 5.2	Login Successful
Fig 5.3	Add Task Screen
Fig 5.4	Database In Firebase Console
Fig 6.1	Registration
Fig 6.2	Login
Fig 6.3	Login Succesful/Home Screen
Fig 6.4	Authentication In Firebase
Fig 6.5	Add Task
Fig 6.6	Firebase Database of tasks Entered

Chapter 1

INTRODUCTION TO ANDROID

1.1 History

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically forAndroid development. It is available for download on Windows, macOS and Linux-based operating systems or as a subscription-based service in 2020.

It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the Google I/O conference.

It was in the n early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. On May 7, 2019, Kotlin replaced Java as Google's preferred language for Android app development. Java is still supported, as is C++.

1.1 Android Versions

Version	Release date	Version	Release date
Arctic Fox (2020.3.1)	July 2021	3.0	October 2017
4.2	May 2021	2.3	March 2017
4.1	Oct 2020	2.2	September 2106
4.0	May 2020	2.1	April 2016
3.6	February 2020	2.0	April 2016
3.5	August 2019	1.5	November 2015
3.4	April 2019	1.4	September 2015
3.3	January 2019	1.3	July 2015
3.2	September 2018	1.2	April 2015
3.1	March 2018	1.1	February 2015
3.0	October2017	1.0	December2014

1.2 Android Architecture

Android architecture contains different number of components to support any android device needs. Android software contains an open-source Linux Kernel having collection of number of C/C++ libraries which are exposed through an application framework services. Among all the components Linux Kernel provides main functionality of operating system functions to smartphones and Dalvik Virtual Machine (DVM) provide platform for running an android application.

The main components of android architecture are following: -

- Applications
- Application Framework
- Android Runtime
- Platform Libraries
- Linux Kernel

Applications – Applications is the top layer of android architecture. The pre-installed applications like home, contacts, camera, gallery etc and third-party applications downloaded from the play store like chat applications, games etc. will be installed on this layer only.

Application framework – Application Framework provides several important classes which are used to create an Android application. It provides a generic abstraction for hardware access and also helps in managing the user interface with application resources. Generally, it provides the services with the help of which we can create a particular class and make that class helpful for the Applications creation. It includes different types of services activity manager, notification manager, view system, package manager etc.

Application runtime – Android Runtime environment is one of the most important part of Android. It contains components like core libraries and the Dalvik virtual machine(DVM). Mainly, it provides the base for the application framework and powers our application with the help of the core libraries. The core libraries enable us to implement android applications using the standard JAVA or Kotlin programming languages.

Platform libraries – The Platform Libraries includes various C/C++ core libraries and Java based libraries such as Media, Graphics, Surface Manager, OpenGL etc. to provide a support for android development.

Media library provides support to play and record an audio and video formats.Surface manager responsible for managing access to the display subsystem.

SGL and OpenGL both cross-language, cross-platform application program interface (API) is used for 2D and 3D computer graphics.

SQLite provides database support and Free Type provides font support.

Linux Kernel – Linux Kernel is heart of the android architecture. It manages all the available drivers such as display drivers, camera drivers, Bluetooth drivers, audio drivers, memory drivers, etc. which are required during the runtime. The Linux Kernel will provide an abstraction layer between the device hardware and the other components of android architecture. It is responsiblefor management of memory, power, devices etc.

1.3 Android Studio Installation

Before downloading and installing Android Studio, the following requirements are essential. Operating System Version - Microsoft Windows 7/8/10 (32-bit or 64-bit).

Random Access Memory (RAM) - Minimum 4 GB RAM and 8 GB RAM recommended.

Free Disk Space - Minimum 2 GB and 4 GB recommended. Minimum Required JDK Version - Java Development Kit (JDK) 8.Minimum Screen Resolution - 1280 * 800.resolution

Step 1: To download the Android Studio, visit the official Android Studio website in your web browser.

Step 2: Click on the "Download Android Studio" option.

Step 3: Double click on the downloaded "Android Studio-ide.exe" file.

Step 4: "Android Studio Setup" will appear on the screen and click "Next" to proceed.

Step 5: Select the components that you want to install and click on the "Next" button.

Step 6: Now, browse the location where you want to install the Android Studio and click "Next" to proceed.

Step 7: Choose a start menu folder for the "Android Studio" shortcut and click the "Install" button to proceed.

Step 8: After the successful completion of the installation, click on the "Next" button.

Step 9: Click on the "Finish" button to proceed. Android studio welcome screen will appear on the screen.

Step 10: "Android Studio Setup Wizard" will appear on the screen with the welcome wizard.

Click on the "Next" button.

Step 11: Select (check) the "Standard" option if you are a beginner and do not have any idea about Android Studio. It will install the most common settings and options for you.Click "Next" to proceed.

Step 12: Now, select the user interface theme as you want. Then, click on the "Next" button.

Step 13: Now, click on the "Finish" button to download all the SDK components.

Step 14: After downloading all the necessary components, click on the "Finish" button.

Android Studio will be successfully installed in the system.

Chapter 2

INTRODUCTION TO PROJECT

2.1 Overview of the project

The office task management Application is an application for maintaining list of tasks in any organization. It provides minimal error in storing and maintaining important tasks efficiently. The main motive behind this software is to replace the traditional pen and register system.

This system developed will avoid redundant data that may be recorded by maintaining the tasks manually. This software can create tasks and notes, and maintain and retrieve the information when needed.

