

SENT EMAIL

AIM & ALGORITHM

AIM :

Design a android program to create a Sending E-mail and SMS using mobApp.

ALGORITHM:

Step 1: Create a new project in Android Studio. Go to File -> New Project and fill all required details to create a new project.

Step 2: Design a user interface that allows users to select the type of message they want to send (email or SMS)

Step 3: Implement an authentication mechanism to verify users before allowing them to send messages.

Step 4: Add the code to res/layout/activity_main.xml.

Step 5: Allow users to compose the message with necessary formatting options such as text formatting, attachments, and multimedia content.

Step 6: Add the Code to src/MainActivity.java

Step 7: Implement the functionality to send the composed message via email or SMS, depending on the user's selection.

Step 8: Add the code to androidManifest.xml

ACTIVITY_MAIN.XML

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

```
android:layout_height="match_parent"
android:orientation="vertical"
android:padding="16dp">

<!-- Email Section -->
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Send Email"
    android:textSize="20sp"
    android:textStyle="bold" />

<EditText
    android:id="@+id/emailTo"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Recipient Email"
    android:inputType="textEmailAddress" />

<EditText
    android:id="@+id/emailSubject"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Subject" />

<EditText
    android:id="@+id/emailBody"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Email Body"
    android:inputType="textMultiLine"
    android:minLines="3" />

<Button
    android:id="@+id/sendEmailButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Send Email" />

<!-- Messaging Section -->
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:text="Send Message"
    android:textSize="20sp"
    android:textStyle="bold" />

<EditText
```

```

        android:id="@+id/messagePhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Recipient Phone Number"
        android:inputType="phone" />

<EditText
    android:id="@+id/messageBody"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Message"
    android:inputType="textMultiLine"
    android:minLines="2" />

<Button
    android:id="@+id/sendMessageButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Send Message" />

</LinearLayout>

```

MAINACTIVITY.JAVA

```

package com.example.studentapp;

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

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import android.content.pm.PackageManager;
import androidx.core.content.ContextCompat;

public class MainActivity extends AppCompatActivity {

    private EditText emailTo, emailSubject, emailBody, messagePhone,
messageBody;
    private static final int SMS_PERMISSION_CODE = 100;

    @Override
    public void onCreate(Bundle savedInstanceState) {

```

```

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

// Initialize UI components
emailTo = findViewById(R.id.emailTo);
emailSubject = findViewById(R.id.emailSubject);
emailBody = findViewById(R.id.emailBody);
Button sendEmailButton = findViewById(R.id.sendEmailButton);
messagePhone = findViewById(R.id.messagePhone);
messageBody = findViewById(R.id.messageBody);
Button sendMessageButton = findViewById(R.id.sendMessageButton);

// Email button click listener
sendEmailButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String to = emailTo.getText().toString();
        String subject = emailSubject.getText().toString();
        String body = emailBody.getText().toString();

        if (to.isEmpty() || subject.isEmpty() || body.isEmpty()) {
            Toast.makeText(MainActivity.this, "Please fill all email
fields", Toast.LENGTH_SHORT).show();
            return;
        }

        Intent emailIntent = new Intent(Intent.ACTION_SEND);
        emailIntent.setType("message/rfc822");
        emailIntent.putExtra(Intent.EXTRA_EMAIL, new String[]{to});
        emailIntent.putExtra(Intent.EXTRA_SUBJECT, subject);
        emailIntent.putExtra(Intent.EXTRA_TEXT, body);

        try {
            startActivity(Intent.createChooser(emailIntent, "Send
email..."));
        } catch (Exception e) {
            Toast.makeText(MainActivity.this, "No email app found",
Toast.LENGTH_SHORT).show();
        }
    }
});

// Message button click listener
sendMessageButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String phone = messagePhone.getText().toString();
        String message = messageBody.getText().toString();

```

```

        if (phone.isEmpty() || message.isEmpty()) {
            Toast.makeText(MainActivity.this, "Please fill all message
fields", Toast.LENGTH_SHORT).show();
            return;
        }

        if (ContextCompat.checkSelfPermission(MainActivity.this,
android.Manifest.permission.SEND_SMS)
            != PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(MainActivity.this,
                new String[]{android.Manifest.permission.SEND_SMS},
SMS_PERMISSION_CODE);
        } else {
            sendSMS(phone, message);
        }
    }
});
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions, @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions,
grantResults);
    if (requestCode == SMS_PERMISSION_CODE) {
        if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            String phone = messagePhone.getText().toString();
            String message = messageBody.getText().toString();
            sendSMS(phone, message);
        } else {
            Toast.makeText(this, "SMS permission denied",
Toast.LENGTH_SHORT).show();
        }
    }
}

private void sendSMS(String phone, String message) {
    try {
        SmsManager smsManager = SmsManager.getDefault();
        smsManager.sendTextMessage(phone, null, message, null, null);
        Toast.makeText(this, "Message sent", Toast.LENGTH_SHORT).show();
    } catch (Exception e) {
        Toast.makeText(this, "Failed to send message",
Toast.LENGTH_SHORT).show();
    }
}
}
}

```


ANDROIDMANIFEST.XML

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.AppCompat.Light.DarkActionBar">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

OUTPUT

1:20

studentapp

Send Email

Recipient Email

Subject

Email Body

SEND EMAIL

Send Message

Recipient Phone Number

Message

SEND MESSAGE