| **Name of Student:** Ajay Karthikesan | | | |
| --- | --- | --- | --- |
| **Roll Number:** 57 | | **Practical Number:** 3 | |
| **Aim of Practical:**  Android program based on intents. | | | |
| **DOP:** 9.10.23 | | **DOS:** 28.10.23 | |
| **CO Mapped:** - | **PO Mapped:** - | **Faculty Signature:** | **Marks:** |

## 

## Practical No. 3

**Aim:** Android program based on intents.

**Theory:**

In Android, it is quite usual for users to witness a jump from one application to another as a part of the whole process, for example, searching for a location on the browser and witnessing a direct jump into Google Maps or receiving payment links in Messages Application (SMS) and on clicking jumping to PayPal or GPay (Google Pay). This process of taking users from one application to another is achieved by passing the Intent to the system. Intents, in general, are used for navigating among various activities within the same application, but note, is not limited to one single application, i.e., they can be utilized from moving from one application to another as well.

Intents could be Implicit, for instance, calling intended actions, and explicit as well, such as opening another activity after some operations like onClick or anything else. Below are some applications of Intents:

Sending the User to Another App

Getting a Result from an Activity

Allowing Other Apps to Start Your Activity

The collaborative nature of Android applications only results in a better user experience. The question here is if the intent is for an application that is not present in the device, what’s the next call?

Some Important Method of Intent and their Description

Methods

Description

Context.startActivity() This is to launch a new activity or get an existing activity to be action.

Context.startService() This is to start a new service or deliver instructions for an existing service.

Context.sendBroadcast() This is to deliver the message to broadcast receivers.

Deep Linking

Deep Link is an URL that redirects the device to the API of that Missing Application and then the service is run on the system to check if a version of that application exists on the device. For time being, let’s assume that the application is not available on the device and no previous versions ever existed. This service then makes a call to the Play Store from the device and the application appears, just a matter of download.

Not deviating from the topic, there are a few examples that already exist in Android Studio for redirecting to other applications, for example, Dialing Numbers, Sending SMSs, Opening Settings, etc. Everyday examples include redirecting to YouTube, Maps, WhatsApp, Facebook, etc. Android community, especially the Kotlin Community is at its zenith every single impending day. Kotlin has witnessed a huge amount of sudden growth in the past few years and could be one of the vital tools in the future replacing Java & possibly Julia also.

Types of Android Intents

There are two types of intents in android

Implicit

Explicit

Implicit Intent

Implicit Intent doesn’t specify the component. In such a case, intent provides information on available components provided by the system that is to be invoked. For example, you may write the following code to view the webpage.

Explicit Intent specifies the component. In such a case, intent provides the external class to be invoked.

**Code:**

File: MainActivity.java

package vesit.ajayk57.mc.practical3;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

findViewById(R.id.goToWebSiteBtn).setOnClickListener(view -> goToWebsite());

findViewById(R.id.openMapBtn).setOnClickListener(view -> openMap());

findViewById(R.id.shareTextBtn).setOnClickListener(view -> shareText());

}

private void goToWebsite(){

final EditText websiteTxt =findViewById(R.id.websiteTxt);

String url=websiteTxt.getText().toString();

if (!url.startsWith("http://") && !url.startsWith("https://"))

{

url = "http://" + url;

}

final Intent browserIntent = new Intent(Intent.ACTION\_VIEW, Uri.parse(url));

startActivity(browserIntent);

}

private void openMap(){

final EditText locationTxt=findViewById(R.id.locationTxt);

final String location = "http://maps.google.co.in/maps?q=" + locationTxt.getText().toString();

final Intent intent = new Intent(Intent.ACTION\_VIEW, Uri.parse(location));

startActivity(intent);

}

private void shareText(){

final Intent intent = new Intent(android.content.Intent.ACTION\_SEND);

final EditText messageTxt=findViewById(R.id.messageTxt);

final String message = messageTxt.getText().toString();

intent.setType("text/plain");

intent.putExtra(android.content.Intent.EXTRA\_TEXT, message);

startActivity(Intent.createChooser(intent, getString(R.string.share\_using)));

}

}

File: activity\_main.java

<?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:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<com.google.android.material.textfield.TextInputLayout

android:id="@+id/textInputLayout"

android:layout\_width="350dp"

android:layout\_height="wrap\_content"

android:layout\_marginTop="16dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.5"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.0">