Office Task Management Application is developed in Android studio, java programming language is used for the field validation and also XML language for the transferring of data. System Uses recycler view to display tasks. Firebase for authentication and real time database storage. The application provides a medium where the user can enter the important tasks with deadlines for each day.

The different screens made available to the user are:

1. Registration Screen: Allows the user to register if not registered option to navigate to login screen is also present
2. Login Screen: The username and password of user is entered to log into the account and start adding tasks, users are updated in the firebase console
3. Home Screen: Lists the tasks added for the day ,will be blank if no task is added for the day.
4. Add Tasks: collects info about the task to be added, a floating action button is used which on clicking add tasks to the database.

2.2 Aim of the project

“Office Task Management System” is the practice of managing a series of tasks to completion, as defined by a project schedule. It is the process of progressing each task through its full life cycle, from the creation phase all the way through to completion and reporting.

Project managers follow systematic task management processes to manage all aspects of tasks. These processes include task sharing and assignment, timelines and shareable to-do lists, prioritization, budgeting, and status communications, among others.

The main aim is to design an android-based mobile attendance application for the management of office tasks in the organization.

The main objectives of this application are as follows:

1. Reduce admin work by integrating the details of all tasks into a single database.
2. Manual work for information retrieval on tasks becomes less as work becomes digitized.
3. It is also time-saving because manual work reduces.
4. There is less chance of error.
5. It eliminates duplicate data attendance entries.
6. Using Firebase ensures security of data and systematic storage of data.
7. Floating Action Button(FAB) makes it easy to add and delete tasks.
8. Recycler View helps to display the information of tasks in a systematic manner.

Chapter 3

SYSTEM DESIGN

One of the most difficult and interesting tasks for career engineers or computer scientists is to design entire system. A system design is an integrated set of interacting functional and non-functional parts and built by a single person.

According to the client requirements engineers start to do the project. As we are engineers, we know how to solve the large problems i.e. by breaking it into set of interacting problems. And again, those problems are decomposed into even smaller problem so, that can be solved easily.

Each module is responsible for part of entire system functionality. Then finally integrate all these functional and non-functional requirement unit problems into single system design to give the overview of the project. This system perspective gives the skeleton of a software product i.e. blueprint of the entire system.

All system perspective should have the following contents

- Inter-dependency among the modules and sub-modules.
- Co-opetition sub-system functionalities among the modules.
- For single unitary purpose whole system (i.e. sub-systems & modules) should terminate.

3.1 System Design

The System Design consists of the following modules

- **Register/Login Module:** Here the user can register himself/herself if already registered can login to the home screen.
- **Home Screen Module:** User can view existing tasks via the recycler view and has option to add or delete tasks.
- **Add/Delete Task Module:**Tasks can be added or deleted using the floating action button(FAB).
- **Firebase:** The user is authenticated and the data entered is encrypted and stored in the firebase console.

Overall module Collaboration and System Design:

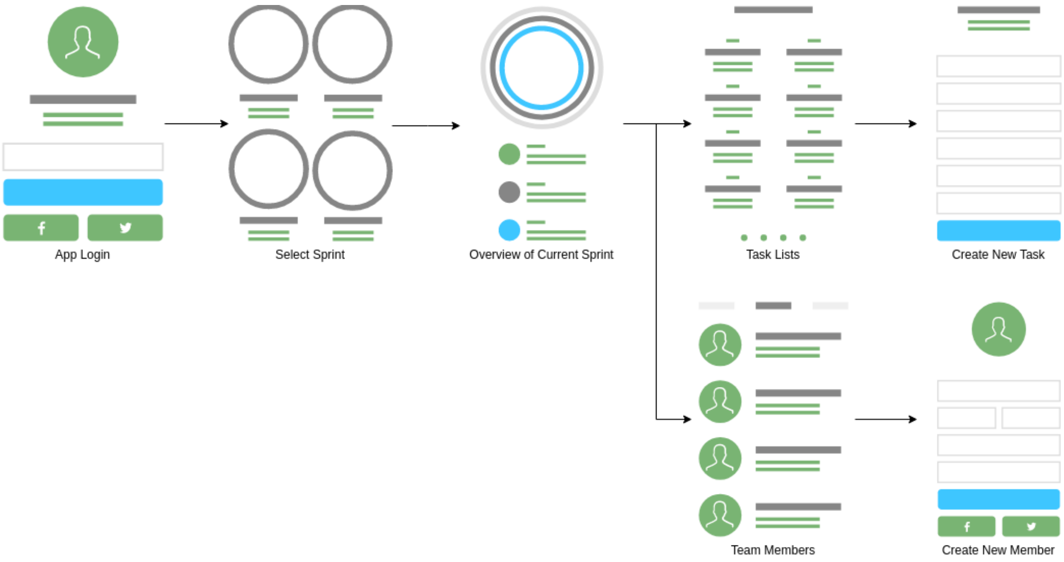


Figure 3.1: System design

3.2 Xml Files

The following codes are used in the project to create an effective User Interface design.

