**程序设计项目课程设计任务书**

**20202231014关竣佑**

**[一，系统需求分析 1](#_Toc14754)**

**[二，总体设计 1](#_Toc30339)**

**[三，界面各功能模块的具体实现 1](#_Toc2191)**

**[四，系统调试 3](#_Toc2191)**

**[五，使用说明及编程体会 3](#_Toc2191)**

**[六，附：源程序 4](#_Toc2191)**

**一，系统需求分析**

众所周知，获取到准确的天气状况是很多人出门或者规划当天的行程的重要依据，因此及时方便地获取天气情况成为了很多人的刚需。 本项目为天气预报安卓APP，目的在于在移动端为用户显示用户所在城市及其它城市三天的温度天气，空气质量等天气信息。便于用户快捷方便地获取到最新的天气资讯。本项目旨在为广大用户提供一个准确可靠的免费天气服务，为广大用户实现“手机在手，天气我知”。

本天气预报APP 具有切换城市显示天气，空气质量状况的功能，并自动获取识别用户所在的城市，为用户提供所在城市的天气预报。用户切换到菜单页，选择城市后跳转到该城市的天气预报，用户亦可以在菜单页等城市识别成功后查看所在城市的天气情况。

**二，总体设计**

本项目是一个无后端的简易天气预报APP，安装环境为安卓系统或鸿蒙系统。本APP设计有三个页面，分别为欢迎页面Home , 菜单页面Menu 以及 天气情况展示页面 weather 。在菜单页面，一共有三个城市选择按钮分别为北京广州和佛山。当用户选择其中一个城市时顶部显示当前选择的城市并跳转到该城市天气预报页面。菜单页面底部有一个面板显示识别用户所处的当前城市，识别成功后以后点击此按钮可跳转到该城市天气。在天气页面，顶部展示今天白天/夜晚的天气，气温。下面展示未来两天的天气和 温度信息。

本项目采用Java+XML+Android SDK等进行开发，使用IDEA作为开发平台。项目大致遵循MVC的设计模式。分为逻辑控制模块，样式模块和数据层模块。由于本项目不同于一般的前后端交互web项目，无需数据持久层，因此数据层模块被弱化。而本项目需要大量请求外界API ，故每个页面从逻辑控制层中剥离出单独的请求类，用于请求接口。在样式层中使用XML来控制每个页面的按钮，背景，图标等样式。在逻辑控制模块使用Java代码控制样式层的样式，文字的改变。

