StudEasy

A Mini Project Report Submitted by

Ardra Madhu (4NM20CS039)
Disha Shetty (4NM20CS066)
Jayashree (4NM20CS079)

UNDER THE GUIDANCE OF

Dr. Anisha P. Rodrigues
Assistant Professor
Department of Computer Science and Engineering

in partial fulfilment of the requirements for the award of the Degree of

Bachelor of Engineering in
Computer Science & Engineering

from

Visvesvaraya Technological University,
Belagavi

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

N.M.A.M. INSTITUTE OF TECHNOLOGY

(An Autonomous Institution under VTU, Belagavi) (AICTE approved, NBA Accredited, ISO 9001:2008 Certified) NITTE -574 110, Udupi District, KARNATAKA.

NITTE EDUCATION TRUST | N.M.A.M. INSTITUTE OF TECHNOLOGY
(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi)
Nitte – 574 110, Karnataka, India
(ISO 9001:2015 Certified), Accredited with 'A' Grade by NAAC
☎: 08258 - 281039 - 281263, Fax: 08258 - 281265
Department of Computer Science and Engineering
B.E. CSE Program Accredited by NBA, New Delhi from 1-7-2018 to 30-6-2021

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

"StudEasy"is a bonafide work carried out byArdra Madhu (4nm20cs039), Disha Shetty(4nm20cs066), Jayashree(4nm20cs079), in partial fulfilment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering prescribed by Visvesvaraya Technological University, Belagavi during the year 2022-2023.

It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report.

The Mini project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the Bachelor of Engineering Degree.

6/1/2023
Signature of Guide
(Dr. Anisha P Rodrigues)

Jyothi Shetty
Signature of HOD

ACKNOWLEDGEMENT

We believe that our project will be complete only after we thank the people who have contributed to make this project successful.

First and foremost, we express our deep sense of gratitude and indebtedness to our guide **Dr. Anisha P. Rodrigues**, Designation, Department of Computer Science and Engineering, for his inspiring guidance, constant encouragement, support and suggestions for improvement during the course of our project.

We sincerely thank **Dr. Jyothi Shetty**, Head of Department of Computer Science and Engineering, Nitte Mahalinga Adyantaya Memorial Institute of Technology, Nitte.

Our sincere thanks to our beloved principal, **Dr. Niranjan N. Chiplunkar** for giving us an opportunity to carry out our project work at our college and providing us with all the needed facilities.

We also thank all those who have supported us throughout the entire duration of our project.

Finally, we thank the staff members of the Department of Computer Science and Engineering and all our friends for their honest opinions and suggestions throughout the course of our project.

Ardra Madhu (4NM20CS039)
Disha Shetty (4NM20CS066)
Jayashree (4NM20CS079)

ABSTRACT

A student companion app can be a useful tool for students of all ages and academic levels. A companion app can provide students with reminders, incentives, and rewards for achieving academic goals. This can help increase motivation and engagement, leading to improved performance. We are hence building an Android application for a student companion app. A companion app can be customized to suit the specific needs of individual students. This can help students learn in a way that works best for them, and can help them achieve their academic goals more effectively. Our app will cover all such issues and make the entire process as relevant user friendly as possible.

The required data is stored and retrieved using firebase database. The app is coded in Java using Android Studio.

Table of Contents:

CHAPTER 1

INTRODUCTION

1.1 Scope

StudEasy is a student-friendly application. The scope of this application is to offer a comprehensive solution to students who want to manage their academic and extracurricular activities efficiently. It provides a unique range of features that allow students to view their attendance status, timetable, events, bus timings, faculty cabins, important contact details, and even current CGPA.

1.2 Importance

StudEasy helps students stay organized and on top of their academic and extracurricular activities. The app provides easy access to all the information that students need to manage their schedules and keep track of their attendance. Basically, it is Parent-Login and Moodle clubbed into an app but on a much smaller scale.

1.3 Objective:

The primary objective of the StudEasy Android app is to help students manage their academic and extracurricular activities efficiently. Instead of logging into Moodle or parent-login all the time students can use StudEasy to access information quickly. The app aims to provide a seamless and convenient experience for students. The app also aims to improve communication between students, faculty, and administration by providing important contact details and Faculty cabins of all the CSE lectures. StudEasy lets students check their attendance status, view their timetable, and get informed about upcoming classes and events. Ultimately, the objective of the StudEasy app is to contribute to the academic success of students by making their lives more comfortable and organized.

CHAPTER 3

SYSTEM REQUIREMENT AND SPECIFICATION

3.1 Introduction

Requirements serve as a critical aspect during the early stages of system development. They act as a set of specifications that outline what the system should be able to do or constraints that must be placed on the system. These requirements may range from a high-level description of system functionality to a more detailed specification of system behaviour, properties, or constraints.

The primary goal of the requirement analysis phase is to generate a requirement specification document that details the results of the analysis phase. This document serves as a communication tool for stakeholders, including software engineers, system designers, and end-users.

System requirements are more detailed descriptions of user requirements and serve as the basis for a contract for implementing the system. A complete and consistent specification of the whole system is essential for the successful implementation of the system. They are used as the starting point for system design, and ideally, they should focus on what the system should do, rather than how it should be implemented.

However, natural language is often used to write system requirements, which can lead to several problems, especially when it comes to detailed specifications. The ambiguity of natural language can result in misunderstandings as the specification relies on both the reader and writer to have a common understanding of the same words for the same concept. Additionally, natural language is over-flexible, making it difficult to determine when requirements are distinct or the same.

3.2 Functional Requirements

The functional requirements are the statement of services the system should provide, how the system reacts to particular inputs, and how the system should behave in a particular situation. It describes the functionality that the system provides.
Our app requires:
I) Active internet connection.
II) A Firebase console to store the data

3.3 User Requirements

Student requires an active internet connection to use the app.

3.4 Software Requirements

1. Operating System: Windows 7/8/10/11 (32-bit or 64-bit)
2. Android SDK
3. Android Studio
4. Firebase

3.4.1 Android SDK

The Android SDK provides you the API libraries and developer tools necessary to build, test, and debug apps for Android. The ADT bundle includes the essential Android SDK components and a version of the Eclipse IDE with built-in Android Developer Tools to streamline the Android app development. ADT bundle consists of following components for developing the application II. Eclipse ADT plugin.