3.2.1 Activity_Main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:gravity="center"
    android:background="@drawable/backdesign">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">
```

```
  <TextView
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:text="Log In"
          android:textAppearance="?android:textAppearanceLarge"
          android:gravity="center"/>

      <EditText

          android:background="@drawable/edtdesign"
          android:id="@+id/email_login"
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:layout_margin="10dp"
          android:hint="Email:"
          android:minHeight="48dp"
          android:padding="10dp"
          />

      <EditText
          android:background="@drawable/edtdesign"
          android:padding="8dp"
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:layout_margin="10dp"
          android:hint="Password:"
          android:minHeight="48dp"
          android:inputType="textPassword"
          android:id="@+id/password_login"/>

      <androidx.appcompat.widget.AppCompatButton
          android:id="@+id/login_btn"
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:layout_margin="10dp"
          android:background="@drawable/edtdesign"
          android:text="Log In"
          android:textAppearance="?android:textAppearanceMedium">
      </androidx.appcompat.widget.AppCompatButton>

      <TextView
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:text="Don't Have Account? Sign Up here"
          android:gravity="center"
          android:padding="10dp"
          android:textAppearance="?android:textAppearanceSmall"
          android:id="@+id/signup_txt"
          android:textStyle="bold"/>

    </LinearLayout>

  </ScrollView>
</LinearLayout>
```

3.2.2 Activity_Home.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:orientation="vertical"
    tools:context=".HomeActivity">

  <include
      android:id="@+id/toolbar_home"
      layout="@layout/toolbar" />

  <androidx.coordinatorlayout.widget.CoordinatorLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent">

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

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fab_btn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|right|end"
        android:layout_margin="15dp"
        android:backgroundTint="@color/teal_200"
        android:src="@drawable/fab"
        app:fabSize="normal"
        tools:ignore="SpeakableTextPresentCheck,ImageContrastCheck" />
  </androidx.coordinatorlayout.widget.CoordinatorLayout>


</LinearLayout>
```

3.2.3 Activity_Register.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:orientation="vertical"
    android:gravity="center"
    android:background="@drawable/backdesign"
    tools:context=".RegistrationActivity">
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">
            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="Registration"
                android:textAppearance="?android:textAppearanceLarge"
                android:gravity="center"/>

            <EditText

                android:background="@drawable/edtdesign"
                android:id="@+id/email_reg"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_margin="10dp"
                android:hint="Email:"
                android:minHeight="48dp"
                android:padding="10dp"
                />

            <EditText
                android:background="@drawable/edtdesign"
                android:padding="8dp"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_margin="10dp"
                android:hint="Password:"
                android:minHeight="48dp"
                android:inputType="textPassword"
                android:id="@+id/password_reg"/>

            <androidx.appcompat.widget.AppCompatButton
                android:id="@+id/reg_btn"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_margin="10dp"
                android:background="@drawable/edtdesign"
                android:text="Register"
                android:textAppearance="?android:textAppearanceMedium">
            </androidx.appcompat.widget.AppCompatButton>
```

```
  <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Already Have Account? Log in here"
            android:gravity="center"
            android:padding="10dp"
            android:textAppearance="?android:textAppearanceSmall"
            android:id="@+id/login_txt"
            android:textStyle="bold"/>

     </LinearLayout>

   </ScrollView>

</LinearLayout>
```

Chapter 4

IMPLEMENTATION

The following java codes are used in the project for correct execution of the app.

4.1 Main_activity.java

```
package com.example.officetaskmanagementapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
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.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
public class MainActivity extends AppCompatActivity {
    private TextView signup;
    private EditText email;
    private EditText pass;
    private Button btnLogin;
    //Firebase..
    private FirebaseAuth mAuth;
    private ProgressDialog mDialog;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mAuth = FirebaseAuth.getInstance();
        if(mAuth.getCurrentUser() != null){
            startActivity(new Intent(getApplicationContext(),HomeActivity.class));
        }
        mDialog = new ProgressDialog(this);
        signup = findViewById(R.id.signup_txt);
        email = findViewById(R.id.email_login);
        pass = findViewById(R.id.password_login);
        btnLogin = findViewById(R.id.login_btn);
        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String mEmail = email.getText().toString().trim();
                String mPass = pass.getText().toString().trim();
                if(TextUtils.isEmpty(mEmail)){
                    email.setError("Required Field..");
                    return;
                }
```

```
 if(TextUtils.isEmpty(mPass)){
                pass.setError("Required Field..");
                return;
            }
            mDialog.setMessage("Processing..");
            mDialog.show();
            mAuth.signInWithEmailAndPassword(mEmail,mPass).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if(task.isSuccessful()){
                        Toast.makeText(getApplicationContext(), "Login Successful",
Toast.LENGTH_SHORT).show();
                        startActivity(new Intent(getApplicationContext(),HomeActivity.class));
                    }else {
                        Toast.makeText(getApplicationContext(), "Problem", Toast.LENGTH_SHORT).show();
                    }
                    mDialog.dismiss();

                }
            });

        }
    });

    signup.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent(getApplicationContext(), RegistrationActivity.class));