由于本APP一共有三页，而且并列，故使用IDEA 的Bottom Navigation Activity Android模板来进行创建。

**三，界面各功能模块的具体实现**

在菜单页面中有以下的关键类：Menu ，getLocationCity ，MyLocationListener ， OnClick

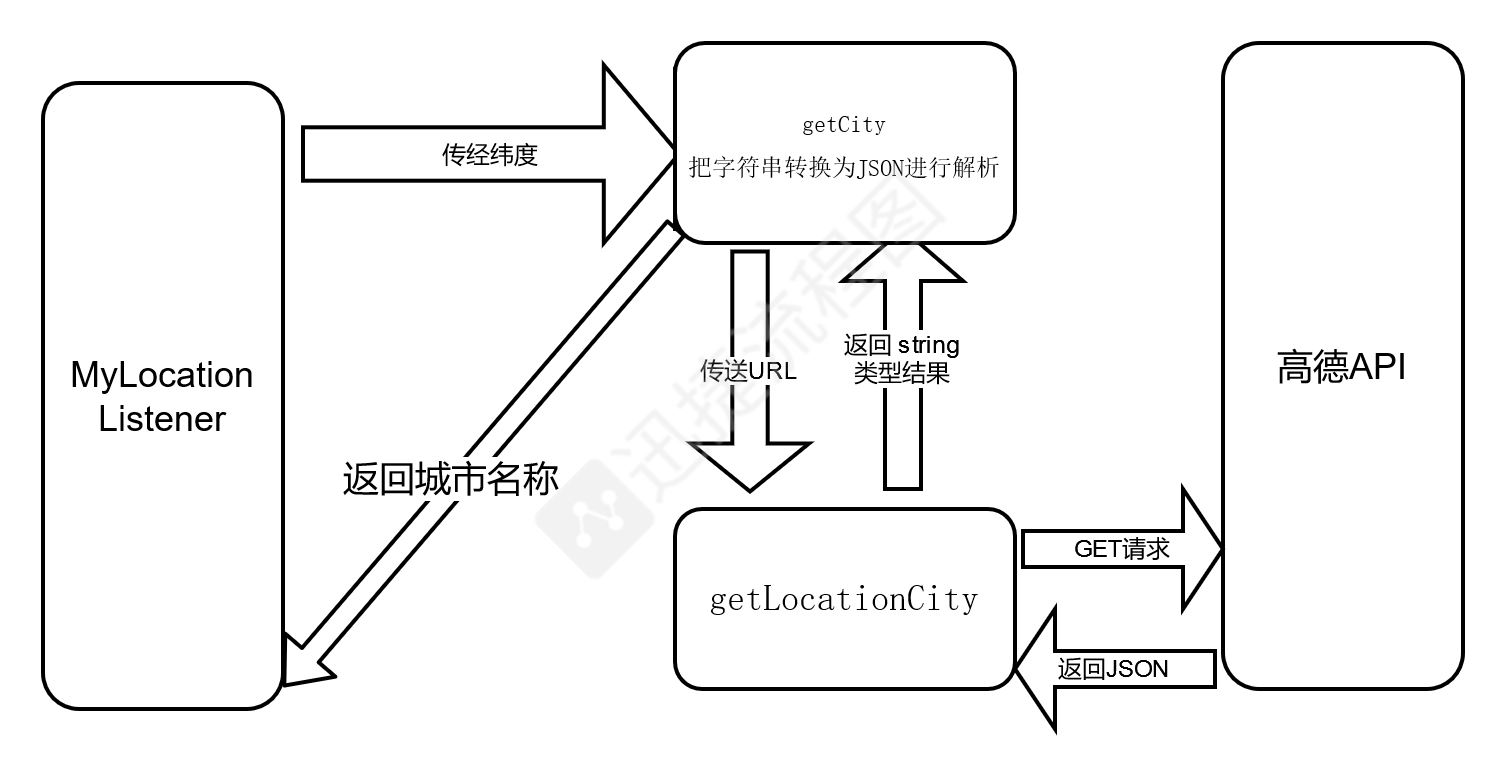
Menu类为主要控制类，MyLocationListener 和OnClick 类都是该类的成员类。由于本页面中用户选择了一个城市后要跳转到天气页面。故考虑编写OnClick类实现View.OnClickListener

接口(重写 onClick方法)来对按钮进行监听。同时由于在页面跳转的时候要把城市名称传过去，所以还要进行跨页面传参。考虑在mobile\_navigation.xml文件中设置页面间传参。在Menu类中把要穿走的city装载后通过controller.navigate 传走，在天气页面类NotificationsFragment中通过getArguments() 接收传来的城市名称。

在Menu类中还有一个关键的功能，识别用户所在城市。此功能需要用户的定位功能。而在Android上进行定位大致有三种方法：使用Android 自带的SDK进行GPS定位，Android 自带的SDK进行网络定位，导入高德/百度等第三方 SDK进行定位。在开发过程中，由于网络定位的精准度高且较稳定，原先考虑使用网络定位。但是测试程序过程中定位结果常常是null ，经查阅资料由于国内的手机厂商将Google的网络服务组件移除了。后又转为采用高德SDK定位，但是由于导入配环境未成功而不得不采用获取手机GPS定位。