- Android SDK Tools
- Android Platform-tools
- The latest Android platform
- The latest Android system image for the emulator

3.4.2 Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on Jet Brains IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS, and Linux-based operating systems. Android Studio is the successor to the Eclipse Android Development Tools (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 early access preview stage starting from version 0.1 in May 2013, then entered the 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. The current stable version is 4.2.1, which was released in May 2021.

3.4.3 Firebase

Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014.As of October 2018, the Firebase platform has 18 products, which are used by 1.5 million apps. Firebase provides a real-time database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud. Firebase Storage provides secure file uploads and downloads for Firebase apps, regardless of network quality. The developer can use it to store images, audio, video, or other user-generated content. Firebase Storage is backed by Google Cloud Storage.

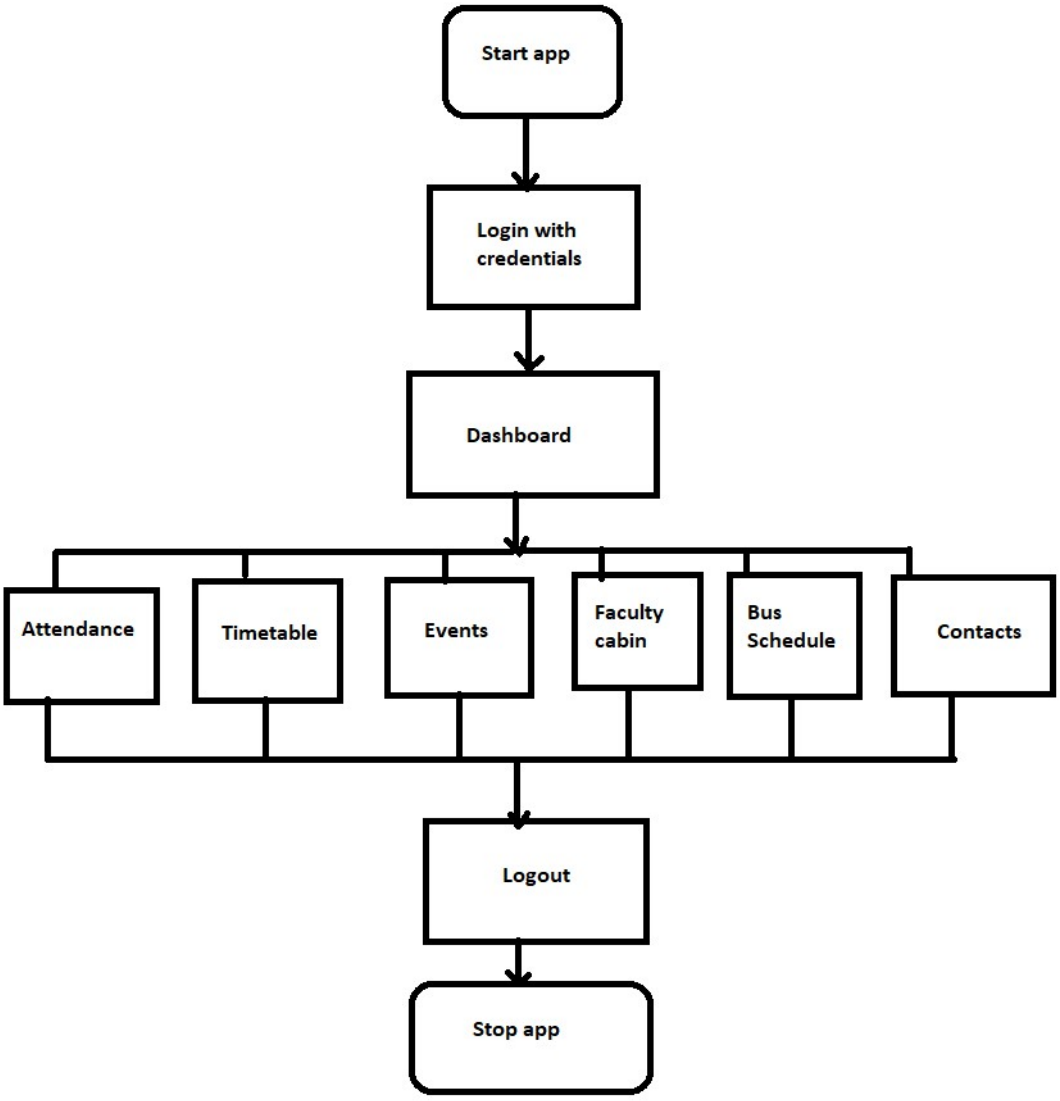
3.5 HARDWARE REQUIREMENTS

1. Minimum 4 GB RAM (8GB recommended).
2. 5GB free disk space
3. USB 2.0 or higher
4. Android Device

CHAPTER 4

SYSTEM DESIGN

4.1 Dataflow Diagram



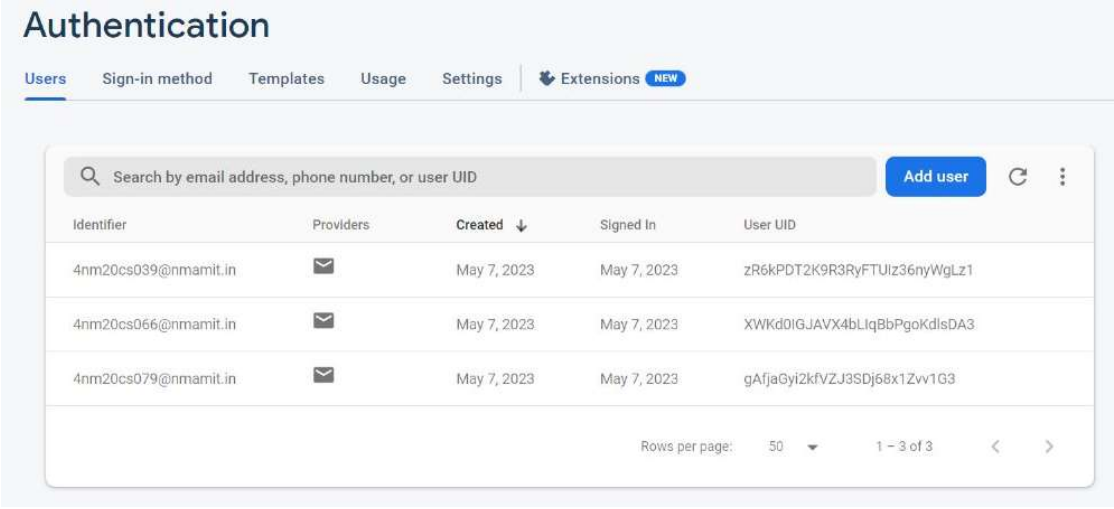
STUDENT

- Students can log in using their name, USN, and email ID along with a password to the app.
- Dashboard is displayed with Student name, USN, and current cgpa, along with other features.