        }
    });
  }
}
```

4.2 Home_activity.java

```
package com.example.officetaskmanagementapp;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.content.Intent;
import android.os.Bundle;
import android.os.PersistableBundle;
import android.text.TextUtils;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import android.widget.Toolbar;
import com.example.officetaskmanagementapp.Model.Data;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import java.text.DateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.Objects;
public class HomeActivity extends AppCompatActivity {
    private Toolbar toolbar;
    private FloatingActionButton fabBtn;
    //Firebase
    private DatabaseReference mDatabase;
    private FirebaseAuth mAuth;
    //Recycler..
    private RecyclerView recyclerView;
    ArrayList<Data> list;
    MyAdapter adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);
        androidx.appcompat.widget.Toolbar toolbar = findViewById(R.id.toolbar_home);
        setSupportActionBar(toolbar);
        Objects.requireNonNull(getSupportActionBar()).setTitle("Office Task Management App");
        mAuth = FirebaseAuth.getInstance();
        FirebaseUser mUser = mAuth.getCurrentUser();
        String uId = mUser.getUid();
        mDatabase = FirebaseDatabase.getInstance().getReference().child("TaskNote").child(uId);
        mDatabase.keepSynced(true);
        //Recycler..
        recyclerView = findViewById(R.id.recycler);
        LinearLayoutManager layoutManager = new LinearLayoutManager(this);
        layoutManager.setReverseLayout(true);
        layoutManager.setStackFromEnd(true);
        recyclerView.setHasFixedSize(true);
        recyclerView.setLayoutManager(layoutManager);
        fabBtn = findViewById(R.id.fab_btn);
        fabBtn.setOnClickListener(new View.OnClickListener() {
            @Override
```

```
public void onClick(View v) {
                AlertDialog.Builder myDialog = new AlertDialog.Builder(HomeActivity.this);
                LayoutInflater inflater = LayoutInflater.from(HomeActivity.this);
                View myview = inflater.inflate(R.layout.customimputfield,null);
                myDialog.setView(myview);
                AlertDialog dialog = myDialog.create();
                EditText title = myview.findViewById(R.id.edt_title);
                EditText note = myview.findViewById(R.id.edt_note);
                Button btnSave= myview.findViewById(R.id.btn_save);
                btnSave.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        String mTitle = title.getText().toString().trim();
                        String mNote = note.getText().toString().trim();
                        if(TextUtils.isEmpty(mTitle)){
                            title.setError("Required Field..");
                            return;
                        }
                        if(TextUtils.isEmpty(mNote)){
                            note.setError("Required Field..");
                            return;
                        }
                        String id = mDatabase.push().getKey();
                        String datee = DateFormat.getDateInstance().format(new Date());
                        Data data = new Data(mTitle,mNote,datee,id);
                        mDatabase.child(id).setValue(data);
                        Toast.makeText(getApplicationContext(), "Data Inserted",
Toast.LENGTH_SHORT).show();
                        dialog.dismiss();
                    }
                });
                dialog.show();
            }
        });
    }
    @Override
    protected void onStart() {
        super.onStart();
        /*FirebaseRecyclerAdapter<Data,MyViewHolder>adapter = new FirebaseRecyclerAdapter<Data,
MyViewHolder>() {
            @Override
            protected void onBindViewHolder(@NonNull MyViewHolder holder, int position, @NonNull
Data model) {
            }
            @NonNull
            @Override
            public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
                return null;
            }
        };*/
        list = new ArrayList<>();
        recyclerView.setLayoutManager(new LinearLayoutManager(this));
        adapter = new MyAdapter(this, list);
        recyclerView.setAdapter(adapter);
        mDatabase.addValueEventListener(new ValueEventListener() {
```

```
@Override
public void onDataChange(@NonNull DataSnapshot snapshot) {
                for (DataSnapshot dataSnapshot: snapshot.getChildren()){
                    Data data = dataSnapshot.getValue(Data.class);
                    list.add(data);
                    adapter.notifyDataSetChanged();
                }
            }
            @Override
            public void onCancelled(@NonNull DatabaseError error) {
            }
        });
    }
    public static class MyViewHolder extends RecyclerView.ViewHolder{
        View myview;
        public MyViewHolder(@NonNull View itemView) {
            super(itemView);
            myview=itemView;
        }
        public void setTitle(String title){
            TextView mTitle=myview.findViewById(R.id.title);
            mTitle.setText(title);
        }
        public void setNote(String note){
            TextView mNote = myview.findViewById(R.id.note);
            mNote.setText(note);
        }
        public void setDate(String date){
            TextView mDate=myview.findViewById(R.id.date);
            mDate.setText(date);
        }
    }
   /* public void updateData(){
        AlertDialog.Builder mydialog = new AlertDialog.Builder(HomeActivity.this);
        LayoutInflater inflater = LayoutInflater.from(HomeActivity.this);
        View myview = inflater.inflate(R.layout.updateinputfield,null);
        mydialog.setView(myview);
        AlertDialog dialog = mydialog.create();
        dialog.show();
    }*/
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.mainmenu,menu);
        return super.onCreateOptionsMenu(menu);
    }
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case R.id.logout:
                mAuth.signOut();
                startActivity(new Intent(getApplicationContext(),MainActivity.class));
                break;

        }
        return super.onOptionsItemSelected(item);
```

```
}
}
```

4.3 Register_activity.java

```
package com.example.officetaskmanagementapp;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
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.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
public class RegistrationActivity extends AppCompatActivity {
    private EditText email;
    private EditText pass;
    private Button btnReg;
    private TextView login_txt;
    private FirebaseAuth mAuth;
    private ProgressDialog mDialog;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_registration);
        mAuth = FirebaseAuth.getInstance();
        mDialog= new ProgressDialog(this);
        email = findViewById(R.id.email_reg);
        pass = findViewById(R.id.password_reg);
        btnReg = findViewById(R.id.reg_btn);
        login_txt = findViewById(R.id.login_txt);
        login_txt.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(getApplicationContext(),MainActivity.class));
            }
        });
        btnReg.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String mEmail=email.getText().toString().trim();
                String mPass=pass.getText().toString().trim();
                if(TextUtils.isEmpty(mEmail)){
                    email.setError("Required Field..");
                    return;
                }
                if(TextUtils.isEmpty(mPass)){
                    pass.setError("Required Field..");
```

