|  |  |  |  |
| --- | --- | --- | --- |
| **Roll No: 113** | **Name:Harshita Shetty** | **Div: B** | **Batch: B2** |

**Experiment No.08: Location Manager (GPS)**

**Aim:** Develop a native application that uses GPS location information.

**Theory:** A Location Manager App is a software application that helps users manage and track their location-based activities, such as navigation, travel planning, geocaching, and more. The app uses location data from the user's device (such as GPS or Wi-Fi signals) to provide real-time information about their location and surroundings.

The core functionality of a Location Manager App typically includes the following features:

1. Maps and Navigation: The app provides users with access to maps and navigation tools, allowing them to plan and navigate routes to their desired destinations.
2. Location Tracking: The app tracks the user's location in real-time, providing them with information about their current location and nearby points of interest.
3. Geocaching: The app allows users to search for and find hidden geocaches or other hidden items based on GPS coordinates.
4. Travel Planning: The app can help users plan trips and activities by providing information about nearby attractions, restaurants, and accommodations.
5. Safety and Emergency Services: The app can provide users with information about emergency services and safety tips based on their location.

Other features that may be included in a Location Manager App include social sharing and collaboration tools, weather forecasts, and personalized recommendations based on the user's location and preferences.

Overall, a Location Manager App is designed to provide users with a comprehensive set of tools and features for managing and tracking their location-based activities, with the goal of making their travel and exploration experiences more enjoyable, efficient, and safe.

**Program:**

# MainActivity.java

package com.yayandroid.locationmanager.sample;

import android.content.Intent; import android.os.Bundle; import android.view.View;

import androidx.annotation.Nullable;

import androidx.appcompat.app.AppCompatActivity;

import com.yayandroid.locationmanager.sample.activity.SampleActivity; import com.yayandroid.locationmanager.sample.fragment.SampleFragmentActivity; import com.yayandroid.locationmanager.sample.service.SampleServiceActivity;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(@Nullable Bundle savedInstanceState) {

super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

}

public void inActivityClick(View view) {

startActivity(new Intent(this, SampleActivity.class));

}

public void inFragmentClick(View view) {

startActivity(new Intent(this, SampleFragmentActivity.class));

}

public void inServiceClick(View view) {

startActivity(new Intent(this, SampleServiceActivity.class));

}

}

# SamplePresenter.java

package com.yayandroid.locationmanager.sample;

import android.location.Location;

import android.text.TextUtils;

import com.yayandroid.locationmanager.constants.FailType; import com.yayandroid.locationmanager.constants.ProcessType;

public class SamplePresenter {

private SampleView sampleView;

public SamplePresenter(SampleView view) {

this.sampleView = view;

}

public void destroy() {

sampleView = null;

}

public void onLocationChanged(Location location) { sampleView.dismissProgress();

setText(location);

}

public void onLocationFailed(@FailType int failType) { sampleView.dismissProgress();

switch (failType) { case FailType.TIMEOUT: {

sampleView.setText("Couldn't get location, and timeout!"); break;

}

case FailType.PERMISSION\_DENIED: {

sampleView.setText("Couldn't get location, because user didn't give permission!"); break;

}

case FailType.NETWORK\_NOT\_AVAILABLE: {

sampleView.setText("Couldn't get location, because network is not accessible!"); break;

}

case FailType.GOOGLE\_PLAY\_SERVICES\_NOT\_AVAILABLE: {

sampleView.setText("Couldn't get location, because Google Play Services not available!");

break;

}

case FailType.GOOGLE\_PLAY\_SERVICES\_SETTINGS\_DIALOG: {

sampleView.setText("Couldn't display settingsApi dialog!"); break;

}

case FailType.GOOGLE\_PLAY\_SERVICES\_SETTINGS\_DENIED: {

sampleView.setText("Couldn't get location, because user didn't activate providers via settingsApi!");

break;

}

case FailType.VIEW\_DETACHED: {

sampleView.setText("Couldn't get location, because in the process view was detached!");

break;

}

case FailType.VIEW\_NOT\_REQUIRED\_TYPE: { sampleView.setText("Couldn't get location, "

+ "because view wasn't sufficient enough to fulfill given configuration!"); break;

}

case FailType.UNKNOWN: {

sampleView.setText("Ops! Something went wrong!"); break;

}

}

}

public void onProcessTypeChanged(@ProcessType int newProcess) { switch (newProcess) {

case ProcessType.GETTING\_LOCATION\_FROM\_GOOGLE\_PLAY\_SERVICES: {

sampleView.updateProgress("Getting Location from Google Play Services..."); break;

}

case ProcessType.GETTING\_LOCATION\_FROM\_GPS\_PROVIDER: { sampleView.updateProgress("Getting Location from GPS..."); break;

}

case ProcessType.GETTING\_LOCATION\_FROM\_NETWORK\_PROVIDER: {

sampleView.updateProgress("Getting Location from Network..."); break;

}

case ProcessType.ASKING\_PERMISSIONS: case ProcessType.GETTING\_LOCATION\_FROM\_CUSTOM\_PROVIDER:

// Ignored

break;

}

}

private void setText(Location location) {

String appendValue = location.getLatitude() + ", " + location.getLongitude() + "\n"; String newValue;

CharSequence current = sampleView.getText();

if (!TextUtils.isEmpty(current)) { newValue = current + appendValue;

} else {

newValue = appendValue;

}

sampleView.setText(newValue);

}

public interface SampleView {

String getText();

void setText(String text);

void updateProgress(String text);

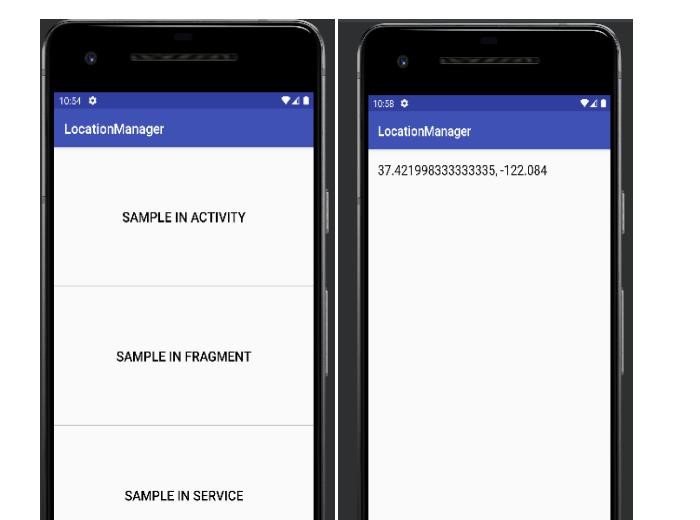
void dismissProgress();

}

}

# GitHub Link: https://github.com/shettyharshita/Mobile-Computing

**Output:**



**Conclusion:** The experiment was about Location Manager and the use of Location to connect the app which is successfully implemented and verified.