采用GPS定位首先要获取用户的GPS权限。在MainActivity类里面写入了，判断若未给予本APP定位权限则需要进行请求。在Menu类中编写 MyLocationListener 类实现 LocationListener 接口，持续对获取用户GPS位置请求进行监听。一旦接收到用户所在的经纬度将交付给 getLocationCity 类处理。GetLocationCity 类中getCity方法接收经纬度并调用同类中的方法getReq 向高德地理编码API发送GET请求。高德API返回JSON数据后经过处理返回一个StringBuilder给getCity方法。该方法将其转换为JSON对象后进行解析，返回用户所在城市名称给Menu类。

流程图如下



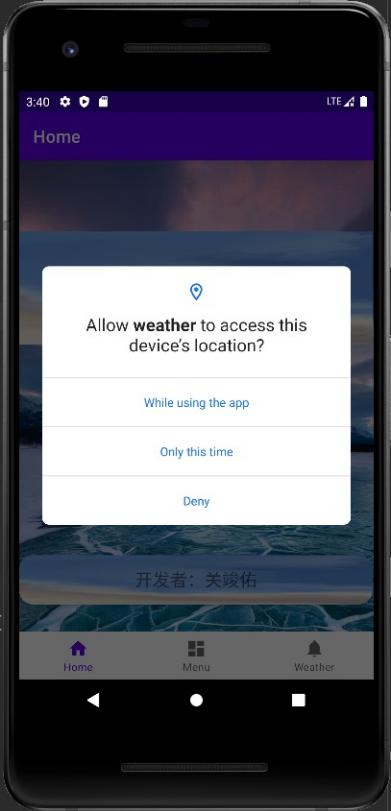
当用户进入了天气页面时有三个关键类：NotificationsFragment ， getRequest , getCityCode 。 getCityCode 类用于GET请求高德接口，用城市名称请求获取高德地理编码。当从上一个页面传来城市名称时，需要调用此类来获取该城市的地理编码。getRequest类用于使用城市地理编码请求高德天气API获取三天天气信息。当NotificationsFragment类获得了地理编码后调用getRequest类获取天气信息。并且获取用户手机的系统时间判断是夜间还是白天。最后通过操纵XML的样式显示天气信息。

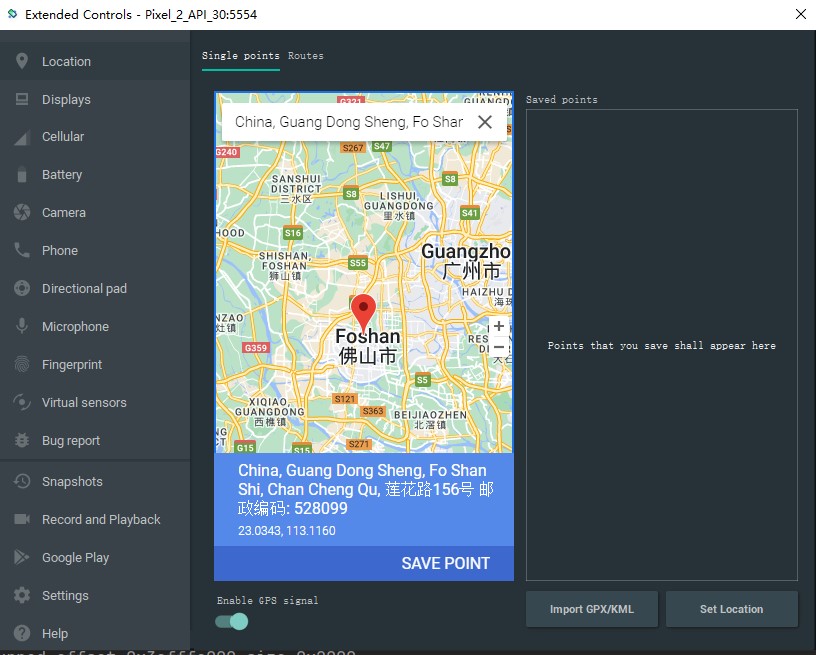
1. **系统调试**
2. **使用说明及编程体会**

使用说明：

1. 在安卓/鸿蒙手机上安装好apk后，点击apk启动。给予定位权限
2. 选择城市或等待定位成功后选择城市
3. 跳转到天气页面查看该城市天气

以下为在Android模拟器上运行的情况(模拟器的GPS位置被设定为佛山市内)