```
 return;
            }
          mDialog.setMessage("Processing..");
            mDialog.show();
            mAuth.createUserWithEmailAndPassword(mEmail,mPass).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if(task.isSuccessful()){
                        Toast.makeText(getApplicationContext(), "Successful", Toast.LENGTH_SHORT).show();
                        startActivity(new Intent(getApplicationContext(),HomeActivity.class));
                    }else{
                        Toast.makeText(getApplicationContext(), "Problem", Toast.LENGTH_SHORT).show();
                    }
                    mDialog.dismiss();
                }
            });
        }
    });
}}
```

4.4 Adapter_activity.java

```
package com.example.officetaskmanagementapp;
import android.content.Context;
import android.view.LayoutInflater;
import android.view.TextureView;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.example.officetaskmanagementapp.Model.Data;
import java.util.ArrayList;
public class MyAdapter extends RecyclerView.Adapter<MyAdapter.MyViewHolder> {
    Context context;
    ArrayList<Data> list;
    public MyAdapter(Context context, ArrayList<Data> list) {
        this.context = context;
        this.list = list;
    }
    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View v = LayoutInflater.from(context).inflate(R.layout.item_data,parent,false);
        return new MyViewHolder(v);
    }
    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {
        Data data = list.get(position);
        holder.date.setText(data.getDate());
        holder.title.setText(data.getTitle());
        holder.note.setText(data.getNote());
}
    @Override
    public int getItemCount() {
        return list.size(); }
    public static class MyViewHolder extends RecyclerView.ViewHolder{
        TextView date, title, note;
        public MyViewHolder(@NonNull View itemView) {
            super(itemView);
            date = itemView.findViewById(R.id.date);
            title = itemView.findViewById(R.id.title);
            note = itemView.findViewById(R.id.note); }
    }
}
```

4.5 Data_activity.java

```
package com.example.officetaskmanagementapp.Model;
import com.google.firebase.database.IgnoreExtraProperties;
@IgnoreExtraProperties
public class Data {
    private String title;
    private String note;
    private String date;
    private String id;
    public Data(){
    }
    public Data(String title, String note, String date, String id) {
        this.title = title;
        this.note = note;
        this.date = date;
        this.id = id;
    }
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
    public String getNote() {
        return note;
    }
    public void setNote(String note) {
        this.note = note;
    }
    public String getDate() {
        return date;
    }
    public void setDate(String date) {
        this.date = date;
    }
    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
}
```

Chapter 5

TESTING

Testing is the process used to help identify the correctness, completeness, security and quality of the developed computer software. Testing is the process of technical investigation and includes the process of executing a program or application with the intent of finding errors.

Testing Login Screen

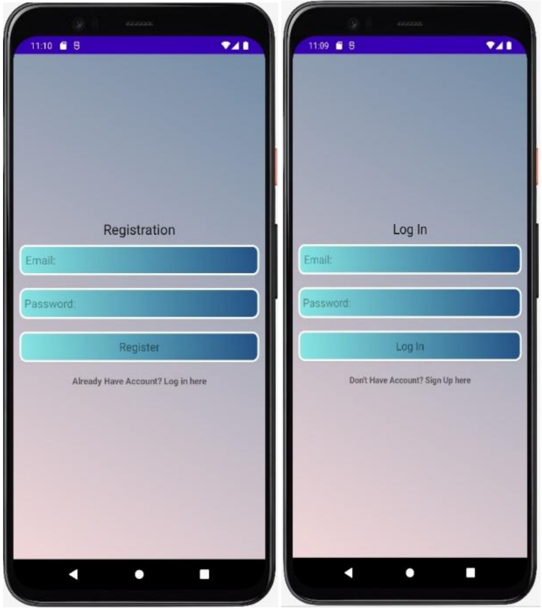
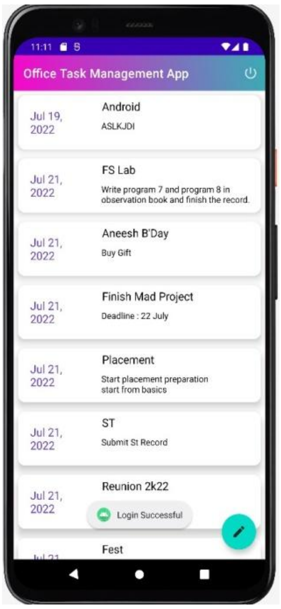


Fig 5.1 Fill Register/Login Details



Office Task Management App

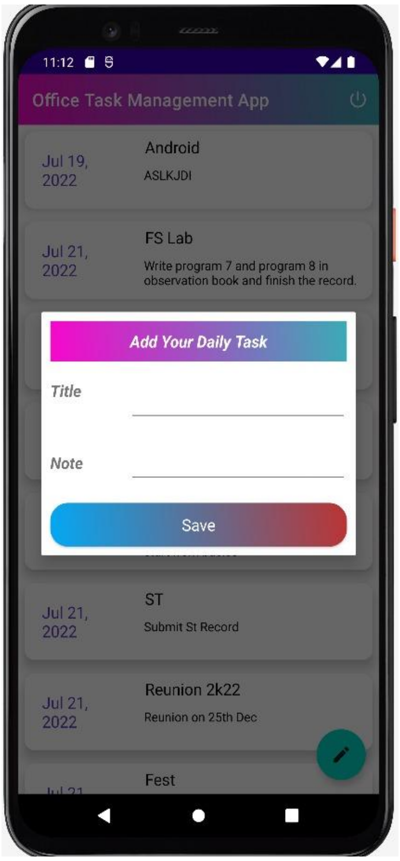
Jul 19, 2022	Android ASLKJDI
Jul 21, 2022	FS Lab Write program 7 and program 8 in observation book and finish the record.
Jul 21, 2022	Aneesh B'Day Buy Gift
Jul 21, 2022	Finish Mad Project Deadline : 22 July
Jul 21, 2022	Placement Start placement preparation start from basics
Jul 21, 2022	ST Submit St Record
Jul 21, 2022	Reunion 2k22
	Fest

Login Successful

Fig 5.2 Login Successful

Fig 5.1 shows the error message Login Unsuccessful if credentials are wrong and Fig 5.2 shows login successful when credentials are correct.

Testing Add Task Screen



Jul 19, 2022 Android
ASLKJDI

Jul 21, 2022 FS Lab
Write program 7 and program 8 in observation book and finish the record.

Add Your Daily Task

Title

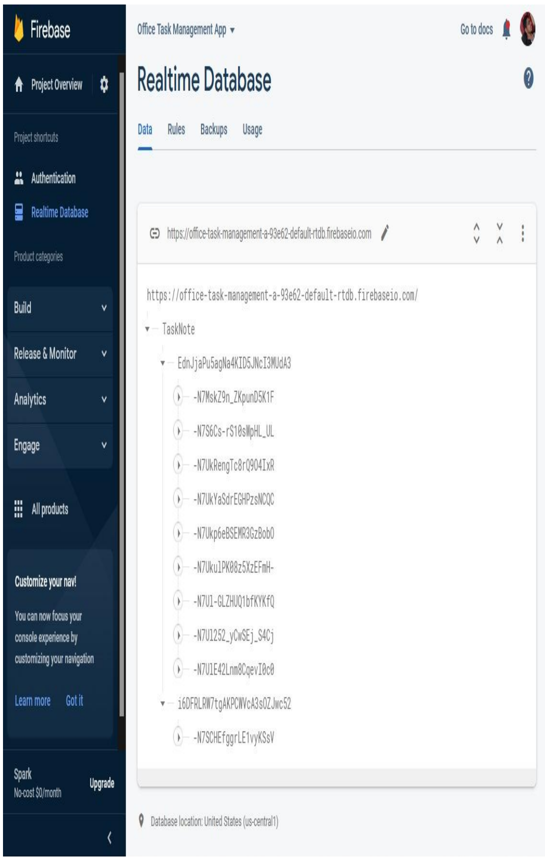
Note

Jul 21, 2022 ST
Submit St Record

Jul 21, 2022 Reunion 2k22
Reunion on 25th Dec

Jul 21 Fest

Fig 5.3 Add Task Screen