<com.google.android.material.textfield.TextInputEditText

android:id="@+id/websiteTxt"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="@string/website\_url"

android:textSize="20sp" />

</com.google.android.material.textfield.TextInputLayout>

<Button

android:id="@+id/goToWebSiteBtn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp"

android:text="@string/go\_to\_webiste"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.5"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textInputLayout"

app:layout\_constraintVertical\_bias="0.0" />

<com.google.android.material.textfield.TextInputLayout

android:id="@+id/textInputLayout2"

android:layout\_width="350dp"

android:layout\_height="wrap\_content"

android:layout\_marginTop="16dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.5"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/goToWebSiteBtn"

app:layout\_constraintVertical\_bias="0.0">

<com.google.android.material.textfield.TextInputEditText

android:id="@+id/locationTxt"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="@string/location"

android:textSize="20sp" />

</com.google.android.material.textfield.TextInputLayout>

<Button

android:id="@+id/openMapBtn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp"

android:text="@string/open\_map"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.5"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/textInputLayout2"

app:layout\_constraintVertical\_bias="0.0" />

<com.google.android.material.textfield.TextInputLayout

android:id="@+id/textInputLayout3"

android:layout\_width="350dp"

android:layout\_height="wrap\_content"

android:layout\_marginTop="16dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.5"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/openMapBtn"

app:layout\_constraintVertical\_bias="0.0">

<com.google.android.material.textfield.TextInputEditText

android:id="@+id/messageTxt"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="@string/message"

android:textSize="20sp" />

</com.google.android.material.textfield.TextInputLayout>

<Button

android:id="@+id/shareTextBtn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp"

android:text="@string/send\_message"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.5"

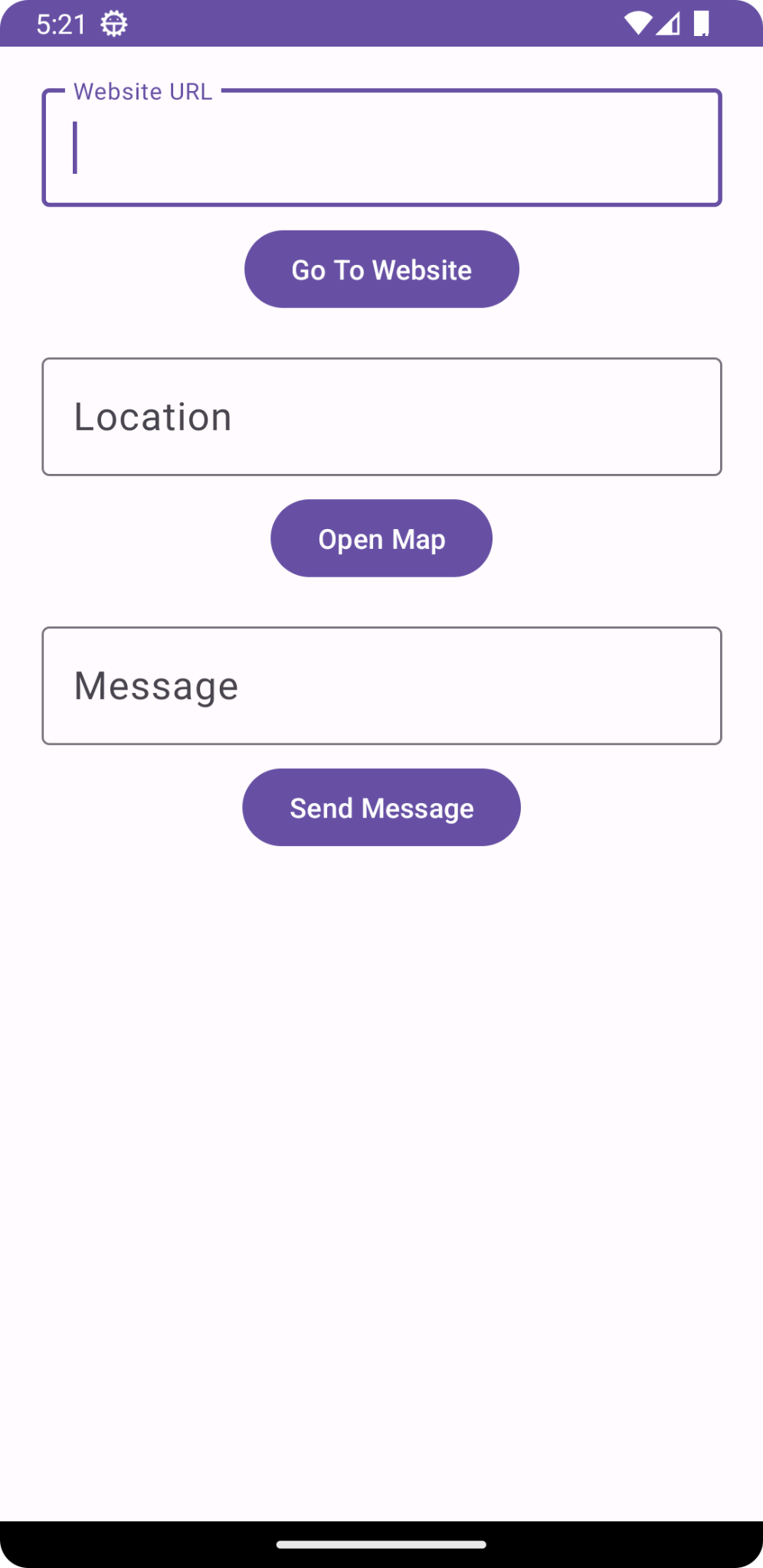
app:layout\_constraintStart\_toStartOf="parent"

