Dharmsinh Desai University

Department of MCA

Semester - III

Mobile Application Development

Termwork Report on

Appointment Scheduler

Submitted by

Roll No	**Name**
MCA005	**Nikhil Lathiya**
MCA015	**Vivek Makwana**

PROJECT TITLE

Appointment Scheduler

PROJECT DEFINITION

The Appointment Scheduler is a user-friendly mobile application designed to streamline the process of booking, managing, and viewing appointments. Whether for personal, medical, or professional use, this app provides an efficient way for users to organize their schedules, ensuring they never miss an important meeting or consultation.

Technologies:

- Android
- SQLite Database

Android Version:

Jellyfish // 2023.3.1
compileSdk = 34
minSdk = 24
targetSdk = 34

Main Functionality:

The Appointment Scheduler facilitates effective management of appointments, includes functionalities for inserting, updating, deleting, displaying, and searching appointments.

Codes:

MAD_PROJECT:

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:tools="http://schemas.android.com/tools">
```

```
<application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@drawable/logo"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.MAD_Project"
    tools:targetApi="31">

    <activity
        android:name=".viewappointment"
        android:exported="false" />
    <activity
        android:name=".appointment"
        android:exported="false" />
    <activity
        android:name=".login"
        android:exported="false" />
    <activity
        android:name=".register"
        android:exported="true">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
          </intent-filter>
        </activity>
    </application>
</manifest>
```

activity_login.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/background_color"
    tools:context=".login">

    <TextView
        android:id="@+id/appName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Appointment Scheduler"
        android:textSize="24sp"
        android:textStyle="bold"
        android:textColor="@color/primary_color"
        app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintBottom_toTopOf="@id/pageName"
    android:layout_marginTop="24dp"/>

<TextView
    android:id="@+id/pageName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Login"
    android:textSize="30sp"
    android:textStyle="bold"
    android:textColor="@color/black"
    app:layout_constraintTop_toBottomOf="@id/appName"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="30dp"
    android:layout_marginBottom="16dp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp"
    android:background="@drawable/form_background"
    app:layout_constraintTop_toBottomOf="@id/pageName"
    app:layout_constraintStart_toStartOf="parent"
```

```
app:layout_constraintEnd_toEndOf="parent"
android:layout_marginHorizontal="16dp"
android:elevation="4dp">
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Username"
    android:textSize="16sp"
    android:textColor="@color/text_color"
    android:layout_marginBottom="4dp"/>

<EditText
    android:id="@+id/username"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter your username"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Password"
    android:textSize="16sp"
    android:textColor="@color/text_color"
    android:layout_marginTop="16dp"
    android:layout_marginBottom="4dp"/>

<EditText
    android:id="@+id/password"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter your password"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:inputType="textPassword"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"/>

<Button
    android:id="@+id/loginBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Login"
    android:backgroundTint="@color/button_color"
    android:textColor="@android:color/white"
    android:layout_marginTop="16dp"
    android:elevation="2dp"/>
```

```
        <Button
            android:id="@+id/registerBtn"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Register"
            android:backgroundTint="@color/button_color"
            android:textColor="@android:color/white"
            android:layout_marginTop="8dp"
            android:elevation="2dp"/>
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

login.java:

```
package com.example.mad_project;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class login extends AppCompatActivity {
    private EditText username, password;
    private DatabaseHelper dbHelper;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
       username = findViewById(R.id.username);
        password = findViewById(R.id.password);
        Button loginButton = findViewById(R.id.loginBtn);
        Button registerButton = findViewById(R.id.registerBtn);
        dbHelper = new DatabaseHelper(this);
        loginButton.setOnClickListener(v -> loginUser());
        registerButton.setOnClickListener(v -> startActivity(new Intent(this,
register.class)));
    }
    private void loginUser() {
        String user = username.getText().toString().trim();
        String pass = password.getText().toString().trim();

        if (user.isEmpty() || pass.isEmpty()) {
            Toast.makeText(this, "Please fill in all fields",
Toast.LENGTH_SHORT).show();
            return;
        }
        if (dbHelper.checkUser(user, pass)) {
                      Integer userId = dbHelper.getUserId(user);
            if (userId != null) {
                SharedPreferences sharedPreferences =
getSharedPreferences("user_prefs", MODE_PRIVATE);
                SharedPreferences.Editor editor = sharedPreferences.edit();
                editor.putInt("userId", userId);
                editor.apply();
                Toast.makeText(this, "Login successful",
Toast.LENGTH_SHORT).show();
                              startActivity(new Intent(this, appointment.class));
                finish();
            }
        } else {
            Toast.makeText(this, "Invalid username or password",
Toast.LENGTH_SHORT).show();
        }
    }
}
```

activity_register.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".register"
    android:background="@color/background_color">


    <TextView
        android:id="@+id/appName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Appointment Scheduler"
        android:textSize="24sp"
        android:textStyle="bold"
        android:textColor="@color/primary_color"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintBottom_toTopOf="@id/pageName"
        android:layout_marginTop="24dp"/>

    <TextView
        android:id="@+id/pageName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Register"
        android:textSize="30sp"
        android:textStyle="bold"
        android:textColor="@color/black"
        app:layout_constraintTop_toBottomOf="@id/appName"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="30dp"
        android:layout_marginBottom="16dp"/>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp"
    android:background="@drawable/form_background"
    app:layout_constraintTop_toBottomOf="@id/pageName"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginHorizontal="16dp"
    android:elevation="4dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Username"
        android:textSize="16sp"
        android:textColor="@color/text_color"
        android:layout_marginBottom="4dp"/>

    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter your username"
```

```xml
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Password"
    android:textSize="16sp"
    android:textColor="@color/text_color"
    android:layout_marginTop="16dp"
    android:layout_marginBottom="4dp"/>

<EditText
    android:id="@+id/password"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter your password"
    android:inputType="textPassword"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"/>
    <Button
      android:id="@+id/registerBtn"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Register"
      android:backgroundTint="@color/button_color"
      android:textColor="@android:color/white"
      android:layout_marginTop="16dp"
      android:elevation="2dp"/>

    <Button
      android:id="@+id/loginBtn"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Login"
      android:backgroundTint="@color/button_color"
      android:textColor="@android:color/white"
      android:layout_marginTop="8dp"
      android:elevation="2dp"/>
  </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

register.java:

```
package com.example.mad_project;
import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class register extends AppCompatActivity {
    private EditText usernameEditText, passwordEditText;
    private DatabaseHelper dbHelper;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);
        usernameEditText = findViewById(R.id.username);
        passwordEditText = findViewById(R.id.password);
        Button registerButton = findViewById(R.id.registerBtn);
        dbHelper = new DatabaseHelper(this);

        registerButton.setOnClickListener(v -> {
            String username = usernameEditText.getText().toString();
            String password = passwordEditText.getText().toString();
            if (!username.isEmpty() && !password.isEmpty()) {
                boolean isAdded = dbHelper.addUser(username, password);
                if (isAdded) {
                    Toast.makeText(this, "Registration Successful!",
Toast.LENGTH_SHORT).show();
                    Intent intent = new Intent(register.this, login.class);
                    startActivity(intent);
                    //finish();
                } else {
                    Toast.makeText(this, "Credentials already exist!",
Toast.LENGTH_SHORT).show();
                }
            } else {
                Toast.makeText(this, "Please fill all fields!",
Toast.LENGTH_SHORT).show();
            }
        });

        Button loginButton = findViewById(R.id.loginBtn);
        loginButton.setOnClickListener(v -> {
            Intent intent = new Intent(register.this, login.class);
            startActivity(intent);
        });
    }
}
```

activity_appointment.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