```
https://office-task-management-a-93e62-default-rtdb.firebaseio.com

https://office-task-management-a-93e62-default-rtdb.firebaseio.com/
▾ — TaskNote
    ▾ — EdnJjaPu5agNa4KID5JNcI3MUdA3
        ▸ — -N7MskZ9n_ZKpunD5K1F
        ▸ — -N7S6Cs-rS10sWpHL_UL
        ▸ — -N7UkRengTc8rQ904IxR
        ▸ — -N7UkYaSdrEGHPzsNCQC
        ▸ — -N7Ukp6eBSEMR3GzBob0
        ▸ — -N7UkulPK08z5XzEFmH-
        ▸ — -N7Ul-GLZHUQ1bfKYKfQ
        ▸ — -N7Ul252_yCwSEj_S4Cj
        ▸ — -N7UlE42Lnm8CqevI0c0
    ▾ — i6DFRLRW7tgAKPCWVcA3sOZJwc52
        ▸ — -N7SCHEfggrLE1vyKSsV
```

Database location: United States (us-central1)

Fig 5.4 Database in Firebase Console

The Fig 5.3 shows that empty task cannot be added Fig 5.4 shows that empty task is not updated in the **Firebase Database**.

Chapter 6

RESULTS

The following screens appear in sequence to the user on clicking of respective buttons and texts.

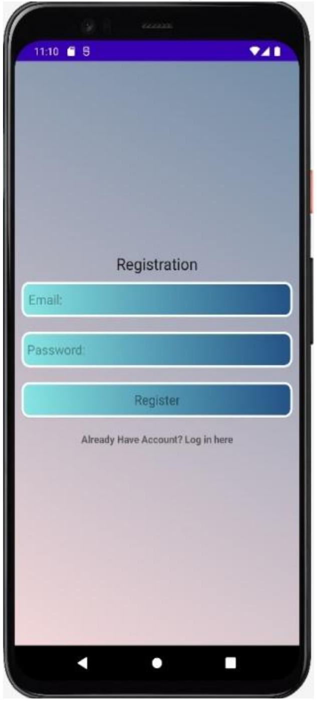


Fig 6.1 Registeration

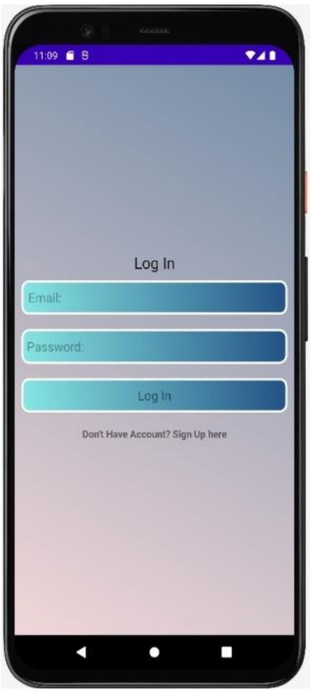
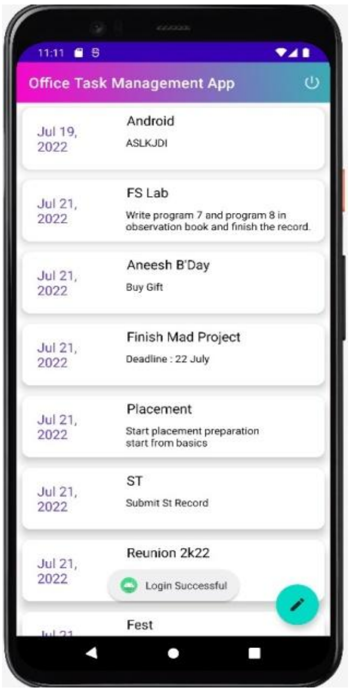


Fig 6.2 Login



Jul 19, 2022 — Android
ASLKJDI