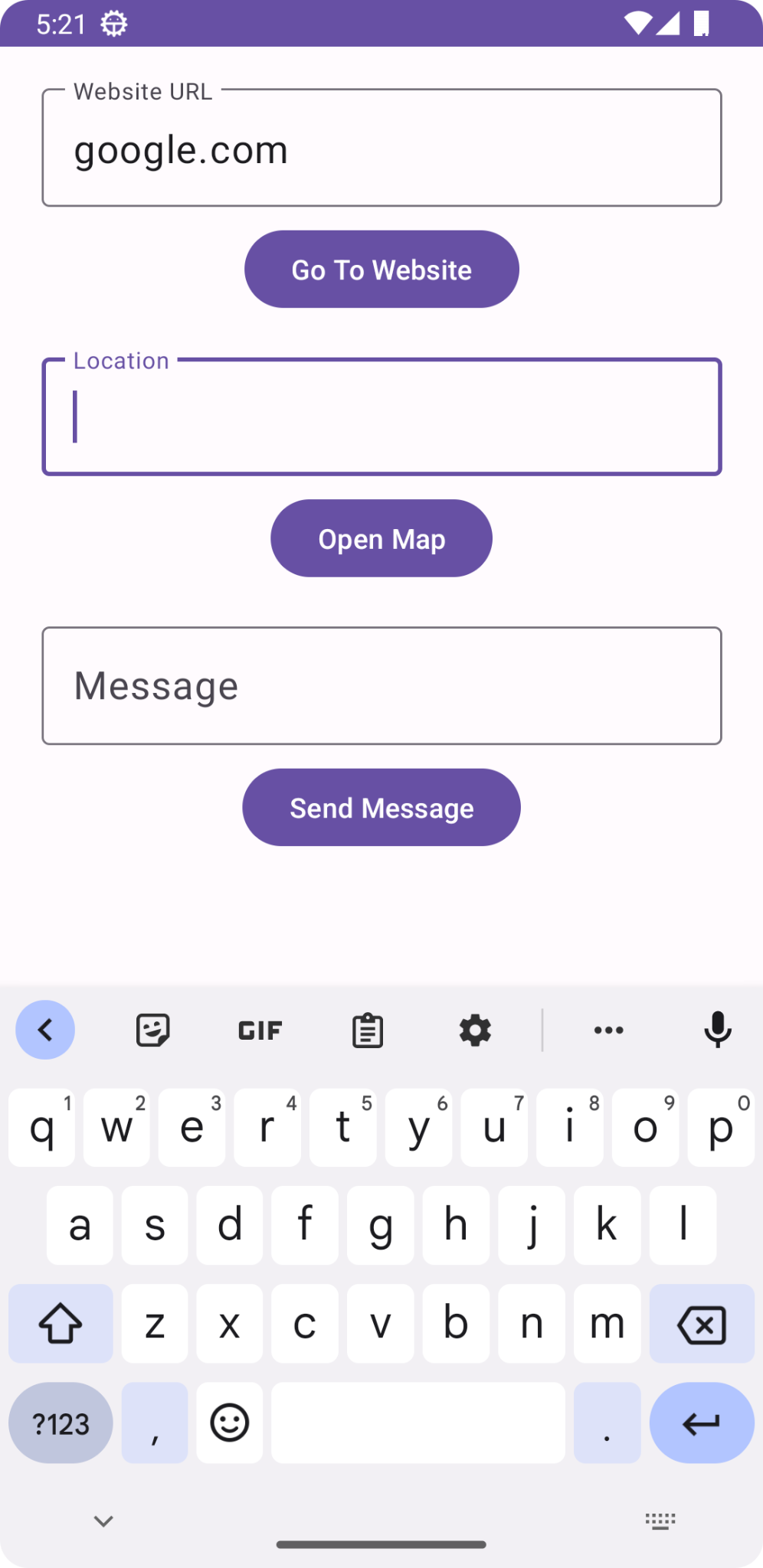
app:layout\_constraintTop\_toBottomOf="@+id/textInputLayout3"

