

B.M.S. COLLEGE OF ENGINEERING

(Autonomous College under VTU)

Bull Temple Road, Basavanagudi, Bangalore – 560019



A project report on

“BMSCE Learn, A Learning Management System”

Submitted in partial fulfillment of the requirements for the award of the degree

**BACHELOR OF ENGINEERING
IN
INFORMATION SCIENCE AND ENGINEERING**

By

Sughosh S Durg, 1BM19IS159

Roshan Warrier, 1BM18IS085

Under the guidance of
Dr. Shubha Rao V, Associate Professor

Department of Information Science and Engineering

2021-22

B.M.S. COLLEGE OF ENGINEERING
(Autonomous College under VTU)
Bull Temple Road, Basavanagudi, Bangalore – 560019



Department of Information Science and Engineering

CERTIFICATE

This is to certify that the project entitled “BMSCE Learn, A Learning Management System” is a bonafide work carried out by Sughosh S Durg (**1BM19IS159**) and Roshan Warrier (**1BM18IS085**) in partial fulfillment for the award of the degree of Bachelor of Engineering in **Information Science and Engineering** from **Visvesvaraya Technological University, Belgaum** during the year **2021-2022**. It is certified that all corrections/suggestions indicated for Internal Assessments have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering Degree.

Signature of the Faculty
Dr. Shubha Rao V, Associate Professor

Signature of the HOD
Dr. Dakshayini, Professor

TABLE OF CONTENTS

Topic	Page No.
Abstract	4
1. Introduction	5
2. Tools used and System Requirements	6
3. System Architecture	8
i. Use-Case Diagram	9
ii. Activity Diagram	10
4. User Interface (UI) and Navigation	11
5. Implementation	16
6. Results	24
7. Test Suite	34
8. References	38

ABSTRACT

Recently the popularity of smartphones has promoted communication for daily life, but it also brings new challenges for current classroom teaching. For example, some students spend too much time playing with phones instead of learning in the classroom. How to effectively instruct students to use phones and promote teaching becomes a very tricky problem. In order to deal with this problem, we propose a new real-time interaction platform for classroom teaching based on a smartphone APP, which adopts the extensive multilayer framework and the quantifiable real-time interaction. In this platform, it's easy for teachers to post assignments, announcements, and manage courses and for the student to keep track of important announcements, assignments, and manage courses. What's more, this platform can strengthen the interaction between teachers and students, and stimulate the interest of all students to participate in classroom learning.

INTRODUCTION

In recent years, with the rapid development of network information technology, many new teaching modes and methods have emerged in education such as network courses, MOOCs, and flipped classrooms. Though these new modes can make teaching convenient for cross-regions, classroom teaching is still a very important teaching model in the current education environment. To deal with the shortcomings of the existing classroom teaching, we propose a new real-time interactive teaching method, which constructs a real-time interaction platform for classroom teaching based on a smartphone application. The application was developed using Java, XML, SQLite (database), and Android Studio as the IDE. The application runs on any Android Device that has Lollipop 5.0 and above.

SYSTEM REQUIREMENTS AND SPECIFICATION

Tools used for application development:

1. **Android Studio:** Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.
2. **Android SDK and SDK Platform tools:** Android SDK Platform-Tools is a component for the Android SDK. It includes tools that interface with the Android platform, such as adb, fastboot, and systrace. These tools are required for Android app development.
3. **Java Development Kit (JDK):** It implements the Java Language Specification and the Java Virtual Machine Specification and provides the Standard Edition of the Java Application Programming Interface.
4. **Java Programming Language:** Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible.
5. **SQLite (Database):** SQLite is an opensource SQL database that stores data to a text file on a device. Android comes in with built in SQLite database implementation.

SQLite supports all the relational database features. In order to access this database, you don't need to establish any kind of connections for it like JDBC, ODBC e.t.c.

6. **XML:** XML stands for Extensible Markup Language. XML is a markup language much like HTML used to describe data. In Android Development, the XML is used to implement UI-related data, and it's a lightweight markup language that doesn't make layout heavy.

System Requirements to run the application:

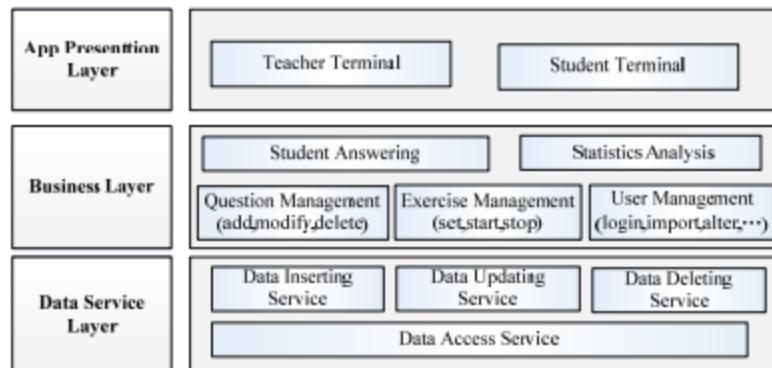
1. Android Studio (3.0 or higher)
2. Java Development Kit (JDK) 9 and above
3. Android SDK and SDK tools (API Level 20 and above)
4. Android Build Tools
5. Android Device or an Emulator with Android Lollipop or greater
6. Storage Read/Write Permissions

SYSTEM ARCHITECTURE

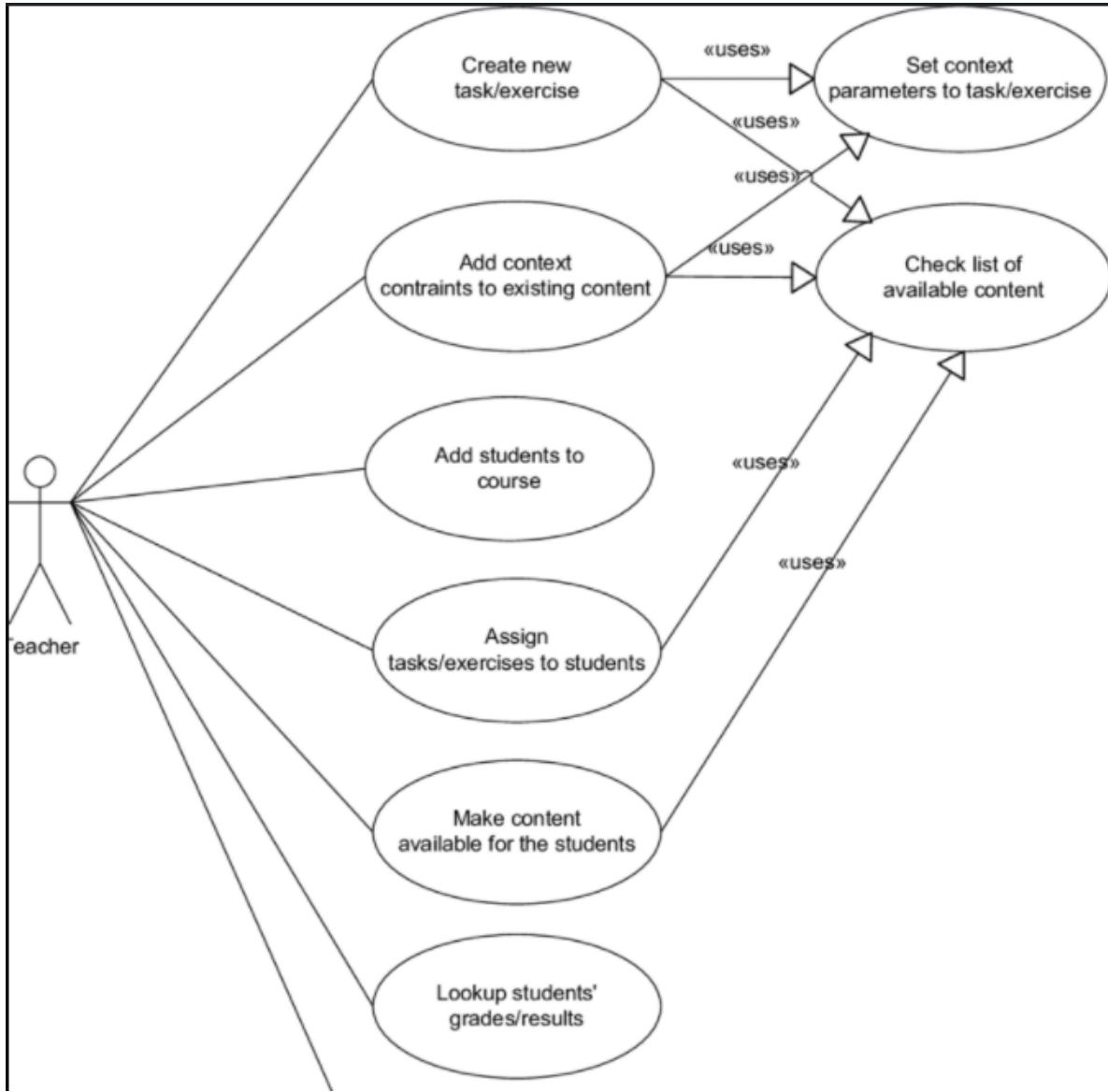
In order to ensure the robustness and extendibility of the platform, we adopted a multilayer framework in the real-time interactive classroom teaching platform.