```
android:background="@color/background_color"
tools:context=".appointment">

<ScrollView
    android:layout_width="0dp"
    android:layout_height="0dp"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="16dp"
        android:clipToPadding="false"
        android:orientation="vertical"
        android:padding="16dp"
        android:background="@drawable/form_background"
        android:elevation="4dp">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="8dp"
    android:text="First Name"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
    android:id="@+id/firstName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter First Name"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="text"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Last Name"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
    android:id="@+id/lastName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Last Name"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="text"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Email"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Email"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="textEmailAddress" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Mobile No"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
    android:id="@+id/mobile"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Mobile Number"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="phone"
    android:maxLength="10"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Address"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
    android:id="@+id/address"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="top|start"
    android:hint="Enter Address"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="textMultiLine"
    android:minLines="2"
    android:scrollbars="vertical" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Date (YYYY-MM-DD)"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
```

```
    android:id="@+id/date"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:focusable="false"
    android:hint="Select Date"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="none" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Time (HH:MM)"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
    android:id="@+id/time"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:focusable="false"
    android:hint="Select Time"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="none"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="Appointment Subject"
    android:textSize="16sp"
    android:textColor="@color/text_color" />

<EditText
    android:id="@+id/subject"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Subject"
    android:background="@drawable/edit_text_background"
    android:padding="12dp"
    android:textColor="@color/text_color"
    android:textColorHint="@color/hint_color"
    android:inputType="text" />

<Button
    android:id="@+id/bookAppointmentBtn"
```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Book Appointment"
        android:textColor="@android:color/white"
        android:backgroundTint="@color/button_color"
        android:elevation="2dp"/>

      <Button
        android:id="@+id/viewBookedAppointmentsBtn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="View Booked Appointments"
        android:textColor="@android:color/white"
        android:backgroundTint="@color/button_color"
        android:elevation="2dp" />

    </LinearLayout>
  </ScrollView>
</androidx.constraintlayout.widget.ConstraintLayout>
```

appointement.java:

```
package com.example.mad_project;
import android.app.DatePickerDialog;
import android.app.TimePickerDialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import
com.google.android.material.floatingactionbutton.FloatingActionButton;
import java.util.Calendar;
public class appointment extends AppCompatActivity {
    private EditText firstName, lastName, email, mobile, date, time, subject,
address;
    private DatabaseHelper dbHelper;
      @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_appointment);
        // Initialize views
        firstName = findViewById(R.id.firstName);
        lastName = findViewById(R.id.lastName);
        email = findViewById(R.id.email);
        mobile = findViewById(R.id.mobile);
        date = findViewById(R.id.date);
        time = findViewById(R.id.time);
        subject = findViewById(R.id.subject);
        address = findViewById(R.id.address);
        Button bookAppointmentBtn = findViewById(R.id.bookAppointmentBtn);
        Button viewBookedAppointmentsBtn =
findViewById(R.id.viewBookedAppointmentsBtn);

        dbHelper = new DatabaseHelper(this);
        date.setOnClickListener(v -> showDatePicker());
        time.setOnClickListener(v -> showTimePicker());
        bookAppointmentBtn.setOnClickListener(v -> bookAppointment());
        viewBookedAppointmentsBtn.setOnClickListener(v ->
viewBookedAppointments());
    }
    private void showDatePicker() {
        Calendar calendar = Calendar.getInstance();
        int year = calendar.get(Calendar.YEAR);
        int month = calendar.get(Calendar.MONTH);
        int day = calendar.get(Calendar.DAY_OF_MONTH);

        DatePickerDialog datePickerDialog = new DatePickerDialog(this, (view,
selectedYear, selectedMonth, selectedDay) -> {
            // month start from 1
            String formattedDate = selectedYear + "-" + (selectedMonth + 1) + "-" +
selectedDay;
            date.setText(formattedDate);
        }, year, month, day);
        datePickerDialog.show();
    }
    private void showTimePicker() {
        Calendar calendar = Calendar.getInstance();
        int hour = calendar.get(Calendar.HOUR_OF_DAY);
        int minute = calendar.get(Calendar.MINUTE);

        TimePickerDialog timePickerDialog = new TimePickerDialog(this, (view,
selectedHour, selectedMinute) -> {
            String formattedTime = String.format("%02d:%02d", selectedHour,
selectedMinute);
            time.setText(formattedTime);
        }, hour, minute, true);
        timePickerDialog.show();
    }

    private void bookAppointment() {
        String fName = firstName.getText().toString();
        String lName = lastName.getText().toString();
        String emailText = email.getText().toString();
        String mobileText = mobile.getText().toString();
        String addressText = address.getText().toString();
        String dateText = date.getText().toString();
        String timeText = time.getText().toString();
        String subjectText = subject.getText().toString();

        if (fName.isEmpty() || lName.isEmpty() || emailText.isEmpty() ||
mobileText.isEmpty() || addressText.isEmpty()
                || dateText.isEmpty() || timeText.isEmpty() || subjectText.isEmpty()) {
```

```
            Toast.makeText(this, "Please fill all fields!", Toast.LENGTH_SHORT).show();
            return;
        }
        SharedPreferences sharedPreferences = getSharedPreferences("user_prefs", MODE_PRIVATE);
        int userId = sharedPreferences.getInt("userId", -1);
              if (userId == -1) {
            Toast.makeText(this, "User not logged in.", Toast.LENGTH_SHORT).show();
            return;
        }
boolean isInserted = dbHelper.addAppointment(fName, lName, emailText, mobileText,addressText, dateText, timeText, subjectText, userId);
        if (isInserted) {
            Toast.makeText(this, "Appointment Booked Successfully!", Toast.LENGTH_SHORT).show();
            clear();
        } else {
            Toast.makeText(this, "Failed to Book Appointment.", Toast.LENGTH_SHORT).show();
        }
    }
    private void viewBookedAppointments() {
         Intent intent = new Intent(this, viewappointment.class);
         startActivity(intent);
        Toast.makeText(this, "View Booked Appointments Clicked!", Toast.LENGTH_SHORT).show();
    }
    public void clear() {
        firstName.setText("");
        lastName.setText("");
        email.setText("");
        mobile.setText("");
        date.setText("");
        time.setText("");
        subject.setText("");
        address.setText("");
    }
}
```

activity_viewappointment.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".viewappointment"
    android:background="@android:color/white">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="16dp"
```

```
android:background="@android:color/white"
android:clipToPadding="false"
android:elevation="4dp">

<EditText
   android:id="@+id/appointmentIdInput"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:hint="Enter Appointment ID"
   android:inputType="number"
   android:layout_marginBottom="16dp"
   android:padding="12dp"
   android:background="@drawable/edittext_background"
   android:textColor="@color/black"
   android:textColorHint="@color/black"
   android:textSize="16sp"
   android:elevation="2dp"/>

<Button
   android:id="@+id/searchButton"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:text="Search Appointment"
   android:layout_marginBottom="20dp"
   android:textColor="@android:color/white"
   android:backgroundTint="@color/button_color"
```

```
    android:elevation="2dp"
    android:padding="12dp"
    android:textStyle="bold" />

<Button
    android:id="@+id/logout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Logout"
    android:layout_marginBottom="20dp"
    android:textColor="@android:color/white"
    android:backgroundTint="@color/button_color"
    android:elevation="2dp"
    android:padding="12dp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/appointmentDetails"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Appointment Details will appear here"
    android:textSize="16sp"
    android:padding="16dp"
    android:background="@drawable/textview_background"
    android:layout_marginBottom="16dp"
    android:textColor="@android:color/black"
```