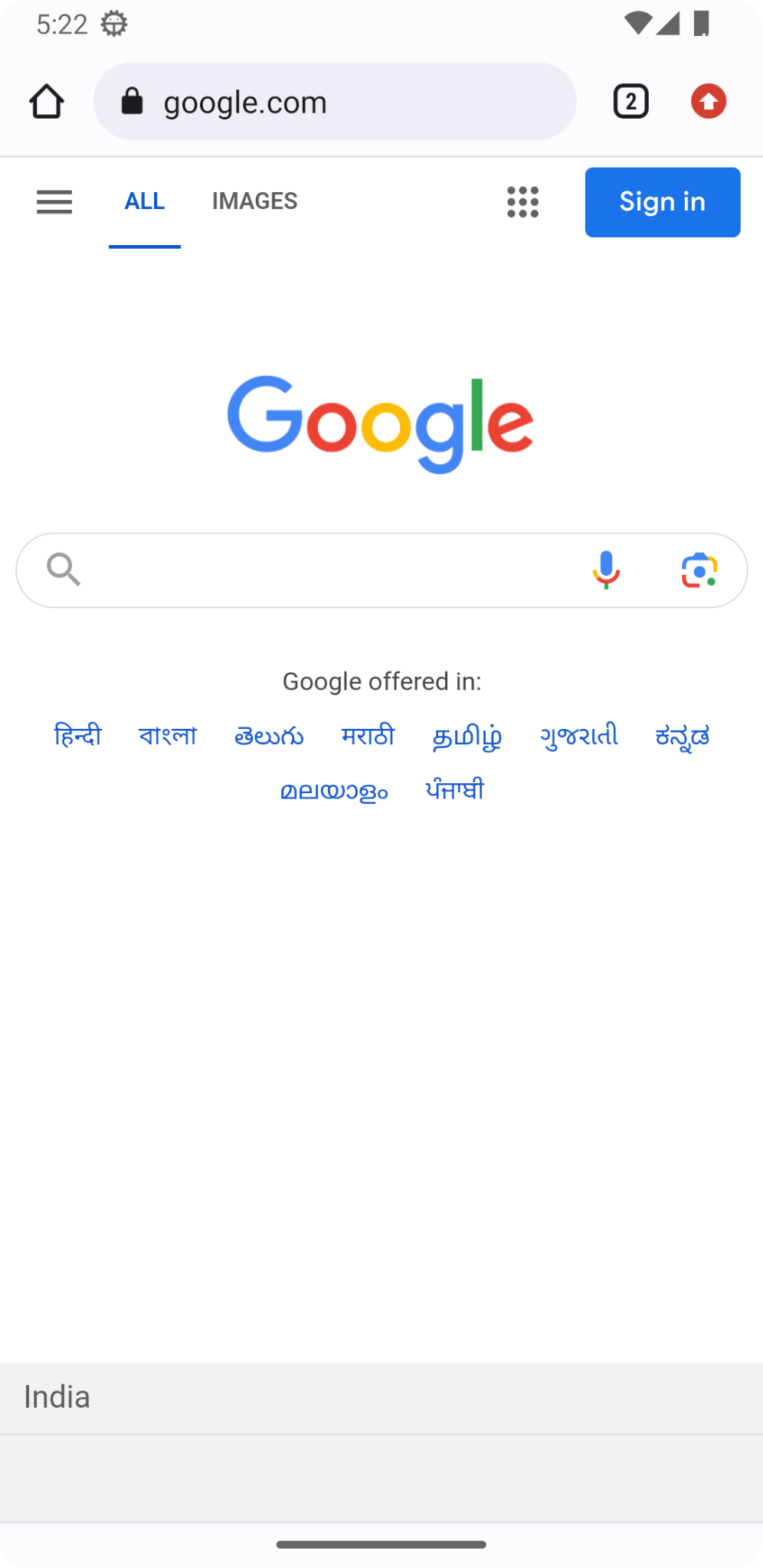
app:layout\_constraintVertical\_bias="0.0" />