Based on the smartphone APP, this platform combines real-time interaction in class with the real-time evaluation of teaching quality. The core businesses of the platform include teacher-side data management, student-side answer management. They are explained as below:

1. Teacher-side data management includes the importing of existing data, student management, question management, exercise set, exercise start, answer information view, and so on.
2. Student-side answer management includes answering questions according to the teacher's exercise tasks, submitting the assignments, viewing historical answers, and so on.

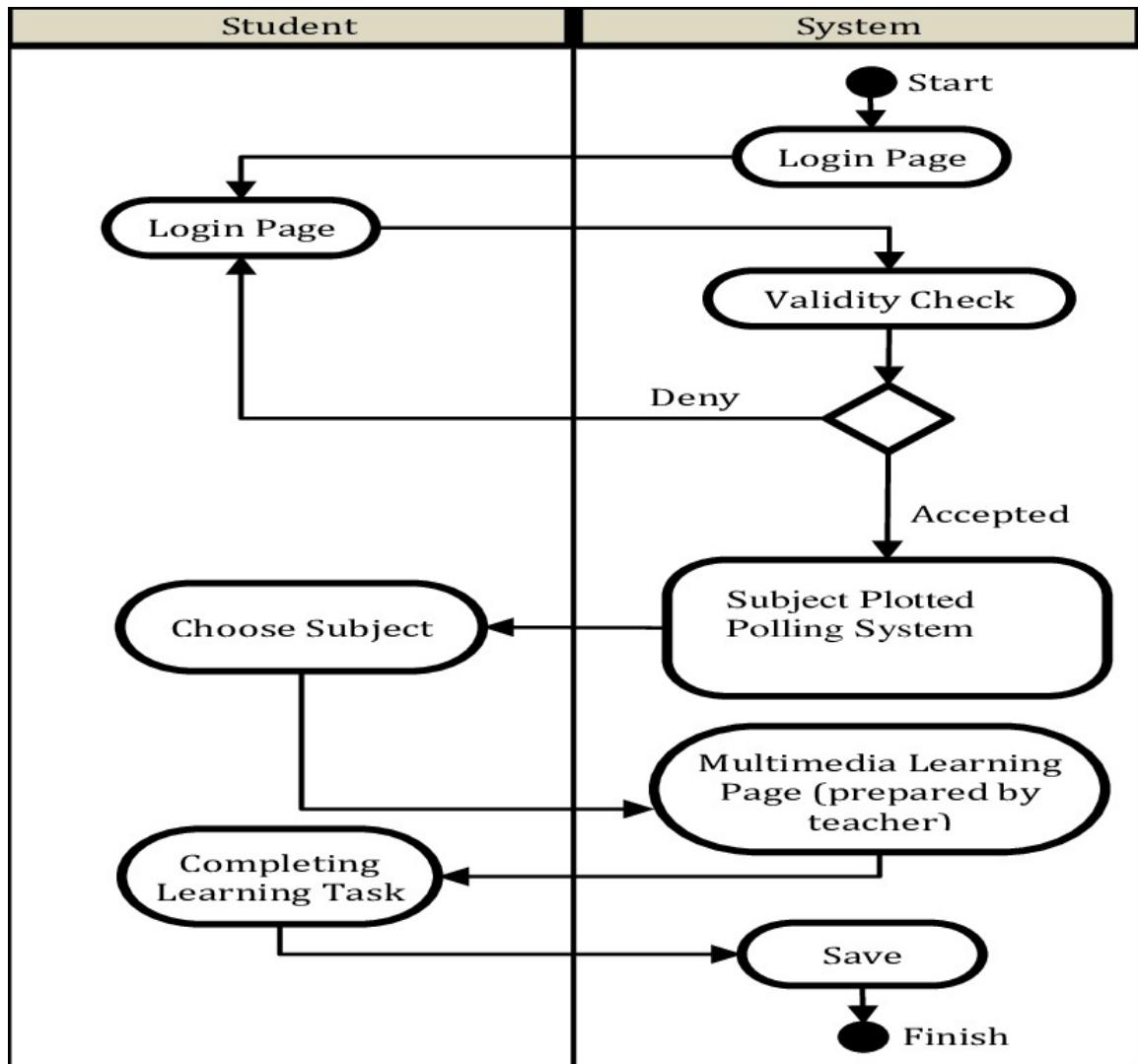


USE-CASE DIAGRAM



This interactive function can be used to assess students' mastery of knowledge. Meanwhile, it can also be used to assess the teaching methods of teachers. The platform can send questions to students on the spot through the APP-based platform immediately after the teacher has finished teaching a certain knowledge point. Students can submit their answers within a specified time. Teachers can accurately grasp the students' acceptance of the knowledge point and timely evaluate the teaching effect of the knowledge point.

ACTIVITY DIAGRAM



There are two types of login i.e two types of users can log into the BMSCE Learn course management app: Teachers and Students. Courses Teachers can add/edit a course. Students can only view the courses they have enrolled in. Teachers can add/edit/delete an assignment and can view submitted works by students. Students can only view the assignments and upload their works. Announcements Teachers can add/edit/delete announcements. Students can only view the announcements.

USER INTERFACE (UI) DESIGN AND NAVIGATION

Providing a user-friendly interface is of the functional requirements of the application and hence care has been taken to use best practices while designing the overall UI.

The Various UI components used are the following :

1. **TextView:** It is used to display texts which are meant to be not editable by the user by default. This has been used for headings in the app.
2. **EditText:** It is used to collect text from the user. It's an editable text box where users can enter values.
3. **ImageView:** ImageView is used to display images in Android applications. An image can be displayed by assigning it to the ImageView control and including the android: src attribute in the XML definition. Images can be assigned dynamically.
4. **Buttons:** A UI component that when clicked performs a certain task.
5. **Intent:** A navigation component used for communication and navigating from one activity to another.
6. **Toast:** A toast provides simple feedback about an operation in a small popup. It automatically disappears after a timeout.
7. **List View:** A list view is an adapter view that does not know the details, such as type and contents, of the views it contains. Instead, list view requests.
8. **Menus:** Used to present user actions and other options in the activities.
9. **Layouts:** the types of layouts used are:- Linear layout and Constraint layout.
 - i) **Linear Layout** - LinearLayout is a view group that aligns all children in a single direction, vertically or horizontally.
 - ii) **Constraint Layout** - it is a ViewGroup (i.e. a view that holds other views) that allows us to create large and complex layouts with a flat view hierarchy, and also allows us to position and size widgets in a very flexible way.