- This app provides a range of features, including student attendance, timetable, events, bus timings, faculty cabins, contacts, and more.
- Students quickly check their attendance status, view their timetable, and access bus timings, Faculty cabins of all the CSE lectures, and important contact details.
- Lastly there is a logout button to logout from the application.

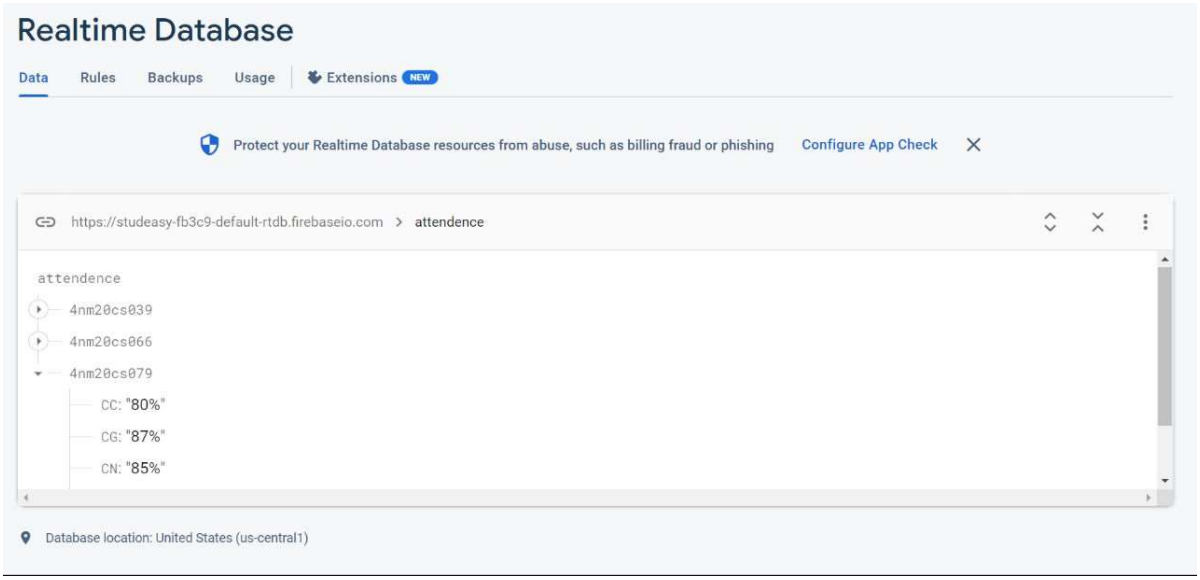
4.3 Database Design

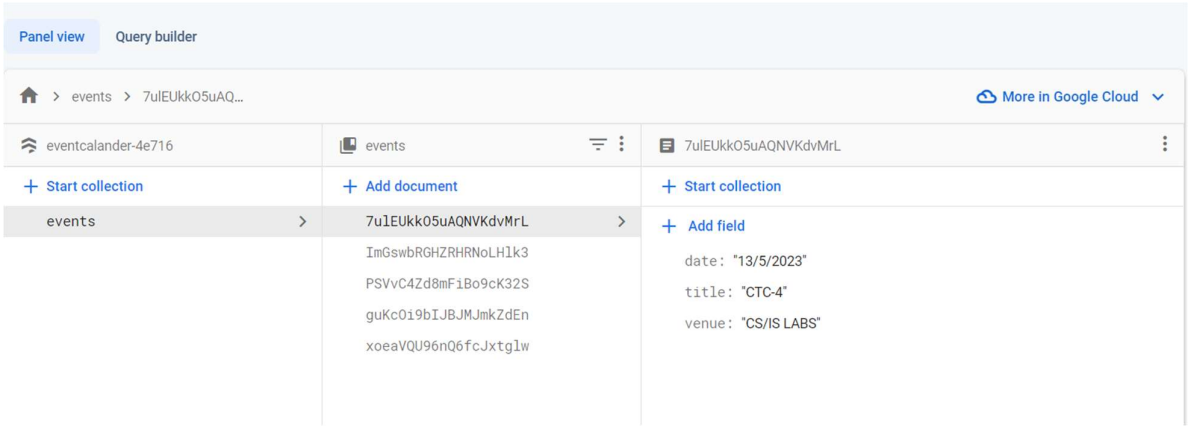
The database is designed using Google Firebase Console in which data is stored in a popular data structure known as JSON tree (JavaScript Object Notation). Every time when the data transfer happens from the client end, the information given to the UI is converted into a JSON tree structure which is an efficient and faster way to retrieve and store data.

A snapshot of user Authentication:



A snapshot of data in the Firebase database:





CHAPTER 5

IMPLEMENTATION

We are here designing an app to provide students with reminders, incentives, and rewards for achieving academic goals.

5.1 SOLUTION APPROACH/METHODOLOGY

We are here using xml and java for the front end and firebase for the backend as a server.

5.1.1 FIREBASE

Firebase is considered as web application platform. It helps query for inserting, updating, deleting or adding data to it. It is the backend of a system that is used as a database for storing data.

Firebase real-time database feature is very easy to use. Once the Firebase and database dependency is added to the app, unstructured data can be added to database.