编程体会：

在编写该Android app的过程中经过许多困难及阻碍，都通过查阅资料以及不懈的尝试去解决。由于第一次接触Android开发，经验能力不足，因此仍有不少问题无法解决。

在请求高德API时，Android抛出了异常不允许在主线程中调用网络。意在防止网络阻塞主线程导致程序“假死”。考虑过开启子线程进行网络操作，但是由于知识不足难以解决线程间的通信，将请求的结果返回给主线程而放弃。最后采用折衷方法，使用Android严苛模式来在主线程强制执行网络操作。

在进行定位获取时，尝试了多种方法，因为自身能力问题环境没配成功以及客观上的阻碍导致被迫选择了手机GPS定位方法。

在测试过程中，发现onLocationChanged函数不断地获取手机GPS定位，每刷新一次就重新向高德API发送请求，导致不过多久API调用量便达到5000。考虑到天气预报APP不必频繁获取定位数据(因为用户不可能在打开APP期间频繁改变所在城市)，所以我在该类设置了一个静态的flag变量， 通过静态成员函数初始化为false. 当获取到城市时赋为true ,则APP前台使用期间只需要取出城市即可，且设置当flag 为 true 时不再执行onLocationChanged函数，避免了重复大量请求。

由于本人对XML样式不够熟悉，开发出的app仅能适配个别模拟器内的机型，无法对许多手机屏幕进行适配，导致手机上可能会出现样式偏差。

本项目已在GitHub开源，GitHub地址: https://github.com/guanjunyou/weatherApk

1. **附：源程序**

由于源程序过多，在此仅附上关键代码。

**MainAcitivity.java**

package com.example.pro2;  
  
import android.Manifest;  
import android.content.pm.PackageManager;  
import android.os.Bundle;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
import com.google.android.material.bottomnavigation.BottomNavigationView;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.navigation.NavController;  
import androidx.navigation.Navigation;  
import androidx.navigation.ui.AppBarConfiguration;  
import androidx.navigation.ui.NavigationUI;  
import com.example.pro2.databinding.ActivityMainBinding;  
  
public class MainActivity extends AppCompatActivity {  
  
 private ActivityMainBinding binding;  
  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
  
 super.onCreate(savedInstanceState);  
  
  
  
 binding = ActivityMainBinding.inflate(getLayoutInflater());  
 setContentView(binding.getRoot());  
  
 BottomNavigationView navView = findViewById(R.id.nav\_view);  
 // Passing each menu ID as a set of Ids because each  
 // menu should be considered as top level destinations.  
 AppBarConfiguration appBarConfiguration = new AppBarConfiguration.Builder(  
 R.id.navigation\_home, R.id.navigation\_dashboard, R.id.navigation\_notifications)  
 .build();  
 NavController navController = Navigation.findNavController(this, R.id.nav\_host\_fragment\_activity\_main);  
 NavigationUI.setupActionBarWithNavController(this, navController, appBarConfiguration);  
 NavigationUI.setupWithNavController(binding.navView, navController);  
  
  
 if (ContextCompat.checkSelfPermission(this,  
 Manifest.permission.ACCESS\_FINE\_LOCATION)  
 != PackageManager.PERMISSION\_GRANTED) {  
 //如果应用之前请求过此权限但用户拒绝了请求，此方法将返回 true。  
 if (ActivityCompat.shouldShowRequestPermissionRationale(this,  
 Manifest.permission.ACCESS\_FINE\_LOCATION)) {//这里可以写个对话框之类的项向用户解释为什么要申请权限，并在对话框的确认键后续再次申请权限.它在用户选择"不再询问"的情况下返回false  
 } else {  
 //申请权限，字符串数组内是一个或多个要申请的权限，1是申请权限结果的返回参数，在onRequestPermissionsResult可以得知申请结果  
 ActivityCompat.requestPermissions(this,  
 new String[]{Manifest.permission.ACCESS\_FINE\_LOCATION,}, 1);  
 }  
 }  
  