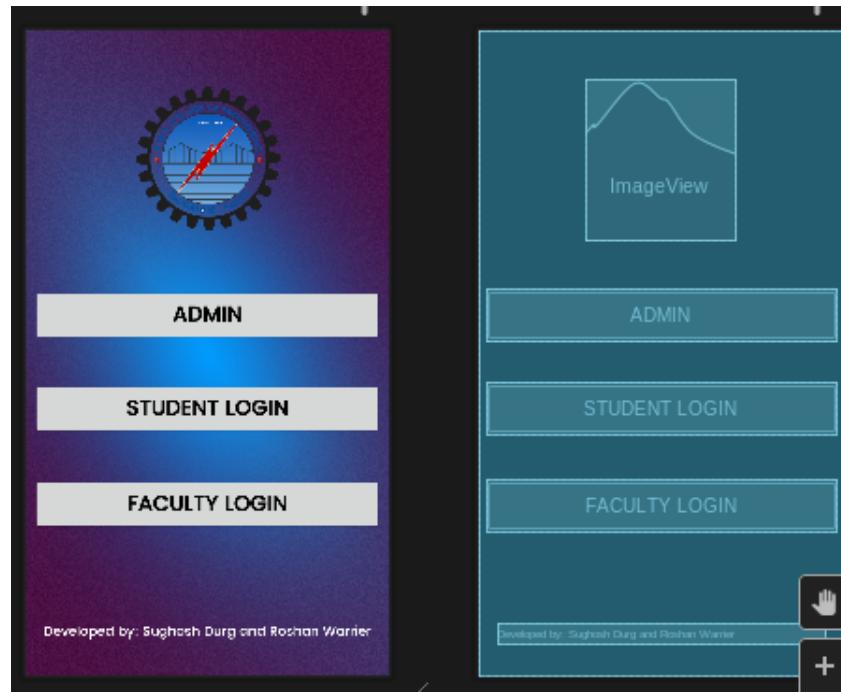


Fig 3.1 Main Activity

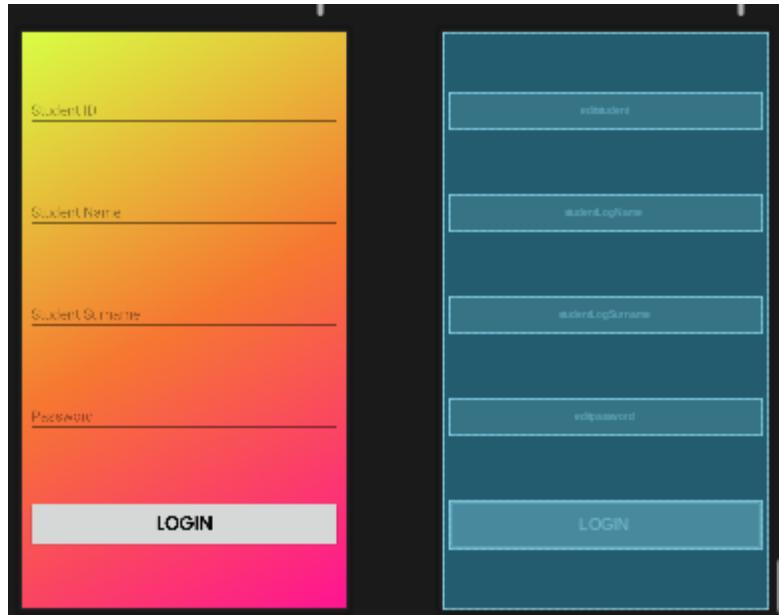


Fig 3.2 Login Activity

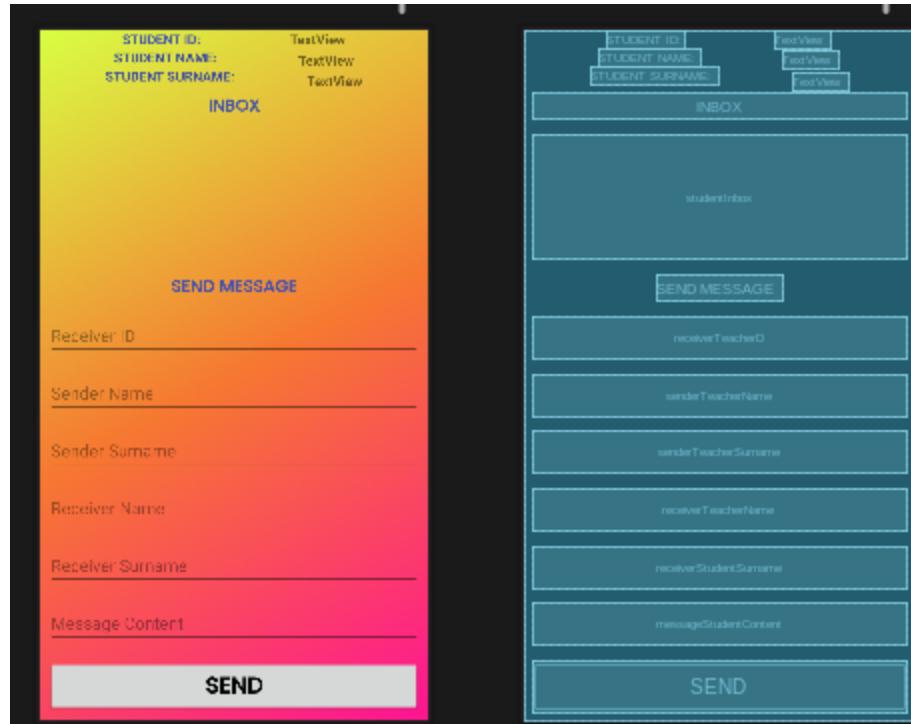


Fig 3.3 Send Message Activity

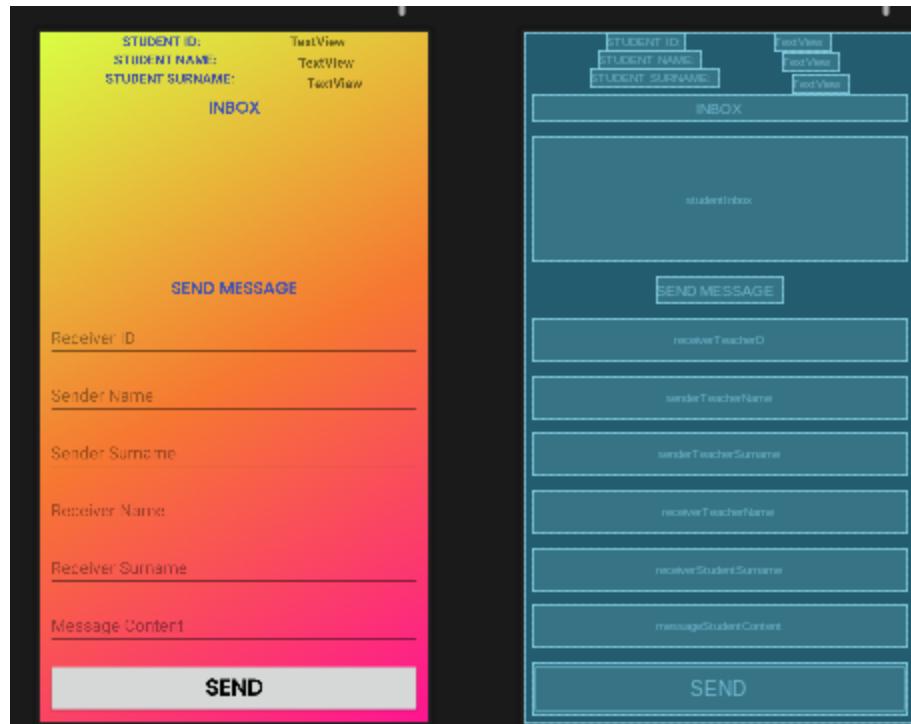


Fig 3.4 Post Announcements Activity

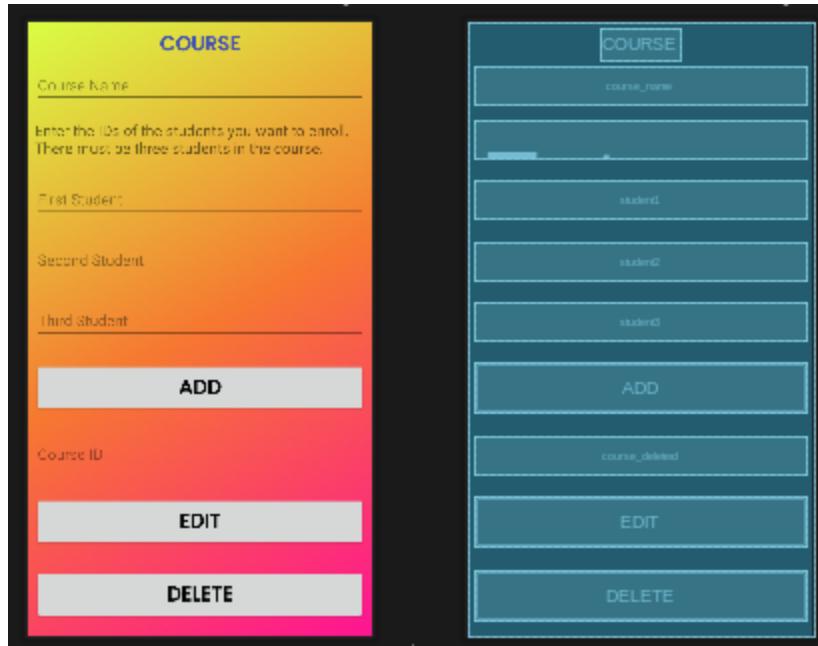


Fig 3.5 Course Activity

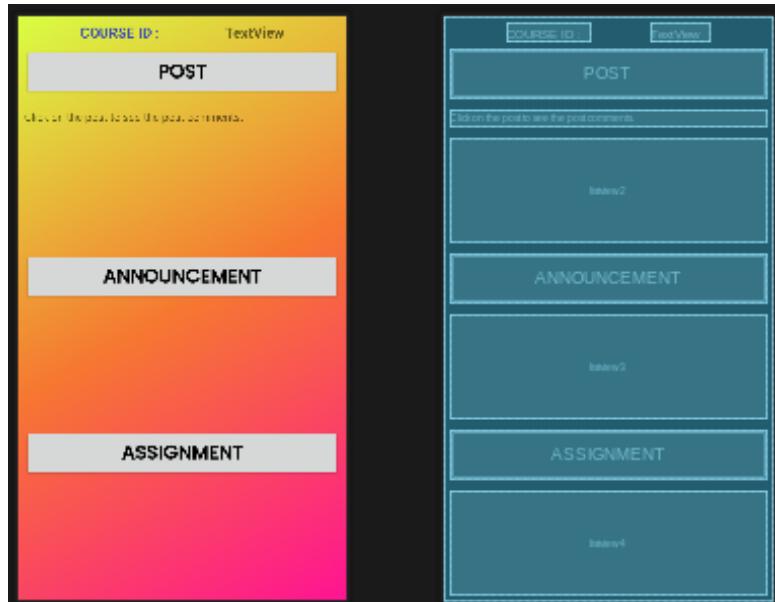


Fig 3.6 Course Menu Activity

Post

CourseID
Post Name
Post Content

ADD

PostID

EDIT

DELETE

POST

courseid
postname
postcontent

ADD

postid

EDIT

DELETE

Fig 3.7 Post Activity

Course ID

Post ID

Comment Content

Name

Surname

ADD COMMENT

editedID

edited1

edited2

edited3

edited4

ADD COMMENT

Fig 3.8 Add Comment Activity

IMPLEMENTATION

GitHub Repository Link - <https://github.com/sughosh-durg/MAD-Project-Final/>

1. MainActivity.java

```
package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;

import android.content.Intent;
import android.graphics.drawable.AnimationDrawable;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {

    Button admin,login_teacher,login_student;

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

        admin=(Button)findViewById(R.id.btn_admin);
        login_teacher=(Button)findViewById(R.id.btn_teacher);
        login_student=(Button)findViewById(R.id.btn_student);

        admin.setOnClickListener(new Clik());
        login_teacher.setOnClickListener(new Clik());
        login_student.setOnClickListener(new Clik());

    }
}
```

```

public class Clik implements View.OnClickListener{

    @Override
    public void onClick(View v) {

        switch (v.getId()){
            case R.id.btn_admin:
                openAdmin();
                break;
            case R.id.btn_teacher:
                openLoginTeacher();
                break;
            case R.id.btn_student:
                openLoginStudent();
                break;
            default:
        }
    }

    public void openAdmin(){
        Intent intent = new Intent(this,Admin.class);
        startActivity(intent);
    }

    public void openLoginTeacher(){
        Intent intent = new Intent(this,Teacher_Login.class);
        startActivity(intent);
    }

    public void openLoginStudent(){
        Intent intent = new Intent(this,Log.class);
        startActivity(intent);
    }
}

```

2. Course.java

```

package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;

```

```

import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Course extends AppCompatActivity {
    EditText coursename,student1,student2,student3,courseid;
    Button add,delete,edit;

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

        coursename=findViewById(R.id.course_name);
        student1=findViewById(R.id.student1);
        student2=findViewById(R.id.student2);
        student3=findViewById(R.id.student3);
        courseid=findViewById(R.id.course_deleteid);

        add=findViewById(R.id.course_add);
        delete=findViewById(R.id.course_delet);
        edit=findViewById(R.id.course_edit);
    }
}

```

3. Assignment.java

```

package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

```

```

public class Assigment extends AppCompatActivity {