```
            android:elevation="2dp" />

        <androidx.recyclerview.widget.RecyclerView
            android:id="@+id/recyc1"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@android:color/white"
            android:clipToPadding="false"
            android:elevation="2dp"/>
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

viewappointment.java:

```
package com.example.mad_project;
import android.content.Intent;
import android.content.SharedPreferences;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import java.util.ArrayList;
import java.util.List;
public class viewappointment extends AppCompatActivity {
   private EditText appointmentIdInput;
    private Button searchButton;
    private Button logout;
    private TextView appointmentDetails;
    private DatabaseHelper dbHelper;
    private RecyclerView recyc;
    private DatabaseHelper db;
    private List<AppointmentModel> appointmentList;
    private AppointmentAdapter adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_viewappointment);
        appointmentIdInput = findViewById(R.id.appointmentIdInput);
        searchButton = findViewById(R.id.searchButton);
        logout = findViewById(R.id.logout);
        appointmentDetails = findViewById(R.id.appointmentDetails);
        dbHelper = new DatabaseHelper(this);

        recyc = findViewById(R.id.recyc1);
        recyc.setLayoutManager(new LinearLayoutManager(this));
        db = new DatabaseHelper(this);
        appointmentList = new ArrayList<>();
        SharedPreferences sharedPreferences =
getSharedPreferences("user_prefs", MODE_PRIVATE);
        int userId = sharedPreferences.getInt("userId", -1);
        fetchAppointments(userId);
        adapter = new AppointmentAdapter(appointmentList,dbHelper,this);
        recyc.setAdapter(adapter);
        searchButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String appointmentIdStr =
appointmentIdInput.getText().toString().trim();
                if (appointmentIdStr.isEmpty()) {
                    Toast.makeText(viewappointment.this, "Please enter an
appointment ID", Toast.LENGTH_SHORT).show();
                } else {
                    int appointmentId = Integer.parseInt(appointmentIdStr);
                    filterAppointmentsById(appointmentId);
                }
            }
        });

        logout.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(viewappointment.this, register.class);
                startActivity(intent);
                Toast.makeText(viewappointment.this, "Logged Out!!", Toast.LENGTH_SHORT).show();
                finish();
            }
        });
    }
    private void fetchAppointments(int userId) {
        Cursor cursor = db.getAppointmentsByUserId(userId);
        if (cursor != null && cursor.moveToFirst()) {
            do {
                int _appid = cursor.getInt(cursor.getColumnIndexOrThrow("appointment_id"));
                String firstName = cursor.getString(cursor.getColumnIndex("first_name"));
                String lastName = cursor.getString(cursor.getColumnIndex("last_name"));
                String email = cursor.getString(cursor.getColumnIndex("email"));
                String date = cursor.getString(cursor.getColumnIndex("date"));
                String time = cursor.getString(cursor.getColumnIndex("time"));
                String subject = cursor.getString(cursor.getColumnIndex("subject"));

                appointmentList.add(new AppointmentModel(_appid, firstName, lastName, email, date, time, subject));
            } while (cursor.moveToNext());
            cursor.close();
            // adapter.notifyDataSetChanged();
        } else {
            if (cursor != null) {
```

```
        cursor.close();
      }
      Toast.makeText(this, "No appointments found",
Toast.LENGTH_SHORT).show();
    }
  }
  private void filterAppointmentsById(int appointmentId) {
    Cursor cursor = dbHelper.getAppointmentById(appointmentId);
    if (cursor != null && cursor.moveToFirst()) {
      appointmentList.clear();
      do {
        int _appid =
cursor.getInt(cursor.getColumnIndexOrThrow("appointment_id"));
        String firstName =
cursor.getString(cursor.getColumnIndex("first_name"));
        String lastName =
cursor.getString(cursor.getColumnIndex("last_name"));
        String email = cursor.getString(cursor.getColumnIndex("email"));
        String date = cursor.getString(cursor.getColumnIndex("date"));
        String time = cursor.getString(cursor.getColumnIndex("time"));
        String subject = cursor.getString(cursor.getColumnIndex("subject"));
        appointmentList.add(new AppointmentModel(_appid, firstName,
lastName, email, date, time, subject));
      } while (cursor.moveToNext());
      cursor.close();
      adapter.notifyDataSetChanged();
    } else {
      if (cursor != null) {
```

```
                cursor.close();
            }
            Toast.makeText(this, "No appointment found with ID: " + appointmentId, Toast.LENGTH_SHORT).show();
        }
    }
}
```

appointment_item_list.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:elevation="6dp"
    app:cardCornerRadius="8dp"
    app:cardBackgroundColor="@android:color/white">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="16dp">

        <TextView
            android:id="@+id/firstName1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="First Name"
            android:textStyle="bold"
            android:textSize="16sp"
            android:textColor="@android:color/black" />

        <TextView
            android:id="@+id/lastName1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Last Name"
            android:textSize="15sp"
            android:textColor="@android:color/black"/>

        <TextView
            android:id="@+id/email1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Email"
            android:textSize="15sp"
            android:textColor="@android:color/black"/>

        <TextView
            android:id="@+id/date1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Date"
            android:textSize="15sp"
            android:textColor="@android:color/black"/>

        <TextView
            android:id="@+id/time1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Time"
            android:textSize="15sp"
            android:textColor="@android:color/black"/>

        <TextView
            android:id="@+id/subject1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Subject"
            android:textSize="15sp"
            android:textColor="@android:color/black"/>

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_marginTop="16dp"
```

```
            android:gravity="end">

            <Button
                android:id="@+id/delete"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Delete"
                android:textColor="@android:color/white"
                android:backgroundTint="@color/colorPrimary"
                android:layout_marginEnd="8dp"
                android:padding="8dp"
                android:elevation="2dp" />

            <Button
                android:id="@+id/update"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Update"
                android:textColor="@android:color/white"
                android:backgroundTint="@color/colorPrimary"
                android:padding="8dp"
                android:elevation="2dp"/>
        </LinearLayout>
    </LinearLayout>
</androidx.cardview.widget.CardView>
```

AppointmentAdapter.java:

```
package com.example.mad_project;

import android.content.Context;

import android.content.DialogInterface;

import android.view.LayoutInflater;

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 androidx.annotation.NonNull;

import androidx.appcompat.app.AlertDialog;

import androidx.recyclerview.widget.RecyclerView;

import java.util.List;


public class AppointmentAdapter extends
RecyclerView.Adapter<AppointmentAdapter.MyViewHolder> {

    private List<AppointmentModel> appointmentList;

    DatabaseHelper db;

    Context context;

    public AppointmentAdapter(List<AppointmentModel> appointmentList,
DatabaseHelper db, Context context) {

        this.appointmentList = appointmentList;

        this.db = db;

        this.context = context;

    }
    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.appointment_item_list, parent, false);
        return new MyViewHolder(view);
    }
    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {
        AppointmentModel appointment = appointmentList.get(position);
        holder.firstName.setText(appointment.getFirstName());
        holder.lastName.setText(appointment.getLastName());
        holder.email.setText(appointment.getEmail());
        holder.date.setText(appointment.getDate());
        holder.time.setText(appointment.getTime());
        holder.subject.setText(appointment.getSubject());

        holder.delete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                int position = holder.getAdapterPosition();
                if (position != RecyclerView.NO_POSITION) {
                    int recid = appointmentList.get(position).getId();
```

```
          new AlertDialog.Builder(context)
              .setTitle("Delete Appointment")
              .setMessage("Are you sure you want to delete this
appointment?")
              .setCancelable(false)
              .setPositiveButton("Yes", new DialogInterface.OnClickListener()
{
                @Override
                public void onClick(DialogInterface dialog, int which) {
                  if (db.deleteAppointment(recid)) {
                    appointmentList.remove(position);
                    notifyItemRemoved(position);
                    Toast.makeText(context, "Deleted.",
Toast.LENGTH_SHORT).show();
                  } else {
                    Toast.makeText(context, "Delete failed.",
Toast.LENGTH_SHORT).show();
                  }
                }
              })
              .setNegativeButton("No", null)
              .show();
        }
      }
    });
    holder.update.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
AlertDialog.Builder builder = new AlertDialog.Builder(context);
View dialogView =
LayoutInflater.from(context).inflate(R.layout.dialog_update_appointment,
null);
        builder.setView(dialogView);