  
 }  
  
}

**Home.java**

package com.example.pro2.ui.home;  
  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.TextView;  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.fragment.app.Fragment;  
import androidx.lifecycle.Observer;  
import androidx.lifecycle.ViewModelProvider;  
import com.example.pro2.databinding.FragmentHomeBinding;  
  
public class Home extends Fragment {  
  
 private HomeViewModel homeViewModel;  
 private FragmentHomeBinding binding;  
  
 public View onCreateView(@NonNull LayoutInflater inflater,  
 ViewGroup container, Bundle savedInstanceState) {  
 homeViewModel =  
 new ViewModelProvider(this).get(HomeViewModel.class);  
  
 binding = FragmentHomeBinding.inflate(inflater, container, false);  
 View root = binding.getRoot();  
  
 final TextView textView = binding.textHome;  
 homeViewModel.getText().observe(getViewLifecycleOwner(), new Observer<String>() {  
 @Override  
 public void onChanged(@Nullable String s) {  
 textView.setText(s);  
 }  
 });  
 return root;  
 }  
  
 @Override  
 public void onDestroyView() {  
 super.onDestroyView();  
 binding = null;  
 }  
}

**Menu.java**

package com.example.pro2.ui.dashboard;  
  
import android.content.Context;  
import android.location.LocationManager;  
import android.os.Bundle;  
import android.os.StrictMode;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.Button;  
import android.widget.TextView;  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.fragment.app.Fragment;  
import androidx.lifecycle.Observer;  
import androidx.lifecycle.ViewModelProvider;  
import androidx.navigation.NavController;  
import androidx.navigation.Navigation;  
import com.example.pro2.R;  
import com.example.pro2.databinding.FragmentDashboardBinding;  
import android.annotation.SuppressLint;  
  
import android.location.Location;  
import android.location.LocationListener;  
  
import com.example.pro2.MyApplication;  
  
public class Menu extends Fragment {  
  
 private DashboardViewModel dashboardViewModel;  
 public FragmentDashboardBinding binding;  
 public String city = "尚未获取";  
 private String lat = ""; //纬度  
 private String lon = ""; //经度  
 private Double latitue = 0.0;  
 private Double longtitue = 0.0;  
 public static boolean flag;// 是否获取到了经纬度  
 public static String localCity;  
  
 public static void init()  
 {  
 Menu.flag = false;  
 }  
  
  
 public class MyLocationListener implements LocationListener  
 {  
 @Override  
 public void onLocationChanged(@NonNull Location location) {  
  
 double late = location.getLatitude();  
 double lone = location.getLongitude();  
  
 if(late != 0.0 && lone!=0.0 && !Menu.flag)  
 {  
 latitue = late;  
 longtitue = lone;  
 //System.out.println(latitue + " " + longtitue);  
 lat = String.valueOf(latitue);  
 lon = String.valueOf(longtitue);  
 System.out.println("经纬度"+lat+" "+lon);  
  
  
 getLocationCity local = new getLocationCity();  
 try {  
 String[] localCity = local.getCity(lon,lat);  
 // System.out.println(localCity);  
 System.out.println("城市："+localCity[0]+"代码："+localCity[1]);  
 binding.button6.setText("我所在的城市: "+localCity[0]);  
 city = localCity[0];  
 Menu.localCity = city;  
 System.out.println(city);  
 Menu.flag = true;  
  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 System.out.println("city:"+city);  
  