5.1.2 STORAGE

The files like images, audio, video etc can be stored in the app. The data stored is highly secured and is robust in nature means it resumes from the last point if any network error occurs.

5.1.3 FIREBASE AND ANDROID APP

An Android application has been developed for the demonstration of Firebase. In this app images along with strings are loaded to Firebase and retrieved from Firebase similar to Instagram. For the development of an Android app to demonstrate the use of Firebase, prototyping model has been followed.

Steps for connecting App to Firebase:

Step1: An account in the Firebase Login has to be created at https://www.firebase.com/login/ using the Google account.

Step2: Creating a new application on Firebase. Firebase creates a new application when one logs in for the first time. Also, at the bottom left corner, one can find an option to create a new application on the Firebase server. The app URL has to be unique among all applications deployed on Firebase.

Step3: Next step is to add Firebase as a project dependency. Make changes to the following lines to the build.gradle file, which is located in the app's project folder, and not the root folder. After adding any dependency, one has to make sure to sync the application. If there is any build error complaint about duplicate files then one can choose to exclude those files by adding the packaging Options directive to the build.gradle file: android

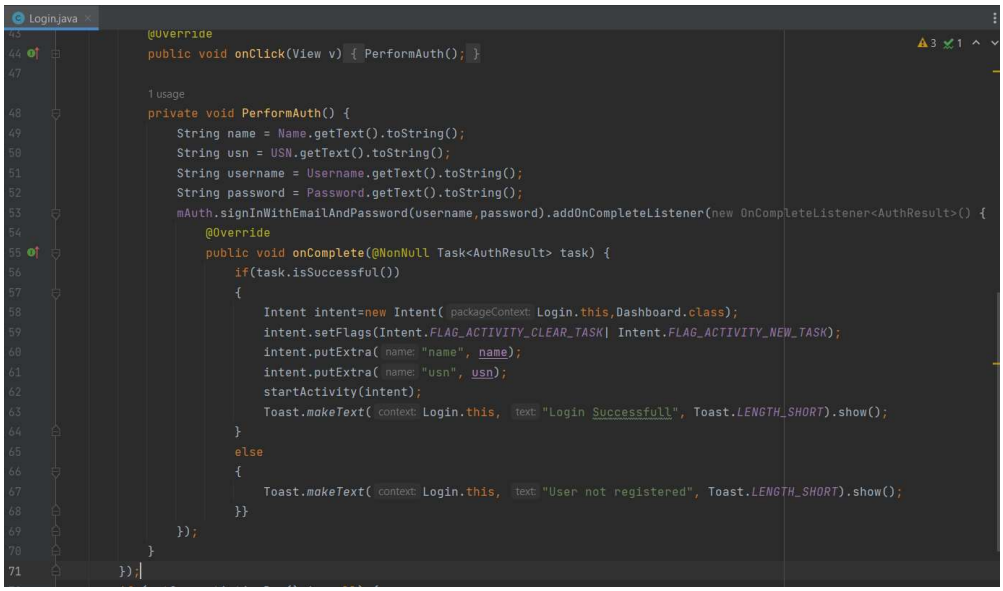
Step4: Next, add permissions to Android application, add network permission to the app, the same way it has been done for parse earlier. Now add the following line to the AndroidManifest.xml file:
<uses-permission android:name="android.permission.INTERNET" / >
Firebase is a Backend-as-a-Service—BaaS—that started as an YC11 start up and grew up into a next-generation app-development platform on Google Cloud Platform.

5.1.4 Java

There are several ways to create apps for Android devices, but the recommended method for most developers is to write native apps using Java and the Android SDK. Java for Android apps is both similar and quite different from other types of Java applications.
If you have experience with Java (or a similar language) then you'll probably feel comfortable diving right into the code and learning how to use the Android SDK to make your app run. But if you're new to programming or object-oriented languages then you'll probably want to get familiar with the syntax of the Java language and how to accomplish basic programming tasks before learning how to use the Android SDK.

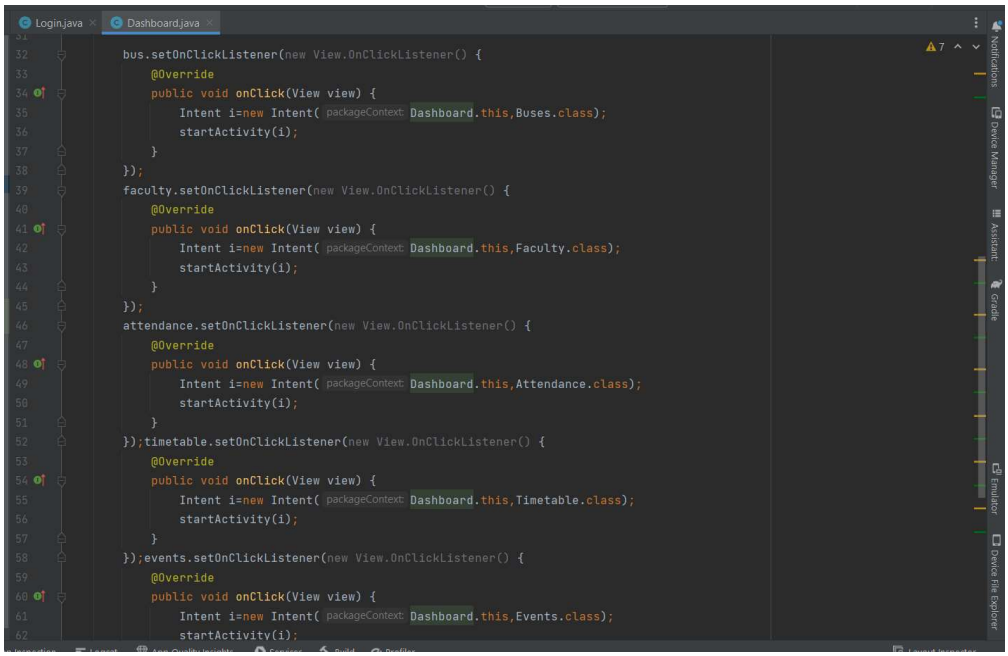
5.2 IMPLEMENTATION CODE

Login:



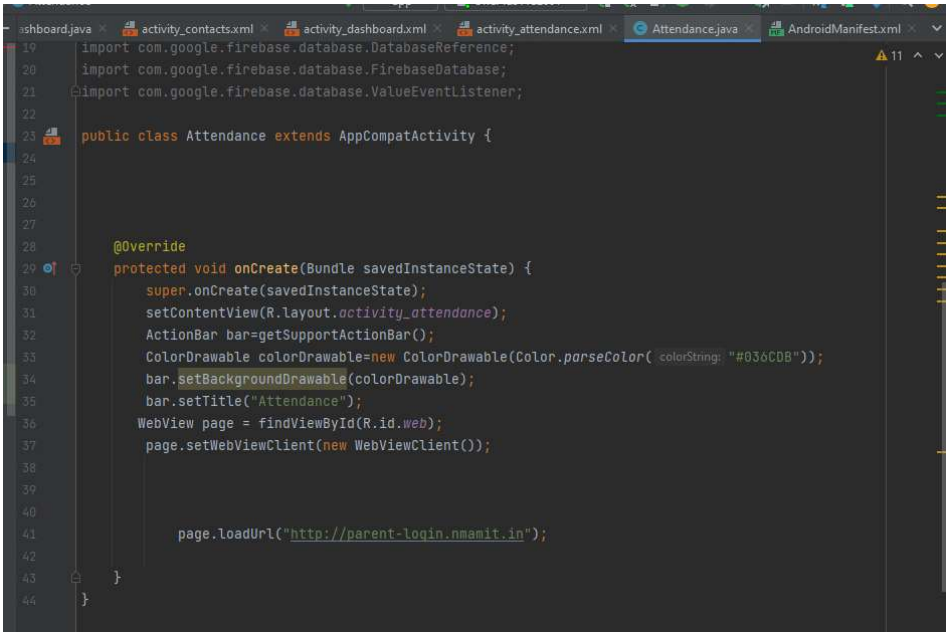
```
Login.java
    @Override
    public void onClick(View v) { PerformAuth(); }

    1 usage
    private void PerformAuth() {
        String name = Name.getText().toString();
        String usn = USN.getText().toString();
        String username = Username.getText().toString();
        String password = Password.getText().toString();
        mAuth.signInWithEmailAndPassword(username,password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                if(task.isSuccessful())
                {
                    Intent intent=new Intent( packageContext: Login.this,Dashboard.class);
                    intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK| Intent.FLAG_ACTIVITY_NEW_TASK);
                    intent.putExtra( name: "name", name);
                    intent.putExtra( name: "usn", usn);
                    startActivity(intent);
                    Toast.makeText( context: Login.this,  text: "Login Successfull", Toast.LENGTH_SHORT).show();
                }
                else
                {
                    Toast.makeText( context: Login.this,  text: "User not registered", Toast.LENGTH_SHORT).show();
                }}
            });
        }
    });
```

Dashboard:

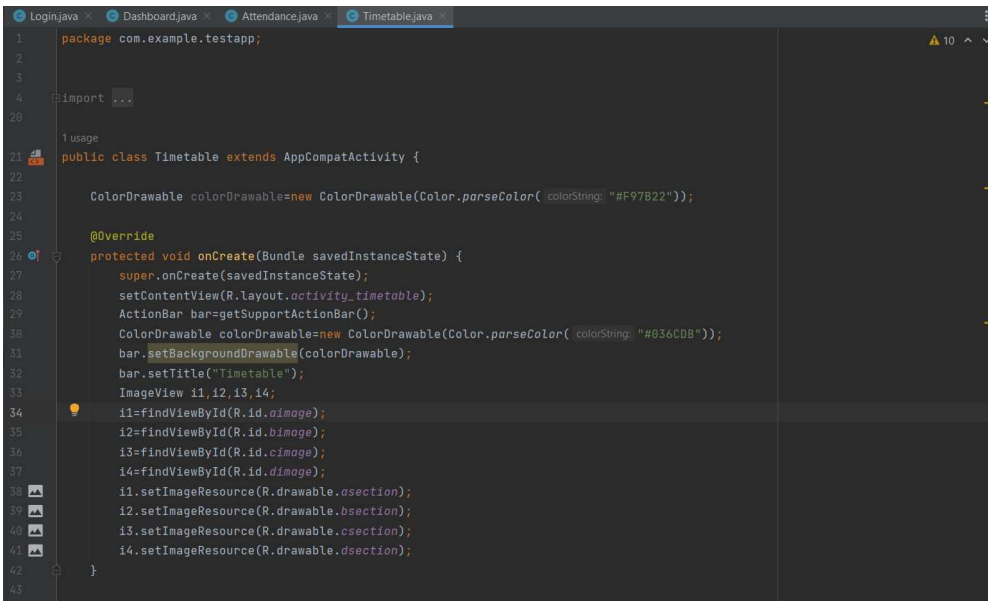


```
Login.java  Dashboard.java
    bus.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i=new Intent( packageContext: Dashboard.this,Buses.class);
            startActivity(i);
        }
    });
    faculty.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i=new Intent( packageContext: Dashboard.this,Faculty.class);
            startActivity(i);
        }
    });
    attendance.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i=new Intent( packageContext: Dashboard.this,Attendance.class);
            startActivity(i);
        }
    });timetable.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i=new Intent( packageContext: Dashboard.this,Timetable.class);
            startActivity(i);
        }
    });events.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent i=new Intent( packageContext: Dashboard.this,Events.class);
            startActivity(i);
```