        EditText updateFirstName =
dialogView.findViewById(R.id.updateFirstName);
        EditText updateLastName =
dialogView.findViewById(R.id.updateLastName);
        EditText updateEmail = dialogView.findViewById(R.id.updateEmail);
        EditText updateDate = dialogView.findViewById(R.id.updateDate);
        EditText updateTime = dialogView.findViewById(R.id.updateTime);
        EditText updateSubject =
dialogView.findViewById(R.id.updateSubject);
        Button updateButton = dialogView.findViewById(R.id.updateButton);


        updateFirstName.setText(appointment.getFirstName());
        updateLastName.setText(appointment.getLastName());
        updateEmail.setText(appointment.getEmail());
        updateDate.setText(appointment.getDate());
        updateTime.setText(appointment.getTime());
        updateSubject.setText(appointment.getSubject());
```

```
            AlertDialog dialog = builder.create();
            dialog.show();
            updateButton.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    String newFirstName = updateFirstName.getText().toString();
                    String newLastName = updateLastName.getText().toString();
                    String newEmail = updateEmail.getText().toString();
                    String newDate = updateDate.getText().toString();
                    String newTime = updateTime.getText().toString();
                    String newSubject = updateSubject.getText().toString();
                    boolean isUpdated = db.updateAppointment(appointment.getId(), newFirstName, newLastName, newEmail, newDate, newTime, newSubject);
                    if (isUpdated) {

                        appointmentList.set(position, new AppointmentModel(appointment.getId(), newFirstName, newLastName, newEmail, newDate, newTime, newSubject));
                        notifyItemChanged(position);
                        Toast.makeText(context, "Appointment Updated", Toast.LENGTH_SHORT).show();
                    } else {
                        Toast.makeText(context, "Update Failed", Toast.LENGTH_SHORT).show();
                    }
                    dialog.dismiss();
                }
            });
        }
    });
        }
@Override
public int getItemCount() {
    return appointmentList.size();
}
public static class MyViewHolder extends RecyclerView.ViewHolder {
    TextView firstName, lastName, email, date, time, subject;
    Button delete,update;

    public MyViewHolder(@NonNull View itemView) {
        super(itemView);
        firstName = itemView.findViewById(R.id.firstName1);
        lastName = itemView.findViewById(R.id.lastName1);
        email = itemView.findViewById(R.id.email1);
        date = itemView.findViewById(R.id.date1);
        time = itemView.findViewById(R.id.time1);
        subject = itemView.findViewById(R.id.subject1);
        delete = itemView.findViewById(R.id.delete);
        update = itemView.findViewById(R.id.update);
    }
}
}
```

dialog_update_appointment.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp"
    android:background="@android:color/white"
    android:layout_margin="16dp"
    android:elevation="4dp"
    android:backgroundTint="@android:color/white">

    <EditText
        android:id="@+id/updateFirstName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="First Name"
        android:padding="12dp"
        android:textSize="16sp"
        android:textColor="@color/black"
        android:background="@drawable/edittext2_background"
        android:layout_marginBottom="8dp"/>

    <EditText
        android:id="@+id/updateLastName"
        android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Last Name"
    android:padding="12dp"
    android:textSize="16sp"
    android:textColor="@color/black"
    android:background="@drawable/edittext2_background"
    android:layout_marginBottom="8dp"/>

<EditText
    android:id="@+id/updateEmail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Email"
    android:padding="12dp"
    android:textSize="16sp"
    android:textColor="@color/black"
    android:background="@drawable/edittext2_background"
    android:layout_marginBottom="8dp"/>

<EditText
    android:id="@+id/updateDate"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Date"
    android:padding="12dp"
    android:textSize="16sp"
    android:textColor="@color/black"

    android:background="@drawable/edittext2_background"

    android:layout_marginBottom="8dp"/>


<EditText

    android:id="@+id/updateTime"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:hint="Time"

    android:padding="12dp"

    android:textSize="16sp"

    android:textColor="@color/black"

    android:background="@drawable/edittext2_background"

    android:layout_marginBottom="8dp"/>


<EditText

    android:id="@+id/updateSubject"

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:hint="Subject"

    android:padding="12dp"

    android:textSize="16sp"

    android:textColor="@color/black"

    android:background="@drawable/edittext2_background"

    android:layout_marginBottom="8dp"/>
    <Button
        android:id="@+id/updateButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Update"
        android:textColor="@android:color/white"
        android:backgroundTint="@color/colorPrimary"
        android:padding="12dp"
        android:textSize="16sp"
        android:elevation="2dp"/>
</LinearLayout>
```

Appointment.java:

```
package com.example.mad_project;
public class AppointmentModel {
    private int id;
    private String firstName, lastName, email, date, time, subject;
    public AppointmentModel(int id,String firstName, String lastName, String email, String date, String time, String subject) {
        this.id = id;
        this.firstName = firstName;
        this.lastName = lastName;
        this.email = email;
        this.date = date;
        this.time = time;
        this.subject = subject;


    }
```

```
   public int getId() {
        return id;
    }
    public String getFirstName() {
        return firstName;
    }
    public String getLastName() {
        return lastName;
    }
    public String getEmail() {
        return email;
    }
    public String getDate() {
        return date;
    }
    public String getTime() {
        return time;
    }
    public String getSubject() {
        return subject;
    }
}
```