 }  
  
 }  
  
 @Override  
 public void onStatusChanged(String provider, int status, Bundle extras) {  
  
 }  
  
 @Override  
 public void onProviderEnabled(String provider) {  
  
 }  
  
 @Override  
 public void onProviderDisabled(String provider) {  
  
 }  
 }  
  
 @SuppressLint({"MissingPermission", "SetTextI18n"})  
 public View onCreateView(@NonNull LayoutInflater inflater,  
 ViewGroup container, Bundle savedInstanceState) {  
  
 StrictMode.ThreadPolicy policy=new StrictMode.ThreadPolicy.Builder().permitAll().build();  
 StrictMode.setThreadPolicy(policy);  
  
 dashboardViewModel =  
 new ViewModelProvider(this).get(DashboardViewModel.class);  
  
 binding = FragmentDashboardBinding.inflate(inflater, container, false);  
 View root = binding.getRoot();  
  
 final TextView textView = binding.textDashboard;  
 dashboardViewModel.getText().observe(getViewLifecycleOwner(), new Observer<String>() {  
 @Override  
 public void onChanged(@Nullable String s) {  
 textView.setText(s);  
 }  
 });  
  
  
  
 if(!Menu.flag)  
 {  
 System.out.println("进来了！！");  
 Context context = MyApplication.getContext();  
  
 String serviceName = Context.LOCATION\_SERVICE;  
 LocationManager myLocationManager = (LocationManager)context.getSystemService(serviceName);  
  
  
 if (myLocationManager.isProviderEnabled(LocationManager.GPS\_PROVIDER ))  
 {  
 // netLocation = myLocationManager.getLastKnownLocation(LocationManager.GPS\_PROVIDER);  
 myLocationManager.requestLocationUpdates(LocationManager.GPS\_PROVIDER, 0L, 0, (LocationListener) new MyLocationListener());  
 }  
 else  
 System.out.println("GPS 关闭");  
 // }  
 }  
 else  
 {  
 binding.button6.setText("我所在的城市: "+Menu.localCity);  
 }  
  
  
  
  
  
 final Button btn3 = (Button) root.findViewById(R.id.button3);  
 final Button btn4 = (Button) root.findViewById(R.id.button4);  
 final Button btn5 = (Button) root.findViewById(R.id.button5);  
 final Button btn6 = (Button) root.findViewById(R.id.button6);  
  
 OnClick onClick = new OnClick();  
 btn3.setOnClickListener(onClick);  
 btn4.setOnClickListener(onClick);  
 btn5.setOnClickListener(onClick);  
 btn6.setOnClickListener(onClick);  
  
//9uy67  
 return root;  
 }  
  
 @Override  
 public void onDestroyView() {  
 super.onDestroyView();  
 binding = null;  
 }  
  
 public class OnClick implements View.OnClickListener{  
  
  
 @Override  
 public void onClick(View v) {  
 switch (v.getId()){  
  
 case R.id.button3:  
 //System.out.println("北京");  
 city = "北京";  
 binding.textView8.setText("当前城市：北京");  
 break;  
 case R.id.button4:  
 // System.out.println("广州");  
 city = "广州";  
 binding.textView8.setText("当前城市：广州");  
  
 break;  
 case R.id.button5:  
 city = "佛山";  
 binding.textView8.setText("当前城市：佛山");  
 break;  
 case R.id.button6:  
 city = Menu.localCity;  
 binding.textView8.setText("当前城市："+city);  
 break;  
 default:  
 break;  
 }  
 System.out.println("看这里");  
 System.out.println(Menu.flag);  
 if(!Menu.flag)  
 {  
 city = "北京";  
 }  
  
 NavController controller = Navigation.findNavController(v);  
 Bundle bundle = new Bundle();  
 System.out.println(city+"传走");  
 bundle.putString("getcity",city);  
 NavController navController = Navigation.findNavController(v);  
 controller.navigate(R.id.action\_navigation\_dashboard\_to\_navigation\_notifications,bundle);//第2个带参数  
 }  
  