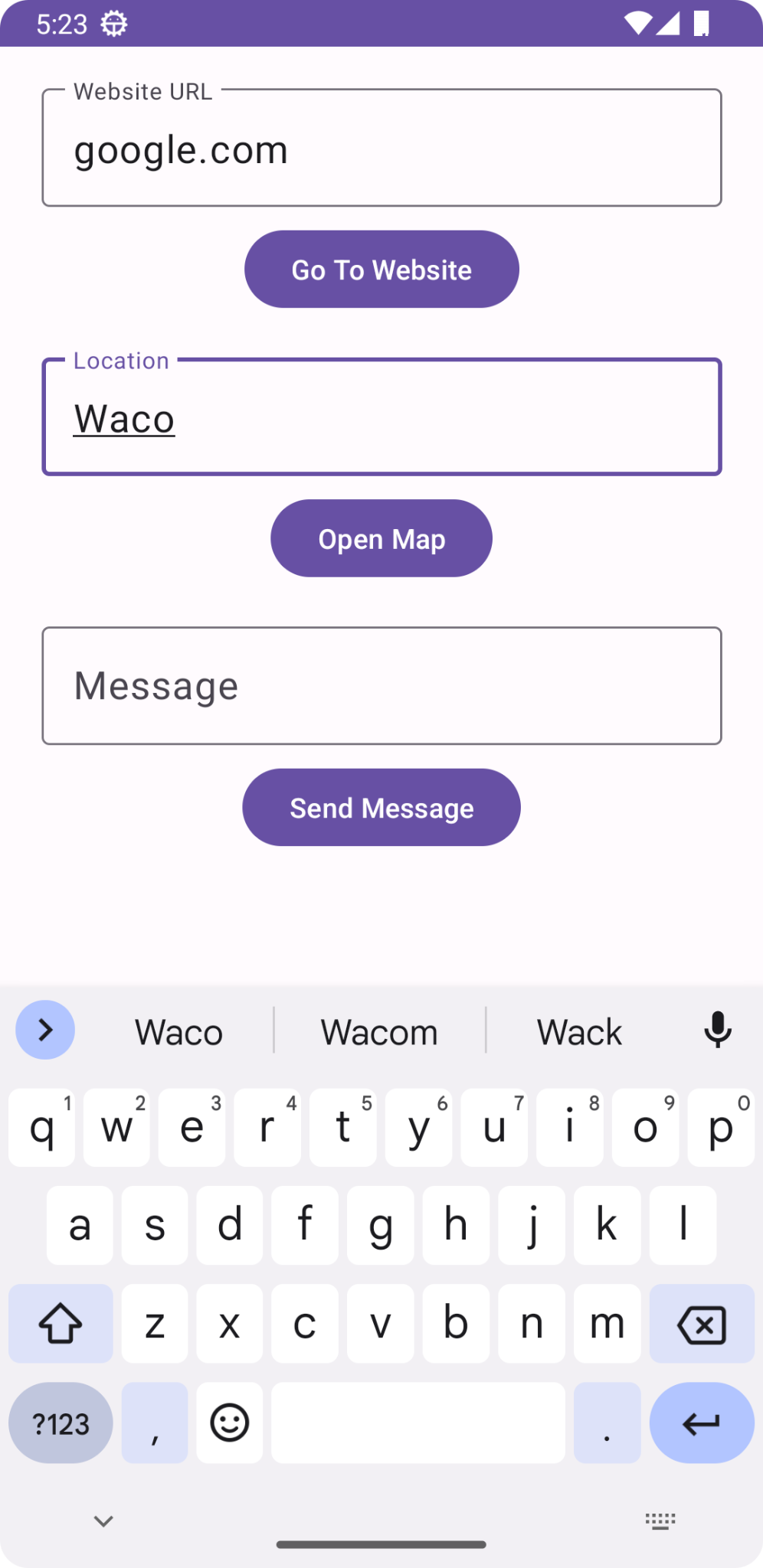
</androidx.constraintlayout.widget.ConstraintLayout>