    EditText asssing_courseid,assing_name,assing_content,assing_id,deudate;
    Button add,delete,edit;

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

        asssing_courseid=findViewById(R.id.assign_courseid);
        assing_name=findViewById(R.id.assign_name);
        assing_content=findViewById(R.id.assing_content);
        deudate=findViewById(R.id.duedate);
        assing_id=findViewById(R.id.assing_id);

        add=findViewById(R.id.assing_add);
        delete=findViewById(R.id.assing_delete);
        edit=findViewById(R.id.assing_edit);

        add.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String x = asssing_courseid.getText().toString();
                String y = assing_name.getText().toString();
                String z = assing_content.getText().toString();
                String w = deudate.getText().toString();

                if (TextUtils.isEmpty(x)){
                    Toast.makeText(Assigment.this,"Error. enter the course code.",Toast.LENGTH_SHORT).show();
                }
                else if (TextUtils.isEmpty(y)){
                    Toast.makeText(Assigment.this,"Error. Enter the assignment name.",Toast.LENGTH_SHORT).show();
                }
                else if (TextUtils.isEmpty(z)){
                    Toast.makeText(Assigment.this,"Error. Enter the assignment

```

```

content.",Toast.LENGTH_SHORT).show();
    }

else{
    Databasehelper databaseHelper = new Databasehelper(Assigment.this);
    boolean success =databaseHelper.addAssignment(x,y,z,w);
    Toast.makeText(Assigment.this,"Adding assignment
successful.",Toast.LENGTH_SHORT).show();
    assing_courseid.setText("");
    assing_name.setText("");
    assing_content.setText("");
    deudate.setText(" ");
}

});

edit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String x = assing_courseid.getText().toString();
        String y = assing_name.getText().toString();
        String z = assing_content.getText().toString();
        String w = deudate.getText().toString();
        String a = assing_id.getText().toString();

        if (TextUtils.isEmpty(x)){
            Toast.makeText(Assigment.this,"Error. Enter the course
code.",Toast.LENGTH_SHORT).show();
        }
        else if (TextUtils.isEmpty(y)){
            Toast.makeText(Assigment.this,"Error. Enter the assignment
name.",Toast.LENGTH_SHORT).show();
        }
        else if (TextUtils.isEmpty(z)){
            Toast.makeText(Assigment.this,"Error. Enter the assignment
content.",Toast.LENGTH_SHORT).show();
        }

    }

}

```

```

Databasehelper databaseHelper = new Databasehelper(Assigment.this);
boolean success =databaseHelper.updateAssingment(a,x,y,z,w);
Toast.makeText(Assigment.this,"Editing assignment
successful.",Toast.LENGTH_SHORT).show();
asssing_courseid.setText("");
assing_name.setText("");
assing_content.setText("");
deudate.setText(" ");
}
}
});

```

4. Send_Message.java

```

package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;

import java.util.List;

public class MessageStudent extends AppCompatActivity {
    TextView textView, textView2, textView3;
    EditText x2, y2, z2, w2, a2, b2;
    Button addmessage;
    ListView studentInbox;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