DatabaseHelper.java:

```
package com.example.mad_project;
import android.annotation.SuppressLint;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
public class DatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "users.db";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_USERS = "users";
    private static final String TABLE_APPOINTMENTS = "appointments";
    private static final String COLUMN_ID = "id";
    private static final String COLUMN_USERNAME = "username";
    private static final String COLUMN_PASSWORD = "password";

    // Appointment columns
    private static final String COLUMN_APPOINTMENT_ID = "appointment_id";
    private static final String COLUMN_FIRST_NAME = "first_name";
    private static final String COLUMN_LAST_NAME = "last_name";
    private static final String COLUMN_EMAIL = "email";
    private static final String COLUMN_MOBILE = "mobile";
    private static final String COLUMN_ADDRESS = "address";
    private static final String COLUMN_DATE = "date";
    private static final String COLUMN_TIME = "time";
    private static final String COLUMN_SUBJECT = "subject";
    private static final String COLUMN_USER_ID = "user_id";
    SQLiteDatabase db;
    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_USERS_TABLE = "CREATE TABLE " + TABLE_USERS + " (" +
                COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
                COLUMN_USERNAME + " TEXT UNIQUE, " +
                COLUMN_PASSWORD + " TEXT)";
        db.execSQL(CREATE_USERS_TABLE);
        String CREATE_APPOINTMENTS_TABLE = "CREATE TABLE " + TABLE_APPOINTMENTS + " (" +
                COLUMN_APPOINTMENT_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
                COLUMN_FIRST_NAME + " TEXT, " +
                COLUMN_LAST_NAME + " TEXT, " +
                COLUMN_EMAIL + " TEXT, " +
                COLUMN_MOBILE + " TEXT, " +
                COLUMN_ADDRESS + " TEXT, " +
                COLUMN_DATE + " TEXT, " +
                COLUMN_TIME + " TEXT, " +
                COLUMN_SUBJECT + " TEXT, " +
                COLUMN_USER_ID + " INTEGER, " +
                "FOREIGN KEY(" + COLUMN_USER_ID + ") REFERENCES " +
TABLE_USERS + "(" + COLUMN_ID + "))";
        db.execSQL(CREATE_APPOINTMENTS_TABLE);
    }
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_APPOINTMENTS);
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_USERS);
        onCreate(db);
    }
    public boolean addUser(String username, String password) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_USERNAME, username);
        values.put(COLUMN_PASSWORD, password);

        if (checkUser(username, password)) {
            return false;
        }
        long result = db.insert(TABLE_USERS, null, values);
        db.close();
        return result != -1;
    }
    public boolean checkUser(String username, String password) {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.query(TABLE_USERS, null, COLUMN_USERNAME + "=?
AND " + COLUMN_PASSWORD + "=?",
            new String[]{username, password}, null, null, null);
        boolean exists = cursor.getCount() > 0;
        cursor.close();
        return exists;
    }
    public boolean addAppointment(String firstName, String lastName, String
email, String mobile,  String address,
        String date, String time, String subject, int userId) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_FIRST_NAME, firstName);
        values.put(COLUMN_LAST_NAME, lastName);
        values.put(COLUMN_EMAIL, email);
        values.put(COLUMN_MOBILE, mobile);
        values.put(COLUMN_ADDRESS, address);
        values.put(COLUMN_DATE, date);
        values.put(COLUMN_TIME, time);
        values.put(COLUMN_SUBJECT, subject);
        values.put(COLUMN_USER_ID, userId);

        long result = db.insert(TABLE_APPOINTMENTS, null, values);
        db.close();
        return result != -1;
    }
    public Integer getUserId(String username) {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.query(TABLE_USERS, new String[]{COLUMN_ID}, COLUMN_USERNAME + "=?", new String[]{username}, null, null, null);

        if (cursor != null && cursor.moveToFirst()) {

            int userId = cursor.getInt(cursor.getColumnIndex(COLUMN_ID));

            cursor.close();

            return userId;

        }

        return null;

    }

    public Cursor getAppointmentsByUserId(int userId) {

        SQLiteDatabase db = this.getReadableDatabase();

        return db.query(TABLE_APPOINTMENTS, null, COLUMN_USER_ID + "=?", new String[]{String.valueOf(userId)}, null, null, COLUMN_DATE + " ASC");

    }

public Cursor getAppointmentById(int appointmentId) {

    SQLiteDatabase db = this.getReadableDatabase();

    return db.query("appointments", null, "appointment_id=?", new String[]{String.valueOf(appointmentId)}, null, null, null);

}

    public boolean deleteAppointment(int appointmentId)

    {

        SQLiteDatabase db = this.getWritableDatabase();

        int ans = db.delete(TABLE_APPOINTMENTS,COLUMN_APPOINTMENT_ID + " = ? ", new String[]{String.valueOf(appointmentId)});

        db.close();

        return  ans>0;

    }
```

```
   public boolean updateAppointment(int id, String firstName, String lastName, String email, String date, String time, String subject) {
       SQLiteDatabase db = this.getWritableDatabase();
       ContentValues contentValues = new ContentValues();
       contentValues.put("first_name", firstName);
       contentValues.put("last_name", lastName);
       contentValues.put("email", email);
       contentValues.put("date", date);
       contentValues.put("time", time);
       contentValues.put("subject", subject);
       int result = db.update("appointments", contentValues, "appointment_id=?", new String[]{String.valueOf(id)});
       return result > 0;
   }
}
```

drawable files:

basline_add_24.xml:

```
<vector xmlns:android="http://schemas.android.com/apk/res/android"
         android:height="24dp" android:tint="#000000"
android:viewportHeight="24"    android:viewportWidth="24"
android:width="24dp">
         <path android:fillColor="@android:color/white"
         android:pathData="M19,13h-6v6h-2v-6H5v-2h6V5h2v6h6v2z"/>
   </vector>
```

edit_text_background.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
   <solid android:color="@android:color/white"/>
```

```
    <corners android:radius="8dp"/>
    <stroke android:width="1dp" android:color="@color/secondary_color"/>
</shape>
```

edittext_background.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <solid android:color="@android:color/white" />
    <corners android:radius="8dp" />
    <stroke android:width="1dp" android:color="@color/secondary_color" />
</shape>
```

edittext2_background.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <solid android:color="@android:color/white" />
    <corners android:radius="8dp" />
    <stroke android:width="1dp" android:color="@android:color/darker_gray" />
</shape>
```

form_background.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <solid android:color="@android:color/white"/>
    <corners android:radius="12dp"/>
    <stroke android:width="1dp" android:color="@color/secondary_color"/>
</shape>
```

logo.jpg:

textview_background.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
   <solid android:color="@android:color/white"/>
   <corners android:radius="12dp"/>
   <stroke android:width="1dp" android:color="@color/secondary_color"/>
</shape>
```

values:

colors.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
   <color name="black">#FF000000</color>
   <color name="white">#FFFFFFFF</color>
   <color name="primary_color">#6200EA</color>
   <color name="secondary_color">#03DAC5</color>
   <color name="background_color">#FFFFFF</color>
```

```
    <color name="text_color">#000000</color>
    <color name="hint_color">#757575</color>
    <color name="button_color">#6200EA</color>
    <color name="colorPrimary">#6200EE</color>
    <color name="colorPrimaryDark">#3700B3</color>
    <color name="colorAccent">#03DAC5</color>
</resources>
```

string.xml:

```
<resources>
    <string name="app_name">MAD_Project</string>
</resources>
```

themes.xml:

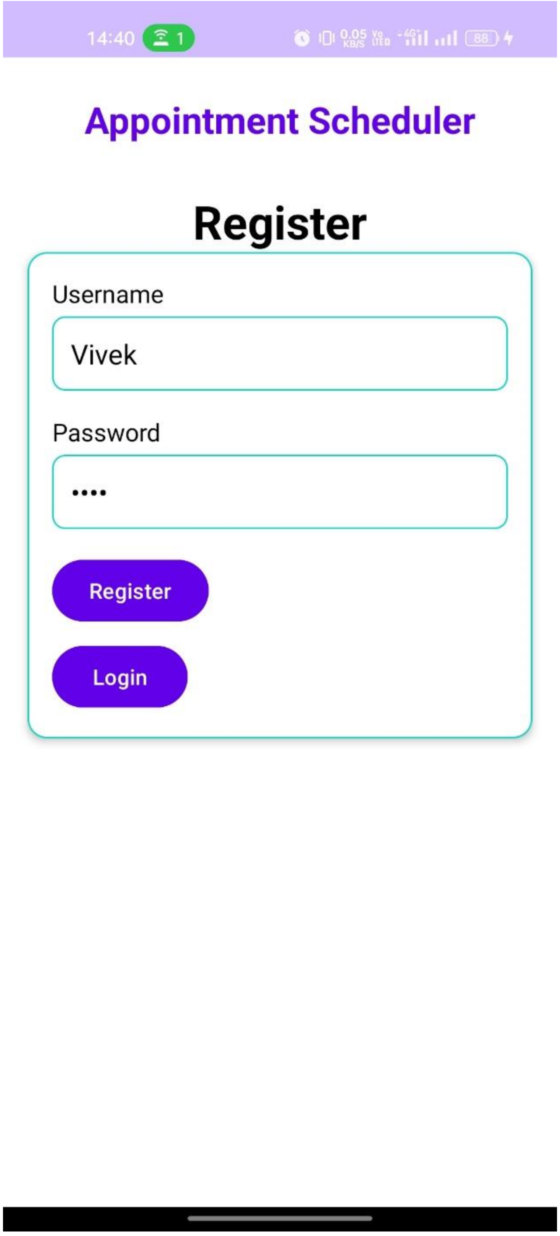
```
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Base.Theme.MAD_Project" parent="Theme.Material3.DayNight.NoActionBar">
        <!-- Customize your light theme here. -->
        <!-- <item name="colorPrimary">@color/my_light_primary</item> -->
    </style>
    <style name="Theme.MAD_Project" parent="Base.Theme.MAD_Project" />
</resources>
```

themes.xml(dark):

```
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Base.Theme.MAD_Project" parent="Theme.Material3.DayNight.NoActionBar">
        <!-- Customize your dark theme here. -->
        <!-- <item name="colorPrimary">@color/my_dark_primary</item> -->
    </style>
</resources>
```