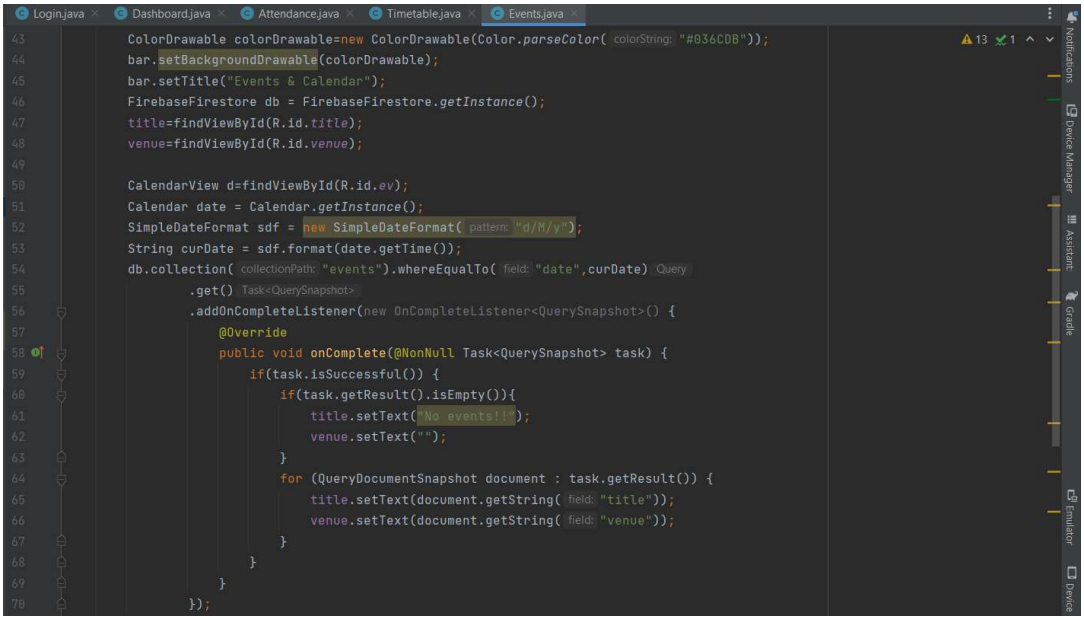
Attendance:



Timetable:

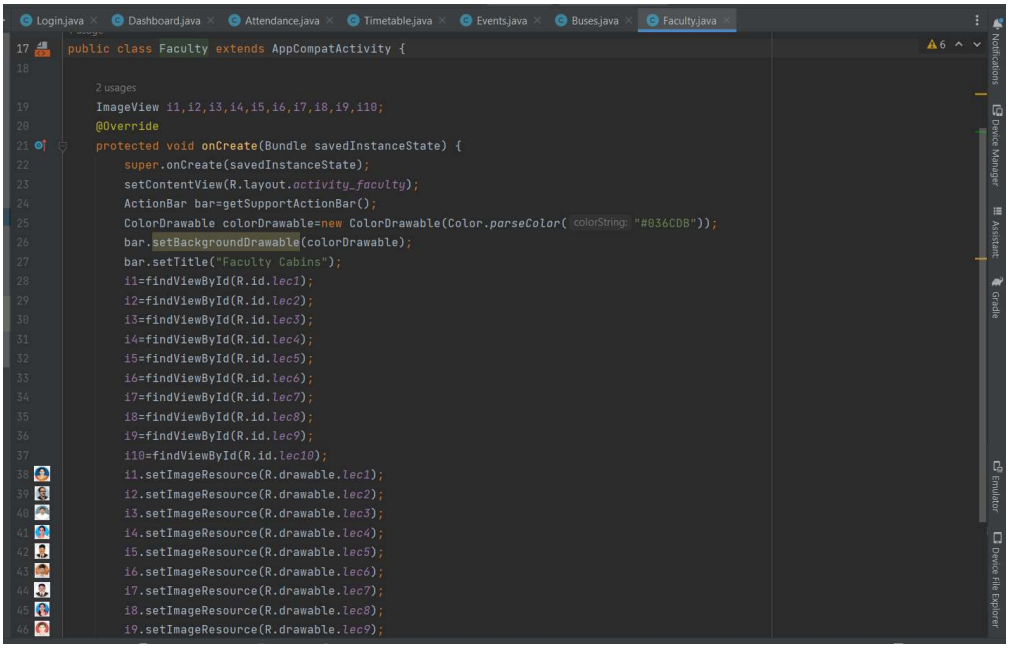


Events:



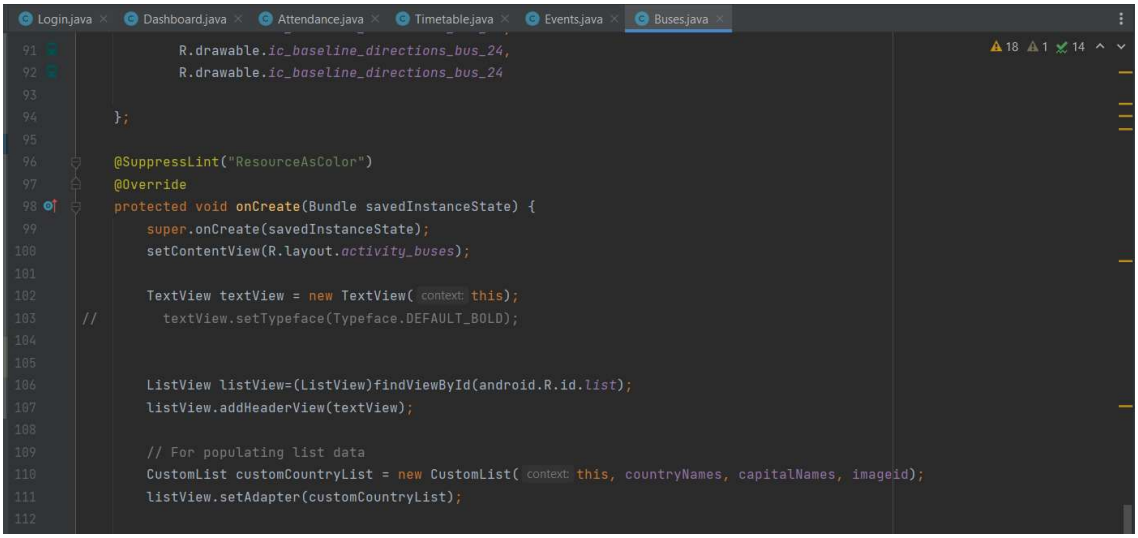
```
ColorDrawable colorDrawable=new ColorDrawable(Color.parseColor( colorString: "#036CDB"));
bar.setBackgroundDrawable(colorDrawable);
bar.setTitle("Events & Calendar");
FirebaseFirestore db = FirebaseFirestore.getInstance();
title=findViewById(R.id.title);
venue=findViewById(R.id.venue);

CalendarView d=findViewById(R.id.ev);
Calendar date = Calendar.getInstance();
SimpleDateFormat sdf = new SimpleDateFormat( pattern: "d/M/y");
String curDate = sdf.format(date.getTime());
db.collection( collectionPath: "events").whereEqualTo( field: "date",curDate) Query
        .get() Task<QuerySnapshot>
        .addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
            @Override
            public void onComplete(@NonNull Task<QuerySnapshot> task) {
                if(task.isSuccessful()) {
                    if(task.getResult().isEmpty()){
                        title.setText("No events!!");
                        venue.setText("");
                    }
                    for (QueryDocumentSnapshot document : task.getResult()) {
                        title.setText(document.getString( field: "title"));
                        venue.setText(document.getString( field: "venue"));
                    }
                }
            }
        });
```