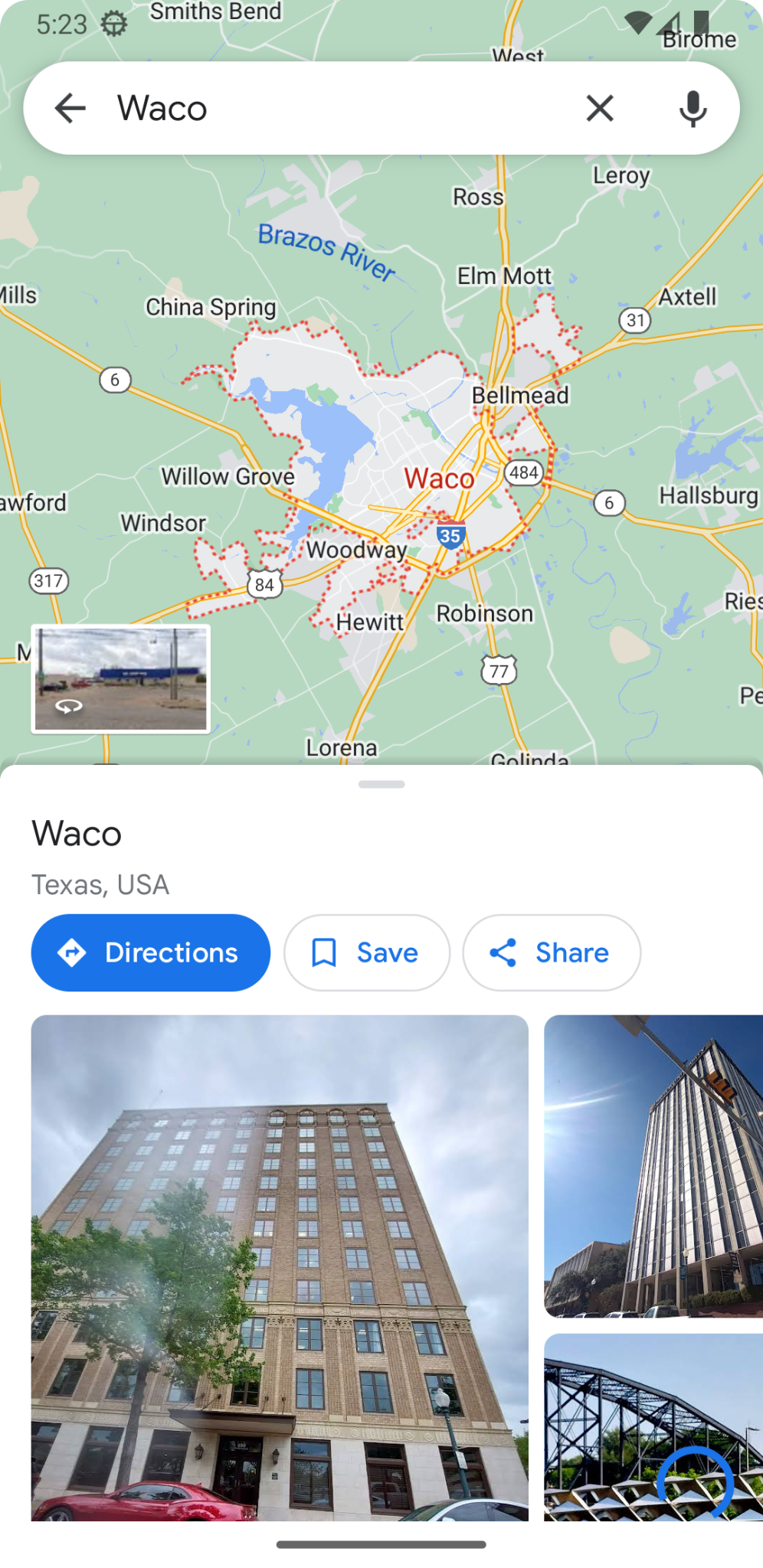
**Output:**