  
 }  
  
  
}

**getLocationCity.java**

package com.example.pro2.ui.dashboard;  
  
import org.json.JSONArray;  
import org.json.JSONObject;  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import java.net.HttpURLConnection;  
import java.net.URL;  
  
  
public class getLocationCity {  
 /\*\*  
 \* 发送GET请求  
 \* @param urlToRead  
 \* @return  
 \* @throws IOException  
 \*/  
 public StringBuilder getReq (String urlToRead) throws IOException {  
 StringBuilder result = new StringBuilder();  
 URL url = new URL(urlToRead);  
 HttpURLConnection conn = (HttpURLConnection) url.openConnection();  
 conn.setRequestMethod("GET");  
 try (BufferedReader reader = new BufferedReader(  
 new InputStreamReader(conn.getInputStream()))) {  
 for (String line; (line = reader.readLine()) != null; ) {  
 result.append(line);  
 }  
 }  
 //System.out.println(result.toString());  
 // return result.toString();+  
  
 return result;  
 }  
  
 //type : name 城市名称 code 城市代码  
  
 /\*\*  
 \* 通过经纬度获取城市名称  
 \* @param lon  
 \* @param lat  
 \* @return  
 \* @throws Exception  
 \*/  
 public String[] getCity(String lon,String lat) throws Exception  
 {  
 //System.out.println(getHTML("https://restapi.amap.com/v3/weather/weatherInfo?city=110101&key=7e5182d424aed687b5157f62eae915f2"));  
 String Url = "https://restapi.amap.com/v3/geocode/regeo?location=";  
 Url += lon;  
 Url += ",";  
 Url += lat;  
 Url += "&key=7e5182d424aed687b5157f62eae915f2";  
 // StringBuilder str = GetReq("https://restapi.amap.com/v3/weather/weatherInfo?city=110101&key=7e5182d424aed687b5157f62eae915f2&extensions=all");  
 System.out.println(Url);  
 StringBuilder str = getReq(Url);  
 System.out.println(str.toString());  
 String s = str.toString();  
  
 JSONObject jsonObject = new JSONObject(s);  
  
 JSONObject regeocode = jsonObject.getJSONObject("regeocode");  
 JSONObject addressComponent = regeocode.getJSONObject("addressComponent");  
 String adcode = (String) addressComponent.get("adcode");  
 String city = (String) addressComponent.get("city");  
 if(city.equals(""))  
 {  
 city = (String) addressComponent.get("province");  
 }  
  
 String[] ans = new String[2];  
 ans[0] = city;  
 ans[1] = adcode;  
 //System.out.println(ans[0]+ans[1]);  
 return ans;  
 }  
}

**Fragment\_dashboard.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:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".ui.dashboard.DashboardFragment">  
  
 <TextView  
 android:id="@+id/text\_dashboard"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="8dp"  
 android:layout\_marginTop="8dp"  
 android:layout\_marginEnd="8dp"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"/>  
 <LinearLayout  
 android:orientation="vertical"  
 android:layout\_width="410dp"  
 android:layout\_height="737dp" tools:layout\_editor\_absoluteY="-2dp" tools:layout\_editor\_absoluteX="3dp"  
 tools:ignore="MissingConstraints">  
 <androidx.cardview.widget.CardView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:cardBackgroundColor="#ccc"  
 app:cardCornerRadius="@dimen/activity\_horizontal\_margin"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
 <TextView  
 android:text="当前城市：北京"  
 android:layout\_width="match\_parent"  
 android:layout\_height="74dp" android:id="@+id/textView8"  
 android:textSize="40dp"  
 android:gravity="center"  
 android:background="@mipmap/c"  
 />  
 </androidx.cardview.widget.CardView>  
 <Button  
 android:text="北京市"  
 android:layout\_width="match\_parent"  
 android:layout\_height="97dp" android:id="@+id/button3"  
 android:alpha="0.5"  
 android:textSize="40dp"  
 />  
 <Button  
 android:text="广州市"  
 android:layout\_width="match\_parent"  
 android:layout\_height="89dp" android:id="@+id/button4"  
 android:alpha="0.5"  
 android:textSize="40dp"  
 />  
 <Button  
 android:text="佛山市"  
 android:layout\_width="match\_parent"  
 android:layout\_height="80dp" android:id="@+id/button5"  
 android:alpha="0.5"  
 android:textSize="40dp"  
 />  
 <TextView  
 android:text=""  
 android:layout\_width="match\_parent"  
 android:layout\_height="38dp" android:id="@+id/textView9"/>  
 <Button  
 android:text="我所在的城市：获取中..."  
 android:layout\_width="match\_parent"  
 android:layout\_height="88dp" android:id="@+id/button6"  
 android:alpha="0.5"  
 android:textSize="25dp"  
 />  
 </LinearLayout>  
</androidx.constraintlayout.widget.ConstraintLayout>