Output:

Register:


14:40

Appointment Scheduler

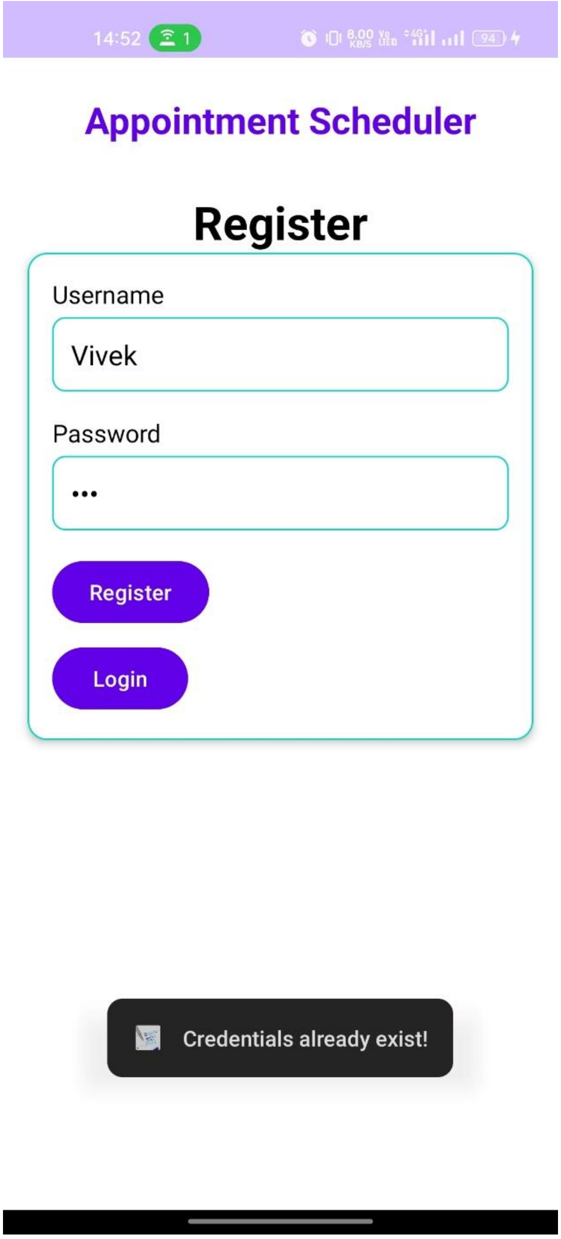
Register

Username

Vivek

Password

••••


14:52

Appointment Scheduler

Register

Username

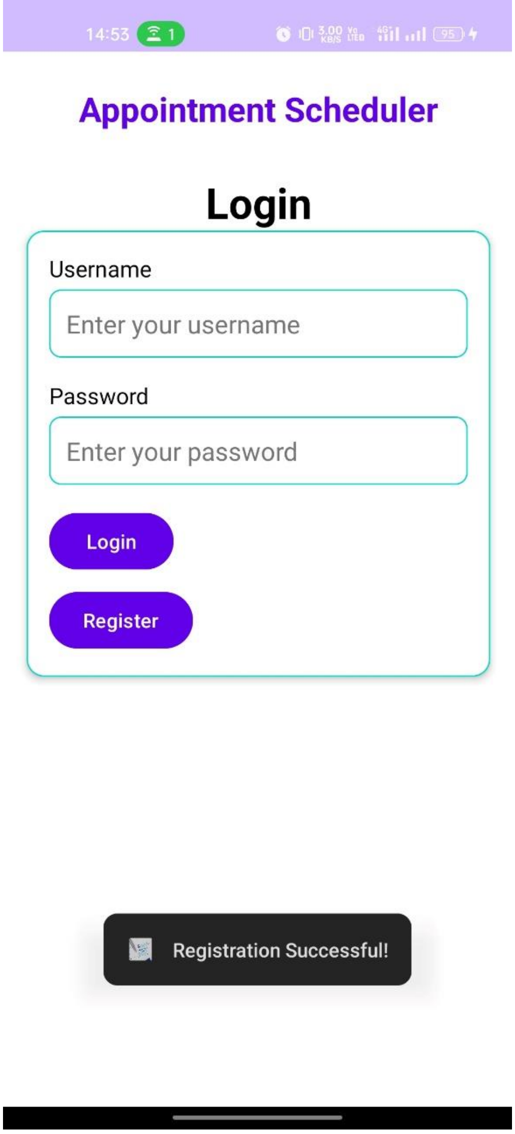
Vivek

Password

•••

Credentials already exist!

Login:


14:53

Appointment Scheduler

Login

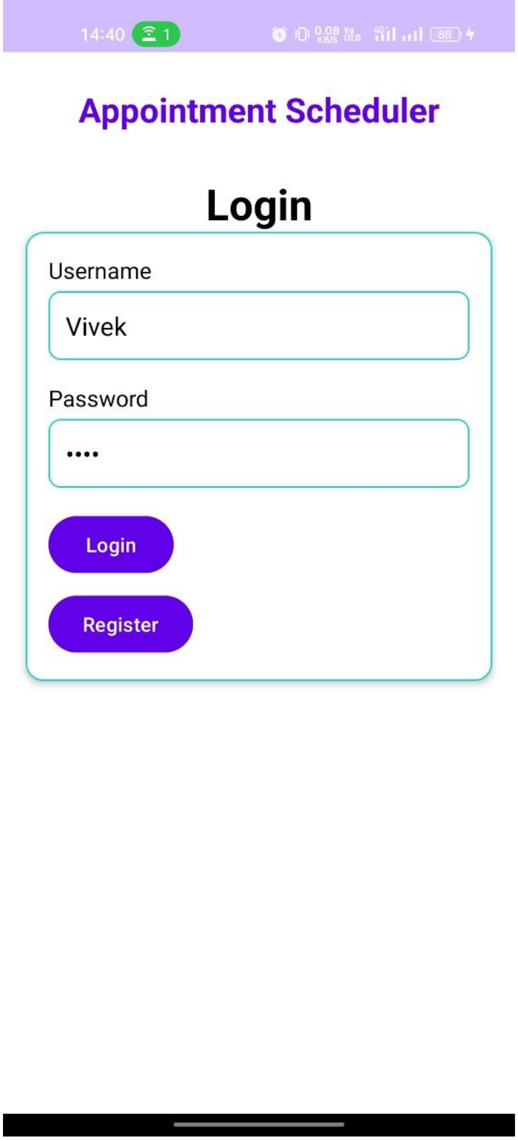
Username

Enter your username

Password

Enter your password

Registration Successful!