Faculty Cabin:

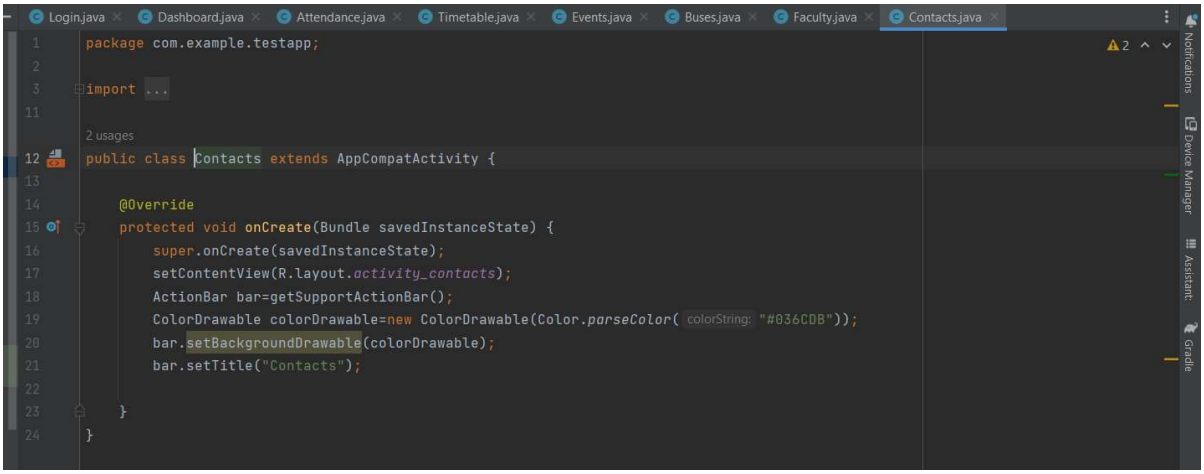


```
public class Faculty extends AppCompatActivity {

    2 usages
    ImageView i1,i2,i3,i4,i5,i6,i7,i8,i9,i10;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_faculty);
        ActionBar bar=getSupportActionBar();
        ColorDrawable colorDrawable=new ColorDrawable(Color.parseColor( colorString: "#036CDB"));
        bar.setBackgroundDrawable(colorDrawable);
        bar.setTitle("Faculty Cabins");
        i1=findViewById(R.id.lec1);
        i2=findViewById(R.id.lec2);
        i3=findViewById(R.id.lec3);
        i4=findViewById(R.id.lec4);
        i5=findViewById(R.id.lec5);
        i6=findViewById(R.id.lec6);
        i7=findViewById(R.id.lec7);
        i8=findViewById(R.id.lec8);
        i9=findViewById(R.id.lec9);
        i10=findViewById(R.id.lec10);
        i1.setImageResource(R.drawable.lec1);
        i2.setImageResource(R.drawable.lec2);
        i3.setImageResource(R.drawable.lec3);
        i4.setImageResource(R.drawable.lec4);
        i5.setImageResource(R.drawable.lec5);
        i6.setImageResource(R.drawable.lec6);
        i7.setImageResource(R.drawable.lec7);
        i8.setImageResource(R.drawable.lec8);
        i9.setImageResource(R.drawable.lec9);
```

Bus Schedule:

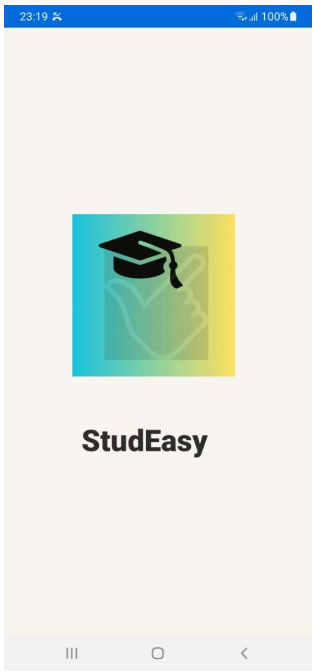


Contacts:

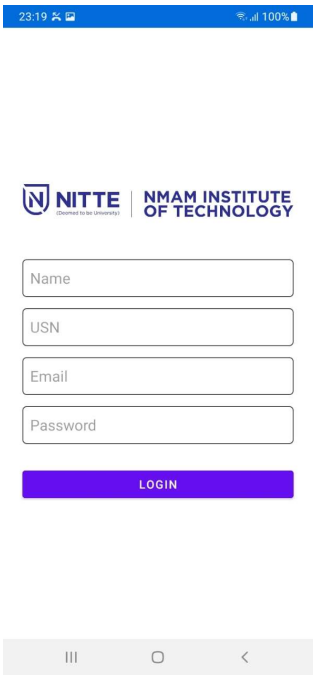


CHAPTER 6

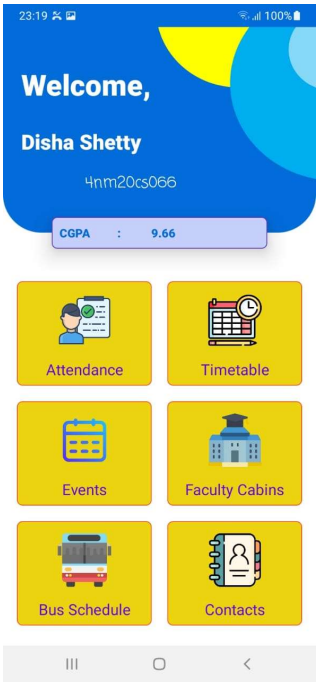
SCREENSHOTS



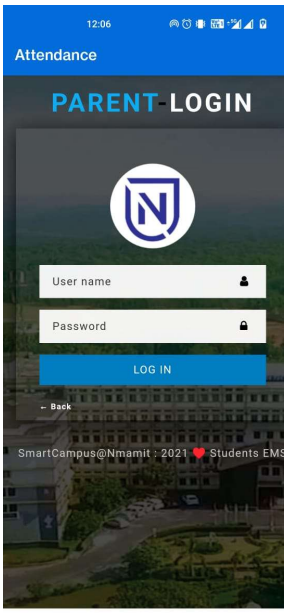
Log In:



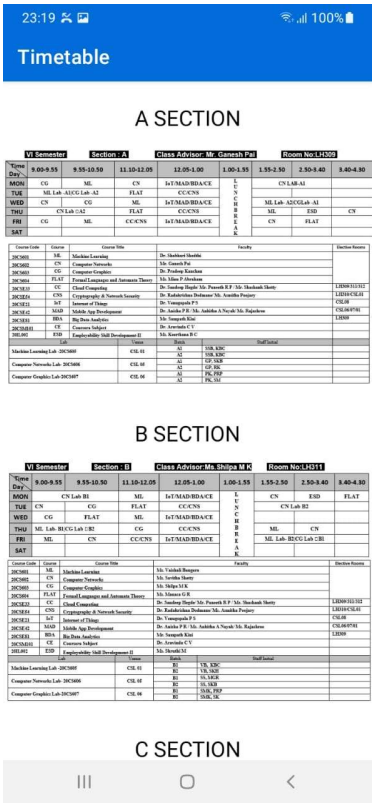
Dashboard:



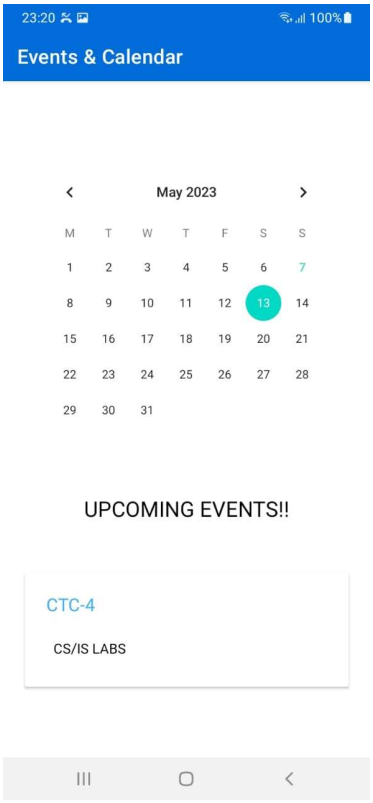
Attendance:



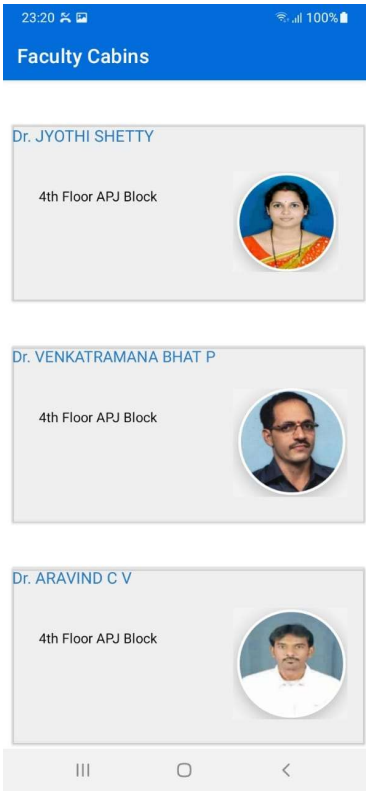
Timetable:



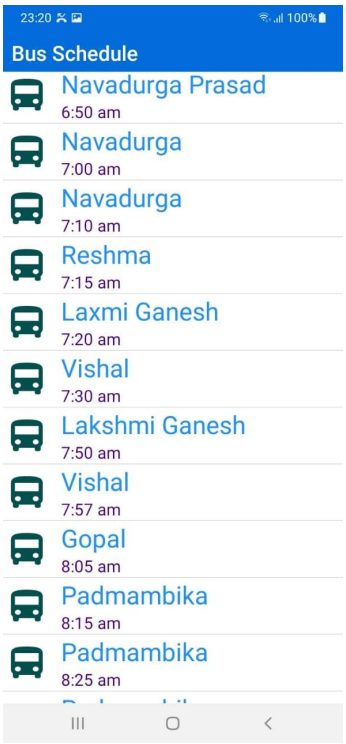
Events:



Faculty Cabins:



Bus Schedule:



Contacts:

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Results/Conclusion:

- StudEasy is an android application for students to view and access their academic related information.
- The student can login to the app and view his/her dashboard which has links to attendance, timetable, faculty cabins, contacts, upcoming events calendar etc.
- The dashboard shows details about the student such as his/her name, usn and cgpa.
- Since students usually use buses as mode of transport, the app provides the functionality of viewing the available buses and their schedules.

7.2 Future Works

In the future, we may extend this project by adding extra features to our android app like,

- Separate modules for admin(faculty) and user (student).
- Provide resources like notes, previous question papers, textbooks etc.
- Display marks in every subject.
- Extend it to provide services to students of all branches.

8 References

- "Overview Guides Reference Samples Libraries Support Go to console" Documentation Firebase, https://firebase.google.com/docs/
- Stack Overflow, https://stackoverflow.com
- YouTube, https://www.youtube.com