Jul 21, 2022 — FS Lab
Write program 7 and program 8 in observation book and finish the record.

Jul 21, 2022 — Aneesh B'Day
Buy Gift

Jul 21, 2022 — Finish Mad Project
Deadline : 22 July

Jul 21, 2022 — Placement
Start placement preparation start from basics

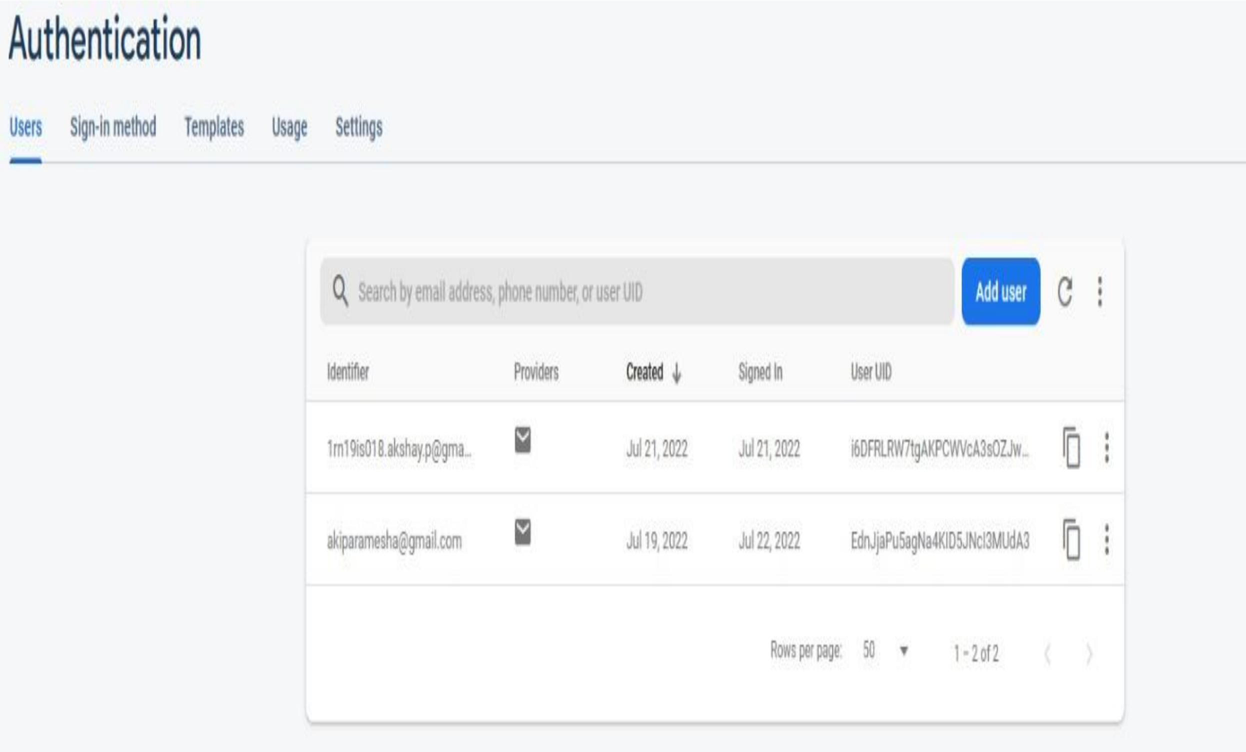
Jul 21, 2022 — ST
Submit St Record

Jul 21, 2022 — Reunion 2k22

Login Successful

Fest

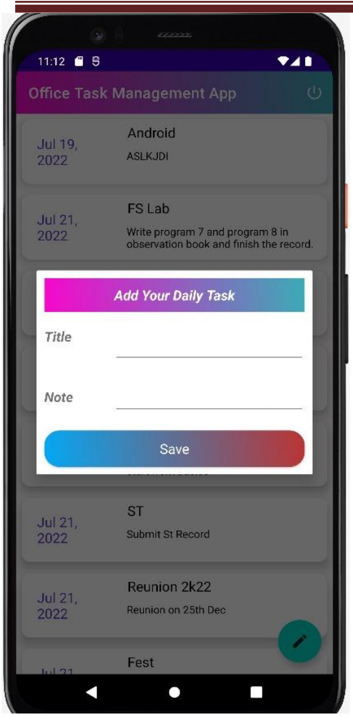
Fig 6.3 Login Successful



Identifier	Providers	Created ↓	Signed In	User UID
1rn19is018.akshay.p@gma...	✉	Jul 21, 2022	Jul 21, 2022	i6DFRLRW7tgAKPCWVcA3sOZJw...
akiparamesha@gmail.com	✉	Jul 19, 2022	Jul 22, 2022	EdnJjaPu5agNa4KID5JNcI3MUdA3

Rows per page: 50 ▾ 1 - 2 of 2

Fig 6.4 Authentication In Firebase



Jul 19, 2022 Android
ASLKJDI

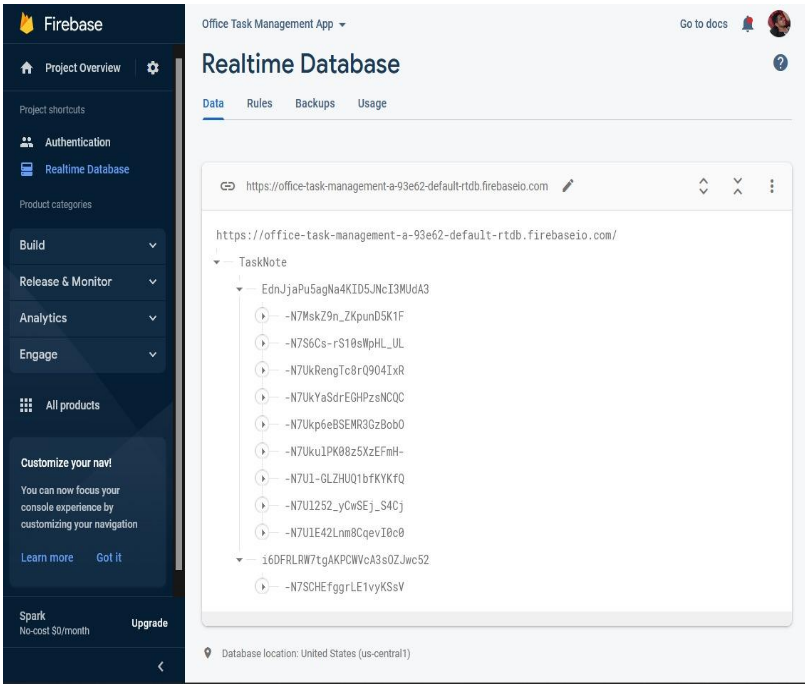
Jul 21, 2022 FS Lab
Write program 7 and program 8 in observation book and finish the record.

Jul 21, 2022 ST
Submit St Record

Jul 21, 2022 Reunion 2k22
Reunion on 25th Dec

Fest

Fig 6.5 Add Task



https://office-task-management-a-93e62-default-rtdb.firebaseio.com