```

```

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_message_student);

textView=findViewById(R.id.textView35);
textView2=findViewById(R.id.textView42);
textView3=findViewById(R.id.textView44);
x2=findViewById(R.id.receiverTeacherD);
y2=findViewById(R.id.senderTeacherName);
z2=findViewById(R.id.senderTeacherSurname);
w2=findViewById(R.id.receiverTeacherName);
a2=findViewById(R.id.receiverStudentSurname);
b2=findViewById(R.id.messageStudentContent);
studentInbox=findViewById(R.id.studentInbox);

addmessage=findViewById(R.id.sendTeacherMessageButton);

final String Temp=getIntent().getStringExtra("ClickedCourse");
textView.setText(Temp);
final String Temp2=getIntent().getStringExtra("ClickedCourse2");
textView2.setText(Temp2);
final String Temp3=getIntent().getStringExtra("ClickedCourse3");
textView3.setText(Temp3);

Databasehelper databaseHelper=new Databasehelper(MessageStudent.this);
final List<String> everyone5=databaseHelper.SurnameSearch1(Temp3);
final ArrayAdapter studentttttt = new
ArrayAdapter<String>(MessageStudent.this,android.R.layout.simple_list_item_1, everyone5);
studentInbox.setAdapter(studentttttt);

addmessage.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String x = x2.getText().toString();
        String y = y2.getText().toString();
        String z = z2.getText().toString();
        String w = w2.getText().toString();
        String a = a2.getText().toString();
    }
});

```

```
String b = b2.getText().toString();
```

5. Announcement.java

```
package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Announcement extends AppCompatActivity {
    EditText annocourseid,anno_name,anno_content,anno_id;
    Button add,delete,edit;

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

        annocourseid=findViewById(R.id.edit_courseid);
        anno_name=findViewById(R.id.announcement_name);
        anno_content=findViewById(R.id.announcement_content);
        anno_id=findViewById(R.id.announcementid);

        add=findViewById(R.id.anno_add);
        delete=findViewById(R.id.anno_delete);
        edit=findViewById(R.id.anno_edit);

        add.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String x = annocourseid.getText().toString();
                String y = anno_name.getText().toString();
                String z = anno_content.getText().toString();
                String id = anno_id.getText().toString();
                // Logic to save data to database or file
            }
        });
    }
}
```

```
String z = anno_content.getText().toString();
```

6. Post.java

```
package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Post extends AppCompatActivity {

    Button add,delete,edit;
    EditText courseid,postname,postcontent,postid;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_post);
        add=findViewById(R.id.post_add);
        delete=findViewById(R.id.post_delete);
        edit=findViewById(R.id.post_edit);
        courseid=findViewById(R.id.courseidd);
        postname=findViewById(R.id.postnameeee);
        postcontent=findViewById(R.id.postcontentttt);
        postid=findViewById(R.id.postidd);

        add.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String x = courseid.getText().toString();
                String y = postname.getText().toString();
                String z = postcontent.getText().toString();
                if (TextUtils.isEmpty(x)){

```

```

        Toast.makeText(Post.this,"Error. Enter the course
code.",Toast.LENGTH_SHORT).show();
    }
    else if (TextUtils.isEmpty(y)){
        Toast.makeText(Post.this,"Error. Enter the post
name.",Toast.LENGTH_SHORT).show();
    }
    else if (TextUtils.isEmpty(z)){
        Toast.makeText(Post.this,"Error. Enter the post
content.",Toast.LENGTH_SHORT).show();
    }
    else{
        Databasehelper databaseHelper = new Databasehelper(Post.this);
        boolean success =databaseHelper.addPost(x,y,z);
        Toast.makeText(Post.this,"Adding post
successful.",Toast.LENGTH_SHORT).show();
        courseId.setText("");
        postname.setText("");
        postcontent.setText("");
    }
}
});
```

7. List_Students.java

```

package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;

import java.util.List;

public class List_Student extends AppCompatActivity {
    ListView list_student;

    @Override
```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_list_student);
    list_student=findViewById(R.id.list_student);
    final Databasehelper databaseHelper=new Databasehelper(List_Student.this);
    final List<Student> everyone=databaseHelper.geteveryone();

                final ArrayAdapter studentArrayAdapter = new
ArrayAdapter<Student>(List_Student.this, android.R.layout.simple_list_item_1, everyone);
    list_student.setAdapter(studentArrayAdapter);

```

8. Login.java

```

package com.example.madlabproject;

import androidx.appcompat.app.AppCompatActivity;

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.Toast;

public class Log extends AppCompatActivity {
    EditText studentid,studentLogName,studentLogSurname;
    EditText password;

    Button log;

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

        studentid = findViewById(R.id.editstudent);
        password = findViewById(R.id.editpassword);
        studentLogName= findViewById(R.id.studentLogName);

```

```

studentLogSurname= findViewById(R.id.studentLogSurname);

log = findViewById(R.id.editlog);

log.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String x = studentId.getText().toString();
        String y = password.getText().toString();
        String z = studentLogName.getText().toString();
        String w = studentLogSurname.getText().toString();