14:40

Appointment Scheduler

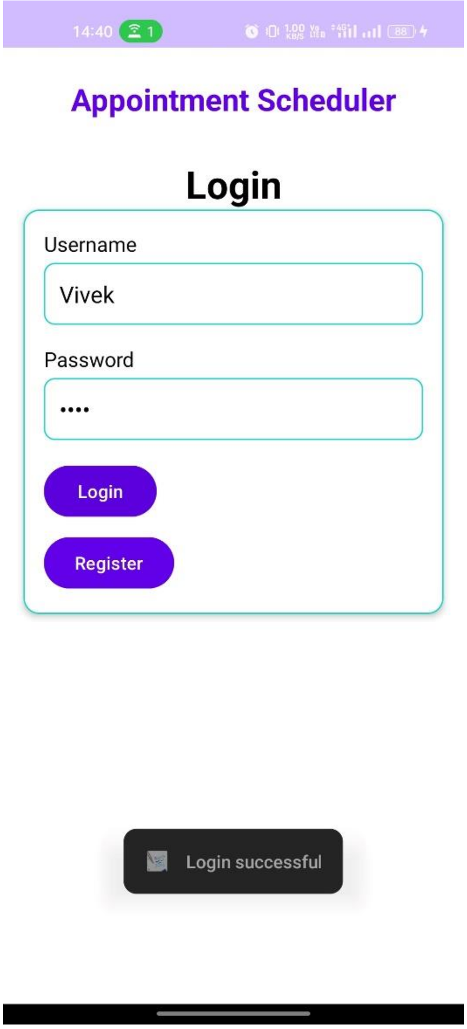
Login

Username

Vivek

Password

••••



Appointment Scheduler

Login

Username

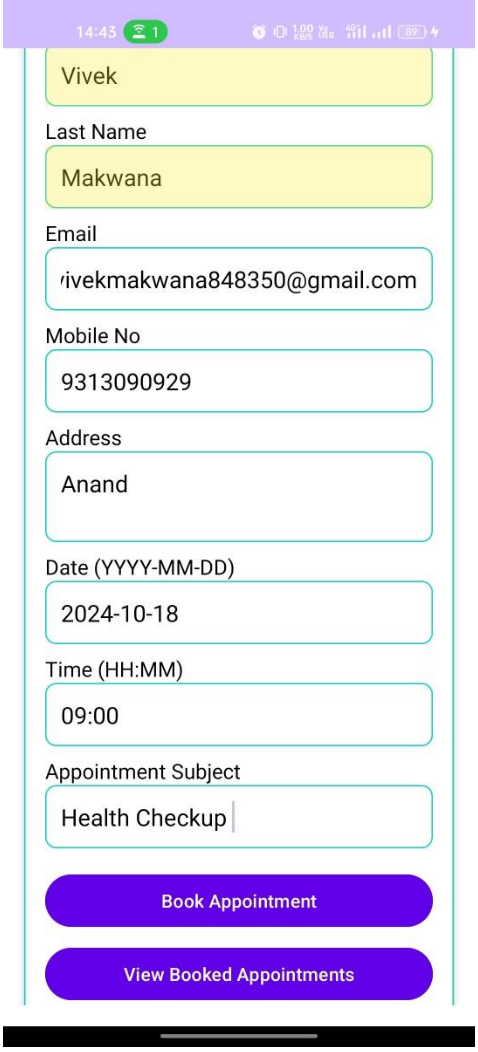
Vivek

Password

••••

Login successful

Appointment:



Vivek

Last Name

Makwana

Email

ivekmakwana848350@gmail.com

Mobile No

9313090929

Address

Anand

Date (YYYY-MM-DD)

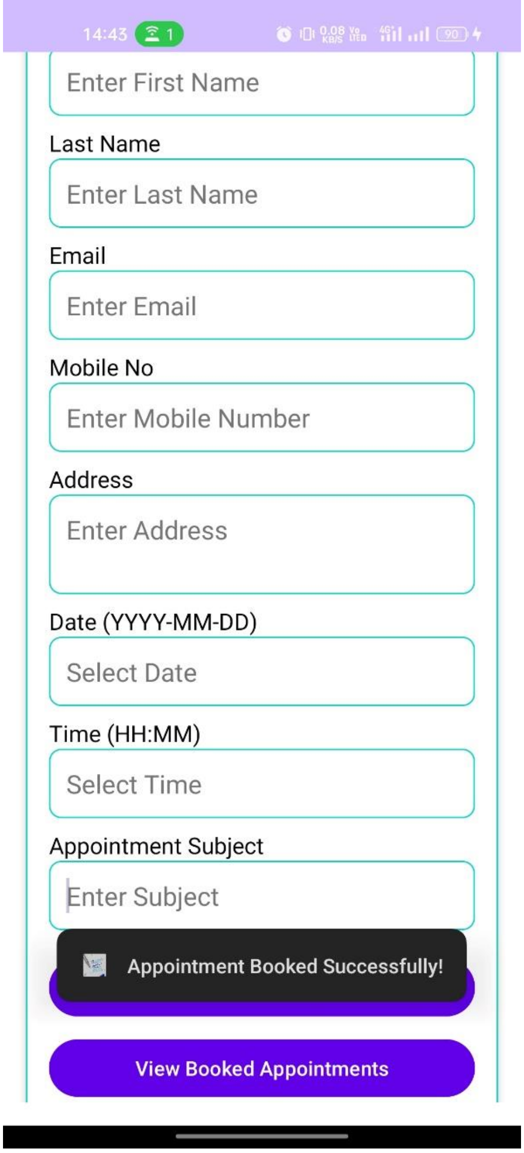
2024-10-18

Time (HH:MM)

09:00

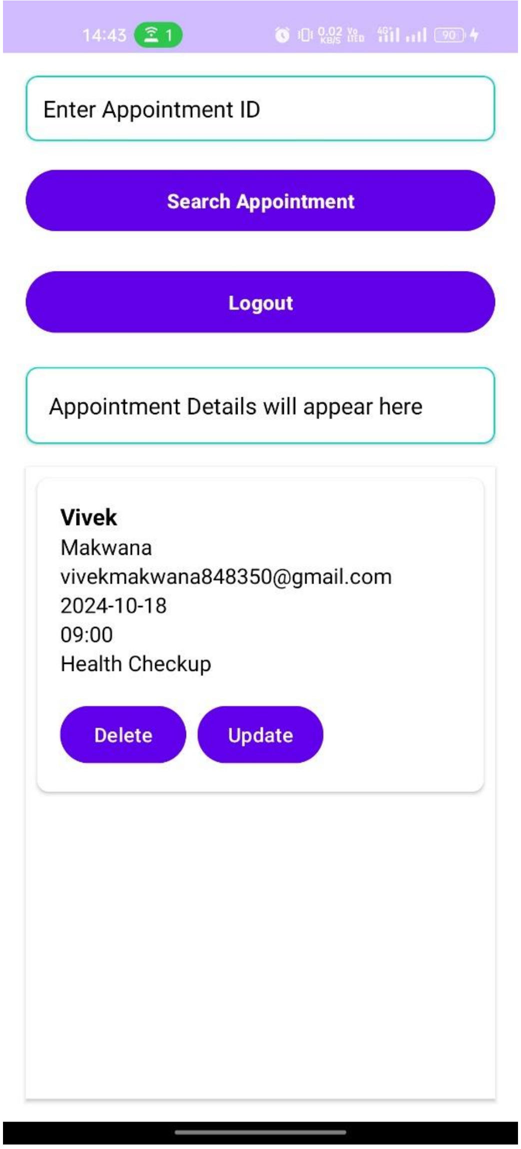
Appointment Subject

Health Checkup



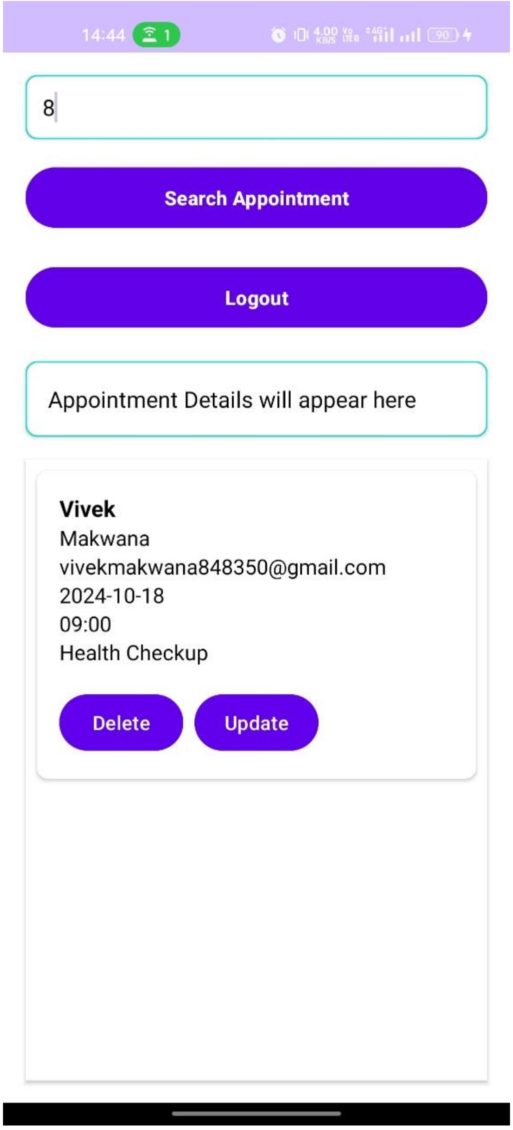
Appointment Booked Successfully!