```
https://office-task-management-a-93e62-default-rtdb.firebaseio.com/
▾ — TaskNote
    ▾ — EdnJjaPu5agNa4KID5JNcI3MUdA3
        ▸ — -N7MskZ9n_ZKpunD5K1F
        ▸ — -N7S6Cs-rS10sWpHL_UL
        ▸ — -N7UkRengTc8rQ904IxR
        ▸ — -N7UkYaSdrEGHPzsNCQC
        ▸ — -N7Ukp6eBSEMR3GzBob0
        ▸ — -N7UkulPK08z5XzEFmH-
        ▸ — -N7Ul-GLZHUQ1bfKYKfQ
        ▸ — -N7Ul252_yCwSEj_S4Cj
        ▸ — -N7UlE42Lnm8CqevI0c0
    ▾ — i6DFRLRW7tgAKPCWVcA3sOZJwc52
        ▸ — -N7SCHEfggrLE1vyKSsV
```

Database location: United States (us-central1)

Fig 6.6 Firebase Database of tasks entered

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusion

The Easy Attendance app simplifies the process of taking attendance in the classroom. It has a very clean and elegant UI which is teacher-friendly.
It allows the Teacher or Administrator to:

- Create a Class with Class name, subject name and a theme
- Add some students to the class
- Edit student details and also call them
- Mark the Attendance for the day and submit.
- View the attendance reports for each day.

This system developed will reduce manual work and avoid redundant data. This app simplifies the process of taking attendance in the classroom. It has a very clean and elegant UI. The software can maintain information and retrieve data whenever needed.

It can be used as a base for creating similar applications for tracking attendance in offices or the workplace. It can be also integrated into the healthcare sector to keep track of nurse-to-patient visits by streamlining the time entry, time approval and management processes.

7.2 Future Enhancement

The future enhancements possible to be carried out in order to improve the applicability and user friendliness are:

- Provision of Multilingual entry to support users with varying background and language
- Addition of extra graphics to make the application more user friendly and easily understandable.
- Provision to create and manage versions of documents online.
- Online Crime Reporting System can also be included as a part of this application, with registered entry for general public.
- Provision to enable citizens to register complaints online and receive an electronic acknowledgment

REFERENCES

[1] https://android-developers.googleblog.com/

[2] https://en.wikipedia.org/wiki/Android_version_history

[3] https://developer.android.com/

[4] https://developer.android.com/reference/android/widget/TextView

[5] https://developer.android.com/training/basics/firstapp

[6] https://firebase.google.com/

[7] https://www.sitepoint.com/starting-android-development-creating-todo-app/

[8] https://www.geeksforgeeks.org/