        if (TextUtils.isEmpty(x)) {
            Toast.makeText(Log.this, "Error. Enter the student's ID.", Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(y)) {
            Toast.makeText(Log.this, "Error. Enter the password.", Toast.LENGTH_SHORT).show();
        } else {
            Databasehelper databasehelper = new Databasehelper(Log.this);
            boolean success = databasehelper.searchlogStudent(x);
            if (success == false) {
                Intent intent = new Intent(Log.this, Student_Menu.class);
                String id = x;
                String id2=z;
                String id3=w;
                intent.putExtra("ClickedCourse",id);
                intent.putExtra("ClickedCourse2",id2);
                intent.putExtra("ClickedCourse3",id3);
                startActivity(intent);
            } else {
                Toast.makeText(Log.this, "There is no student with this ID.", Toast.LENGTH_SHORT).show();
            }
        }
    });
}

```

RESULTS

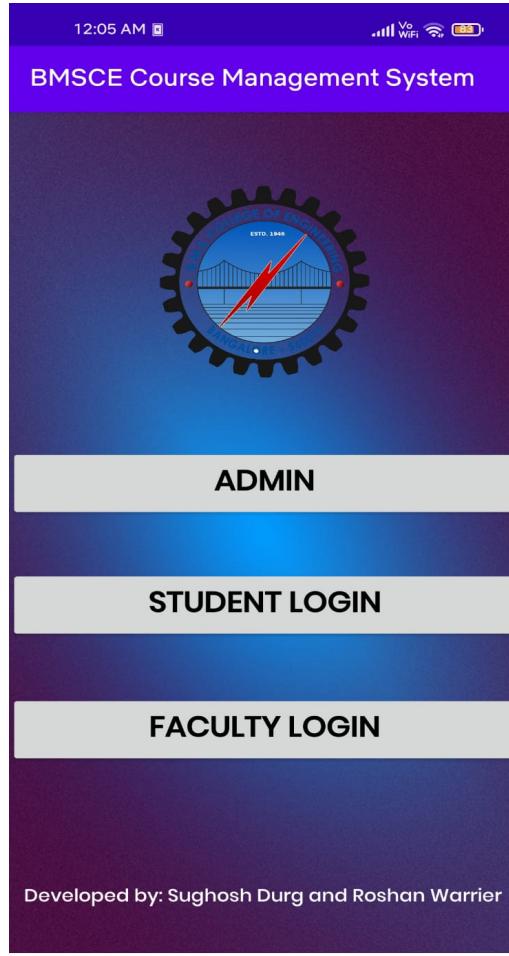


Fig 4.1 Home Screen

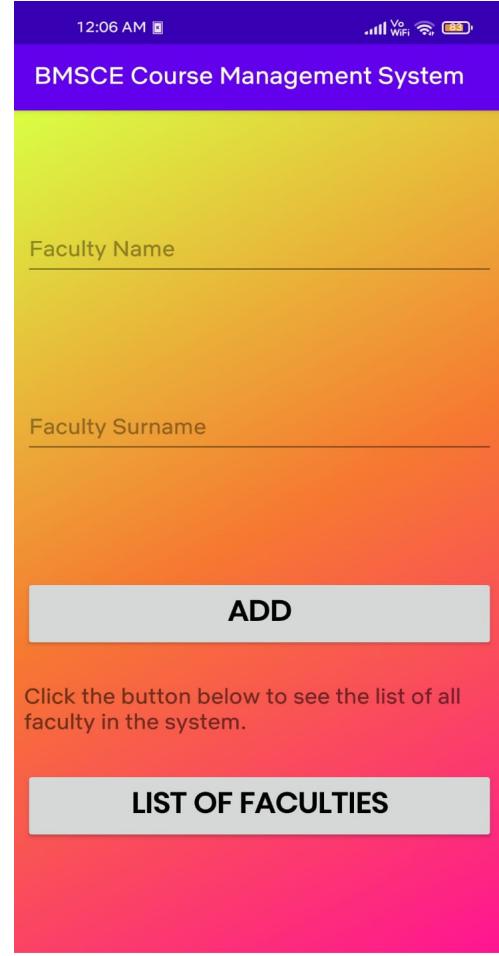


Fig 4.2 Faculty Registration

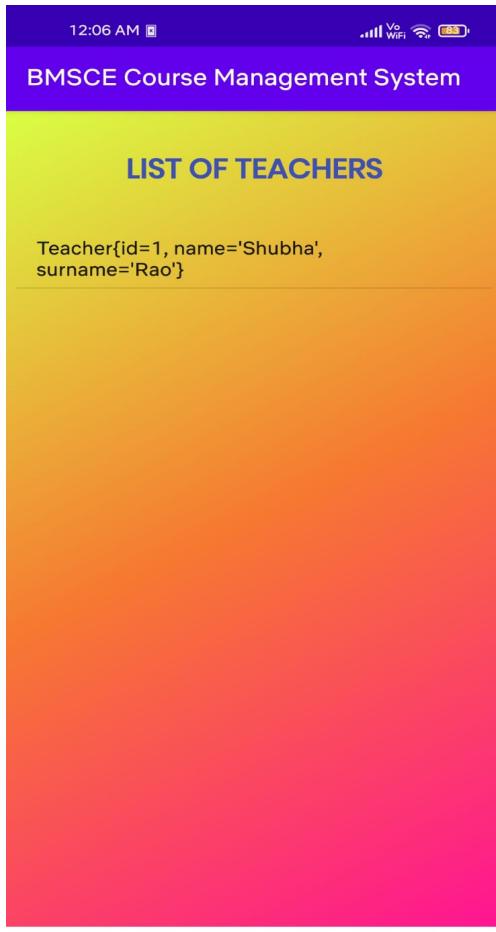


Fig 4.3 List of Faculty

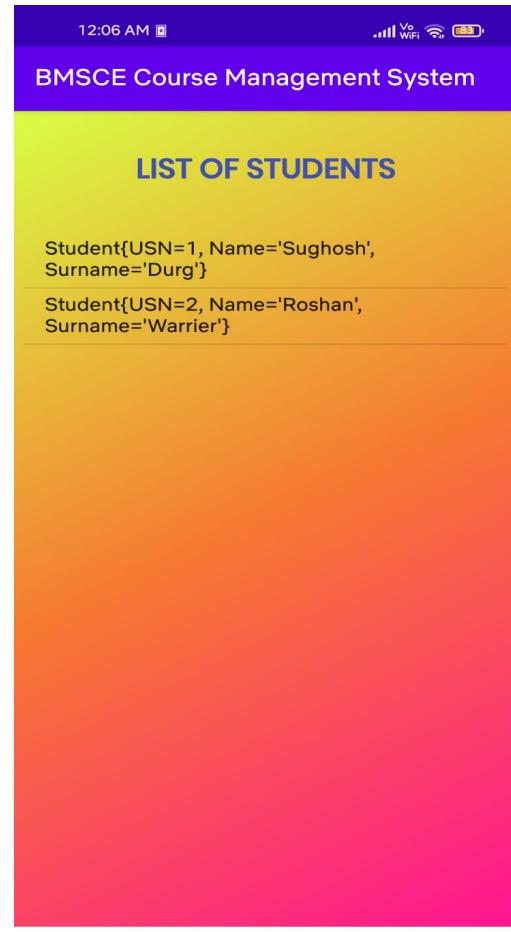


Fig 4.4 List of Students

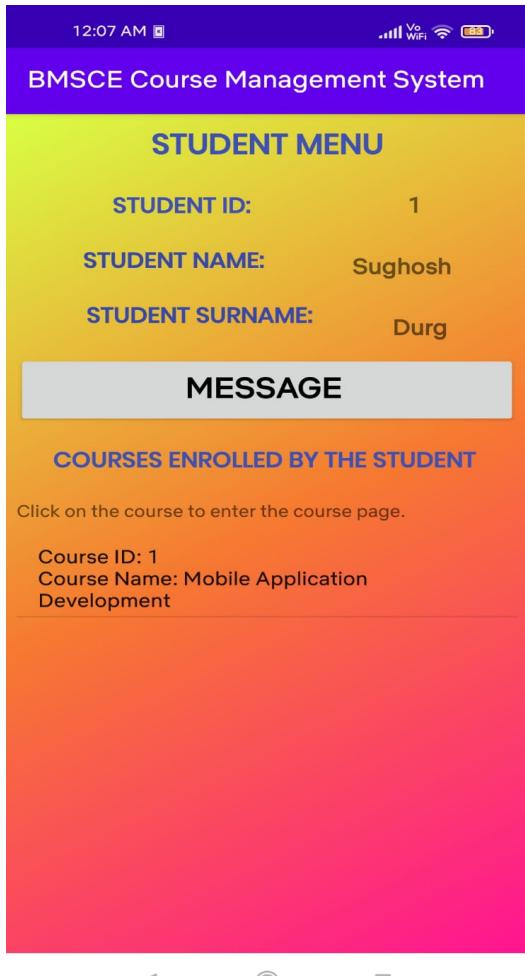


Fig 4.5 Student Dashboard

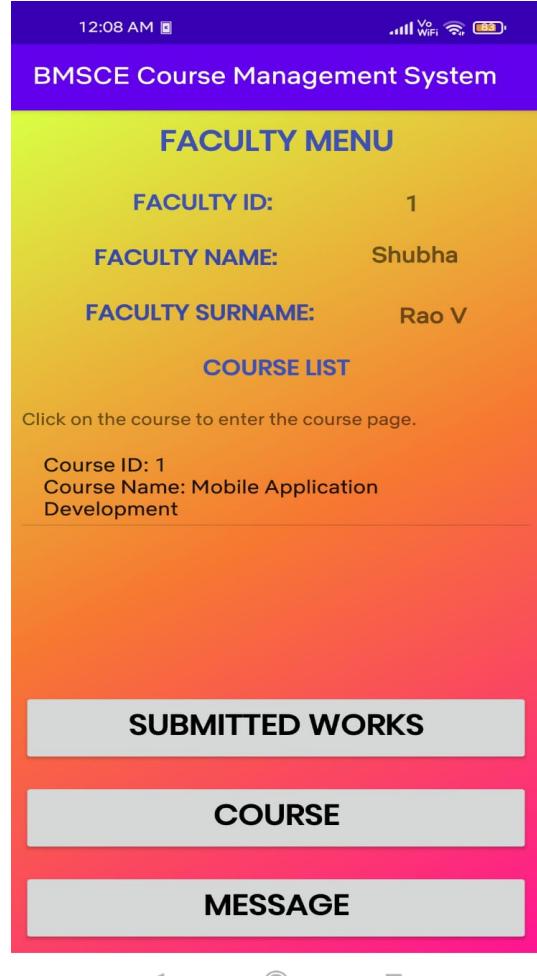


Fig 4.6 Faculty Dashboard

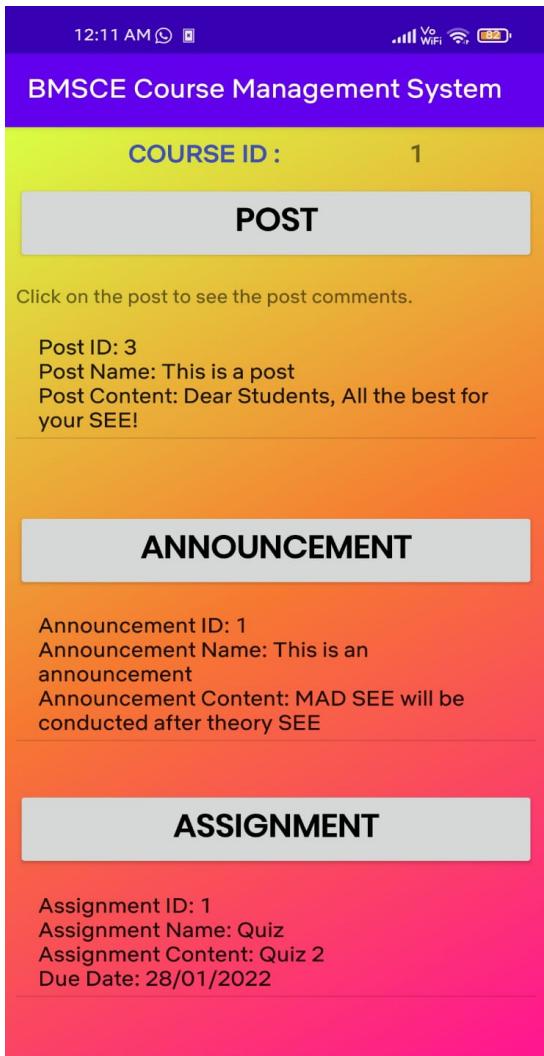


Fig 4.7 Course Updates

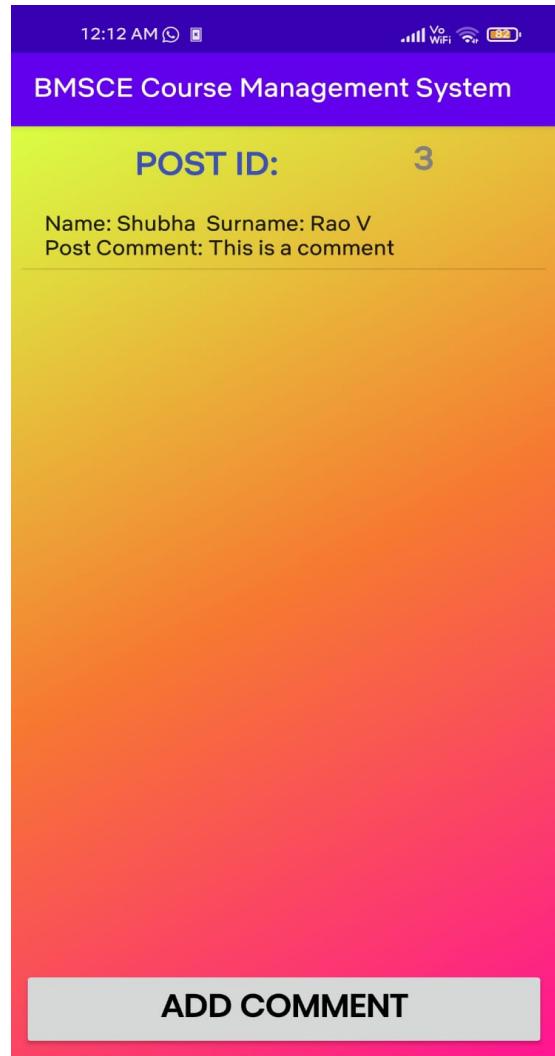


Fig 4.8 Comment on a Post

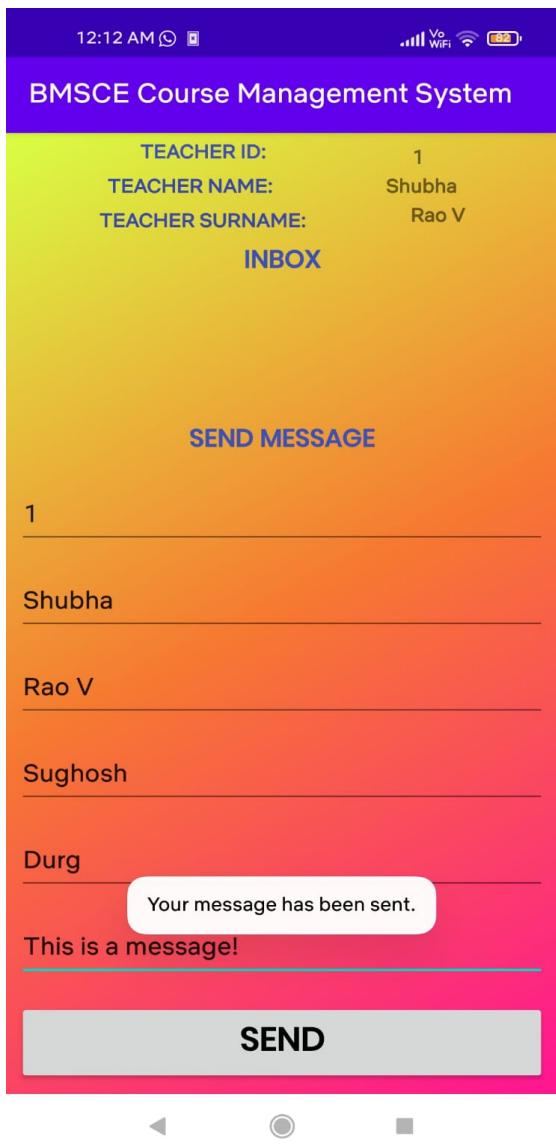


Fig 4.9 Sending a Message

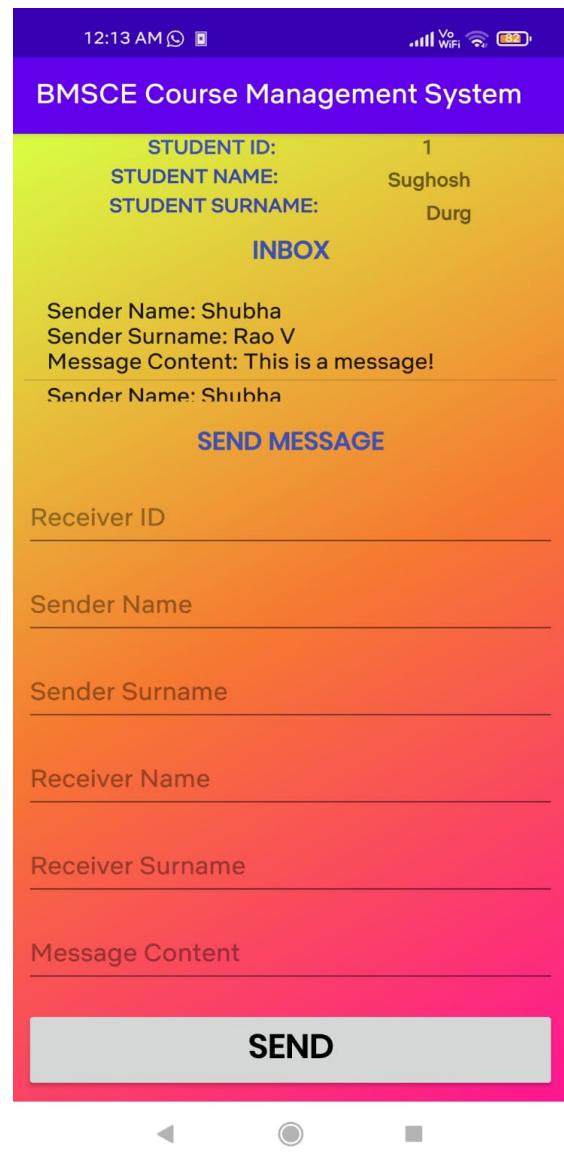


Fig 4.10 Receiving the Message

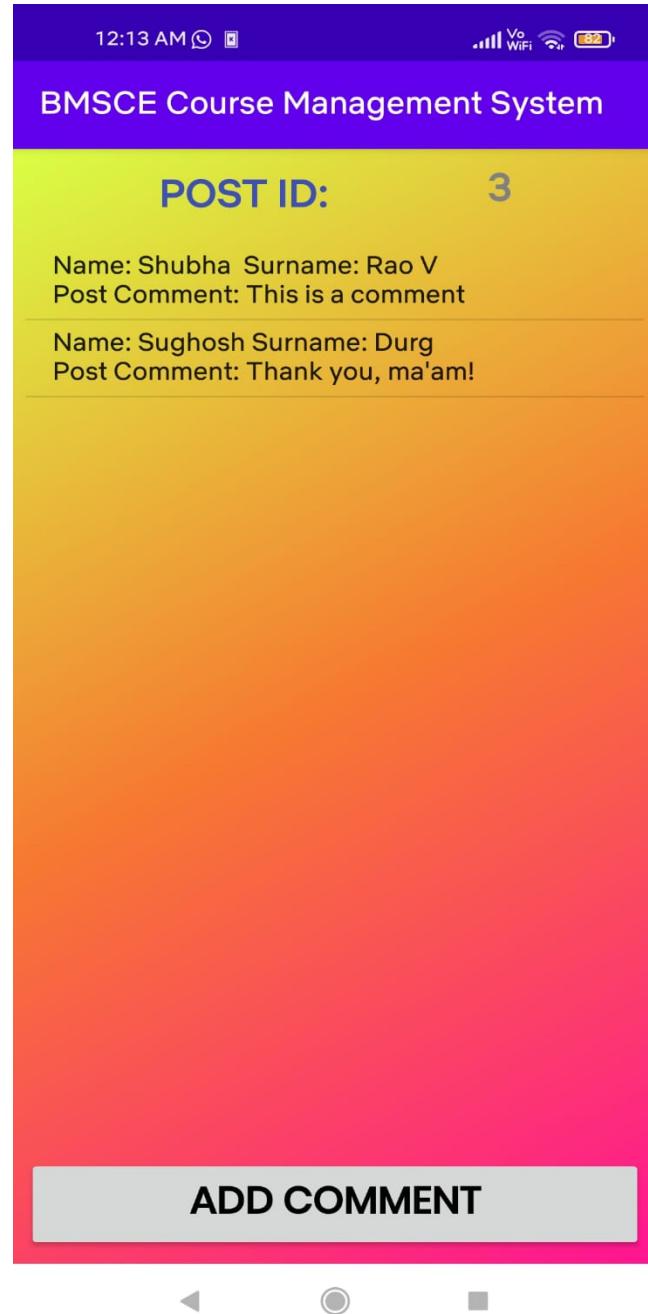


Fig 4.11 Replying to a Comment

TEST SUITE

> Login

Test Scenario ID	Login-1	Test Case ID	Login-1A				
Test Case Description	Login- Successful Case	Test Priority	High				
Prerequisite	Valid User Account	Post-Requisite	Nil				
Test Execution Steps:							
S.No	Action	Inputs/Actions	Expected Output	Actual Output	Test Phone	Test Result	Test Comments
1.	Launch the Application	Press the app icon	App Start Screen Displayed	App Start Screen Displayed	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 1:20 PM]: Launch successful
2.	Navigate to Login Form and enter credentials	Faculty ID and Student ID Password-*****	Login Successful	Authentication Successful	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 1:40 PM]: Login successful

Test Scenario ID	Login-1	Test Case ID	Login-1B				