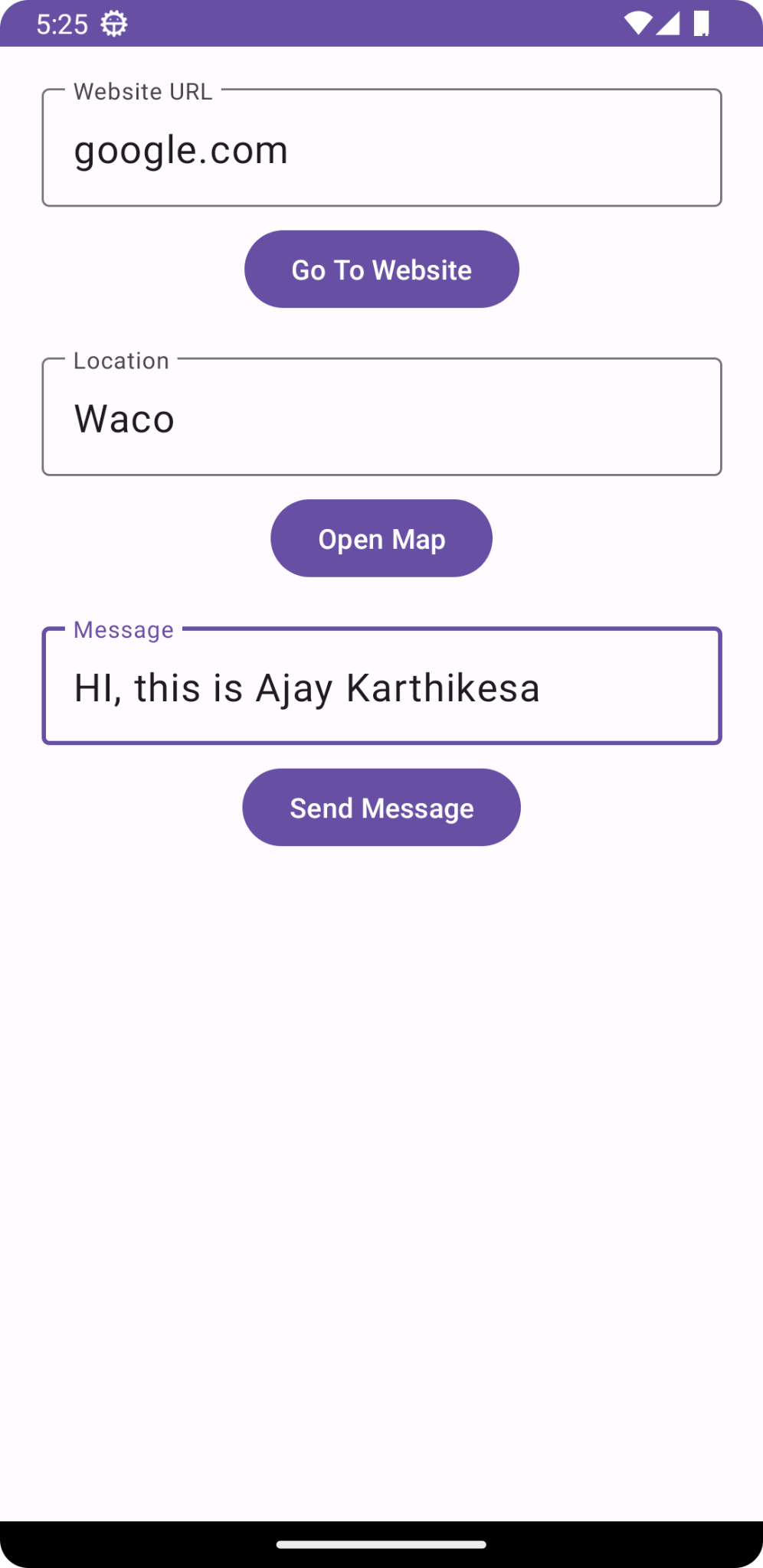


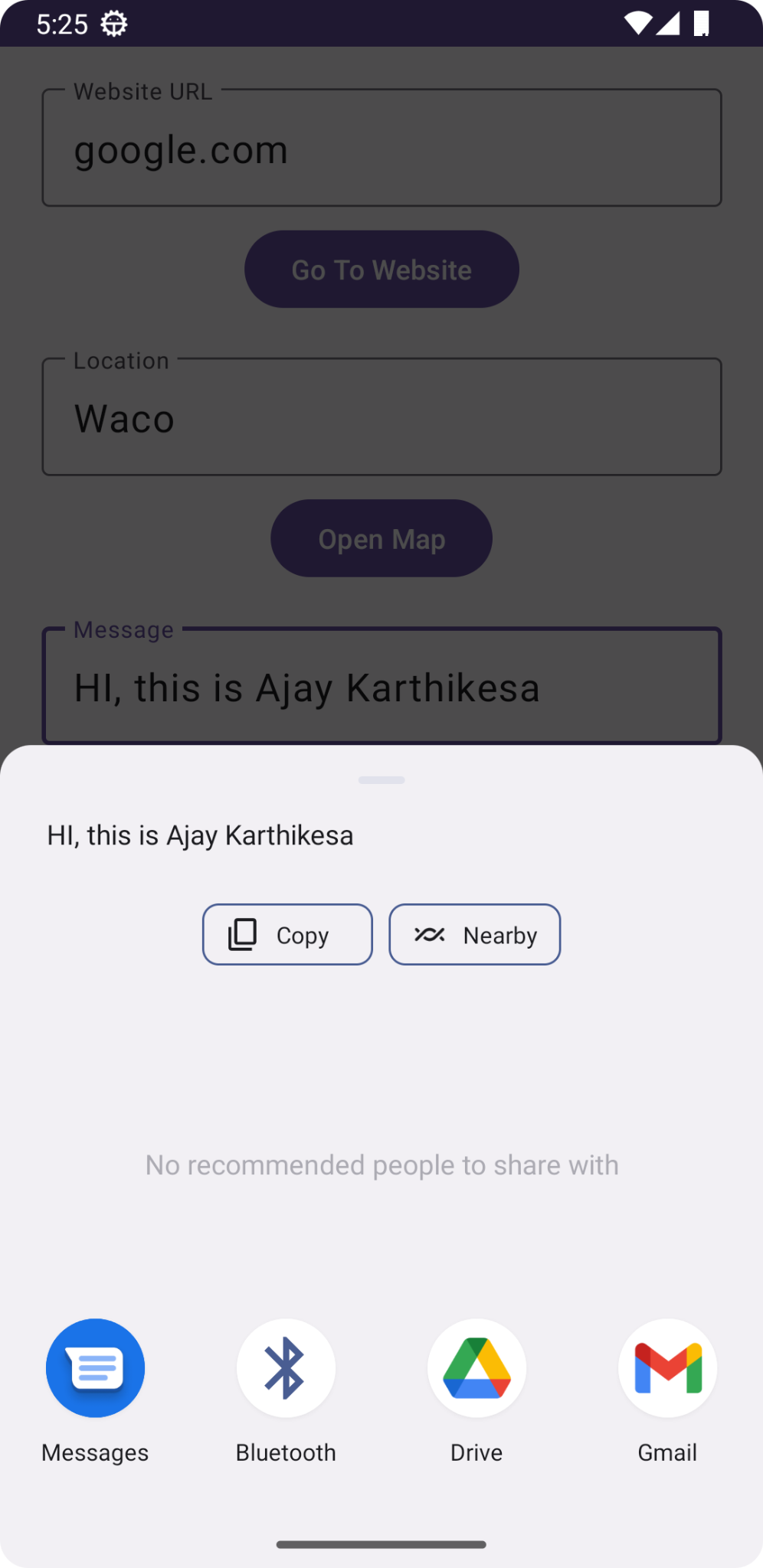












**Conclusion:**

I learnt about intents in android.