Display Appointment:



Vivek
Makwana
vivekmakwana848350@gmail.com
2024-10-18
09:00
Health Checkup

Search By ID:



8

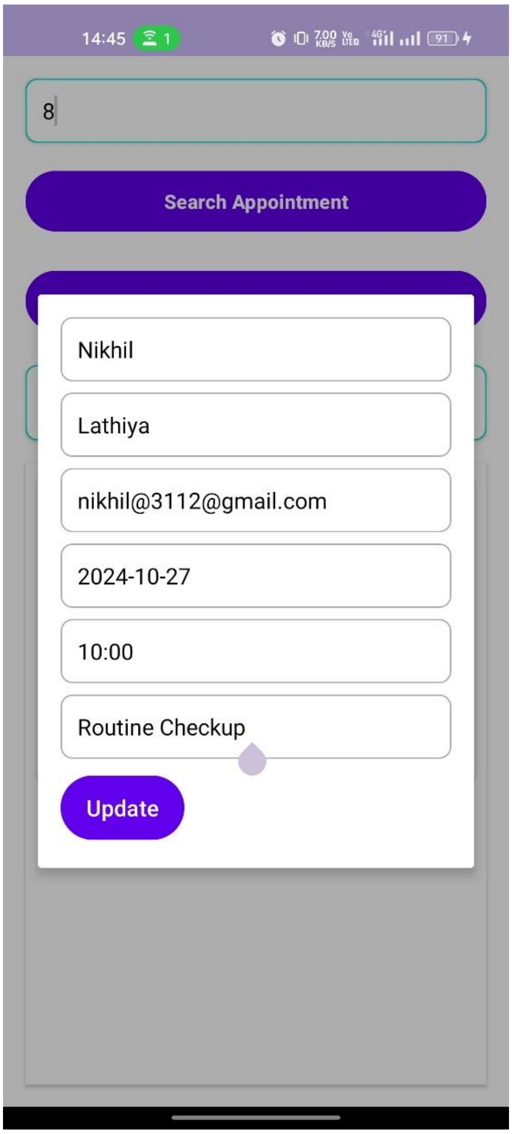
Search Appointment

Logout

Appointment Details will appear here

Vivek
Makwana
vivekmakwana848350@gmail.com
2024-10-18
09:00
Health Checkup

Update:



8

Search Appointment

Nikhil

Lathiya

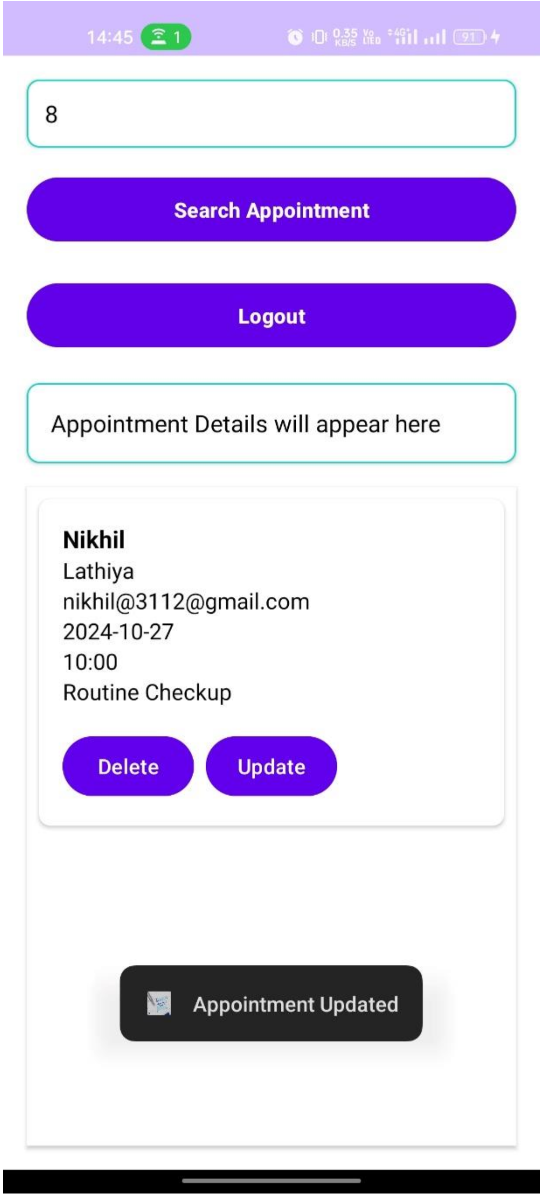
nikhil@3112@gmail.com

2024-10-27

10:00

Routine Checkup

Updated Data:



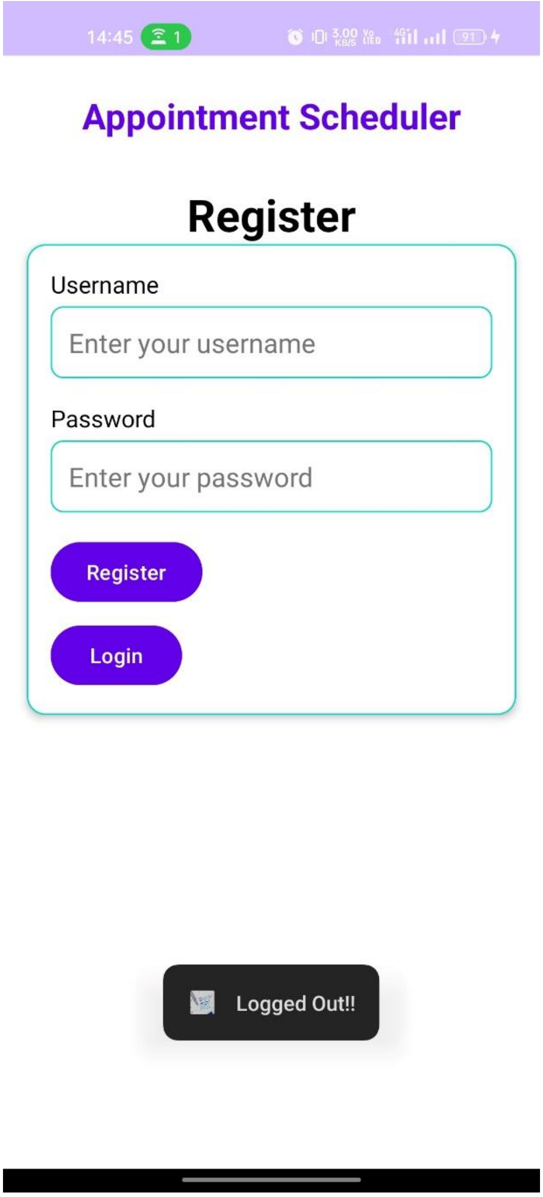
8

Appointment Details will appear here

Nikhil
Lathiya
nikhil@3112@gmail.com
2024-10-27
10:00
Routine Checkup

Appointment Updated

Logout:



Appointment Scheduler

Register

Username

Enter your username

Password

Enter your password

Logged Out!!