Test Case Description	Login- Unsuccessful Case	Test Priority	High				
Prerequisite	invalid User Account	Post-Requisite	Nil				
Test Execution Steps:							
S.No	Action	Inputs/Actions	Expected Output	Actual Output	Test Phone	Test Result	Test Comments
1.	Launch the Application	Press the app icon	App Start Screen Displayed	App Start Screen Displayed	Poco F1	Pass	[Roshan Warrier - 20/01/22 2:20 PM]: Launch successful
2.	Navigate to Login Form and enter credentials	Faculty ID and Student ID Password-*****	Login Failed	Authentication Successful	Poco F1	Pass	[Roshan Warrier - 20/01/22 2:30 PM]: Login Failed

➤ Sending and Receiving a Message

Test Scenario ID	Message - 1	Test Case ID	Communication-1
Test Case Description	Send and Receive Message	Test Priority	Very High
Prerequisite	Valid user account and enrolled in a class.	Post-Requisite	Check inbox

Test Execution Steps:

S.No	Action	Inputs/Actions	Expected Output	Actual Output	Test Phone	Test Result	Test Comments
1.	Launch the Application	Press the app icon	App Start Screen Displayed	App Start Screen Displayed	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 5:20 PM]: Launch successful
2.	Login as a Student or a Faculty	Faculty ID and Student ID Password-*****	Login Successful	Authentication Successful	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 5:30 PM]: Login Successful
3.	Send a message as a student or a faculty	Receiver ID, Sender Name, Receiver Name and the Message	Message is sent successfully	Message is sent successfully	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 5:40 PM]: Message sent
4.	Login as a Student or a Faculty	Faculty ID and Student ID Password-*****	Login Successful	Authentication Successful	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 5:45 PM]: Login Successful
5.	Check the inbox	Navigate to the inbox	Message received	Message received	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 5:50 PM]: Message Received

> Adding, Editing, and Deleting a Course

Test Scenario ID	Course-1	Test Case ID	CourseManagement
Test Case Description	Add, Edit, and Delete a course	Test Priority	High
Prerequisite	Valid user account and teacher handling a class.	Post-Requisite	Nil

Test Execution Steps:

S.No	Action	Inputs/Actions	Expected Output	Actual Output	Test Phone	Test Result	Test Comments
1.	Launch the Application	Press the app icon	App Start Screen Displayed	App Start Screen Displayed	Poco F1	Pass	[Roshan Warrier - 20/01/22 7:20 PM]: Launch successful
2.	Login as a Faculty	Faculty ID and Password-*****	Login Successful	Authentication Successful	Poco F1	Pass	[Roshan Warrier - 20/01/22 7:25 PM]: Login Successful
3.	Add a course	Course ID, Student 1, Student 2, and Student 3	Course added successfully	Course added successfully	Poco F1	Pass	[Roshan Warrier - 20/01/22 7:30 PM]: Course added
4.	Edit a course	Course ID	Course edited successfully	Course edited successfully	Poco F1	Pass	[Roshan Warrier - 20/01/22 7:40 PM]: Course edited
5.	Delete a course	Course ID	Course deleted successfully	Course deleted successfully	Poco F1	Pass	[Roshan Warrier - 20/01/22 7:50 PM]: Course deleted

> Creating and Uploading an Assignment

Test Scenario ID	Assignment-01	Test Case ID	assignment1
Test Case Description	Creating and uploading an assignment	Test Priority	High
Prerequisite	Valid user account, Student enrolled in the class and teacher handling the class.	Post-Requisite	Grade the assignment

Test Execution Steps:

S.No	Action	Inputs/Actions	Expected Output	Actual Output	Test Phone	Test Result	Test Comments
1.	Launch the Application	Press the app icon	App Start Screen Displayed	App Start Screen Displayed	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 9:20 PM]: Launch successful
2.	Login as a Faculty	Faculty ID and Password-*****	Login Successful	Authentication Successful	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 9:25 PM]: Login Successful
3.	Create an assignment	Course ID, Assignment name, Assignment Content and Due Date	Assignment created successfully	Assignment created successfully	Xiaomi Redmi Note 9	Pass	[Sughosh Durg- 20/01/22 9:40 PM]: Assignment Created
4.	Login as a Student	Student ID and Password-*****	Login Successful	Authentication Successful	Xiaomi Redmi Note 9	Pass	[Sughosh Durg - 20/01/22 9:55 PM]: Login Successful
5.	Upload the assignment as a PDF or a doc file	Navigate to course menu	Assignment uploaded successfully	Error in uploading PDF and DOC file	Xiaomi Redmi Note 9	Fail	[Sughosh Durg - 20/01/22 10:00 PM]: Upload failed

REFERENCES

1. <https://www.geeksforgeeks.org/how-to-create-and-add-data-to-sqlite-database-in-android/>
2. <https://www.studytonight.com/post/gradient-drawable-in-android-create-gradient-backgrounds>
3. <https://developer.android.com/reference/android/widget/ArrayAdapter>
4. <https://stackoverflow.com/questions/736571/using-intent-in-an-android-application-to-how-another-activity>
5. https://www.tutorialspoint.com/android/android_user_interface_layouts.htm
6. <https://developer.android.com/reference/android/widget/ListView>
7. <https://www.lucidchart.com/pages/uml-use-case-diagram>
8. <https://developer.android.com/reference/android/util/Xml>