**NotificationsFragment.java**

package com.example.pro2.ui.notifications;  
  
import android.graphics.Color;  
//import android.media.tv.TimelineRequest;  
import android.os.Bundle;  
import android.os.StrictMode;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.TextView;  
import androidx.annotation.NonNull;  
import androidx.annotation.Nullable;  
import androidx.fragment.app.Fragment;  
import androidx.lifecycle.Observer;  
import androidx.lifecycle.ViewModelProvider;  
import com.example.pro2.R;  
import com.example.pro2.databinding.FragmentNotificationsBinding;  
  
import java.sql.Time;  
import java.util.Calendar;  
import java.util.HashMap;  
import java.util.Map;  
import java.util.Timer;  
  
public class NotificationsFragment extends Fragment {  
  
 private NotificationsViewModel notificationsViewModel;  
 private FragmentNotificationsBinding binding;  
 private String[] weather;  
  
 public View onCreateView(@NonNull LayoutInflater inflater,  
 ViewGroup container, Bundle savedInstanceState) {  
  
 StrictMode.ThreadPolicy policy=new StrictMode.ThreadPolicy.Builder().permitAll().build();  
 StrictMode.setThreadPolicy(policy);  
  
  
 notificationsViewModel =  
 new ViewModelProvider(this).get(NotificationsViewModel.class);  
  
 binding = FragmentNotificationsBinding.inflate(inflater, container, false);  
 View root = binding.getRoot();  
  
 final TextView textView = binding.textNotifications;  
 notificationsViewModel.getText().observe(getViewLifecycleOwner(), new Observer<String>() {  
 @Override  
 public void onChanged(@Nullable String s) {  
 textView.setText(s);  
 }  
 });  
  
  
  
  
// // Android 4.0 之后不能在主线程中请求HTTP请求 要开子线程  
// new Thread(new Runnable(){  
// @Override  
// public void run() {  
// getRequest Request = new getRequest();  
// try {  
// weather = Request.getJSON(0, "110101");  
// // binding.textView14.setText(weather);  
// } catch (Exception e) {  
// e.printStackTrace();  
// }  
// }  
// }).start();I]K=8  
  
  
 //https://restapi.amap.com/v3/geocode/geo?key=7e5182d424aed687b5157f62eae915f2&city=北京市&address=北京市  
  
 String getcity = getArguments().getString("getcity");  
 System.out.println(getcity+"传过来了");  
 notificationsViewModel.city = getcity; //接收城市  
 //binding.textView4.setText(notificationsViewModel.city); //设置城市  
  
 //以下为请求高德接口获取城市代码  
 getCityCode CityCode = new getCityCode();  
 String code = "" ;  
 try {  
 code = CityCode.getCode(getcity);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
  
  
 //获取当前时间判断是白天还是晚上  
 Calendar calendar = Calendar.getInstance();  
 int hour = calendar.get(Calendar.HOUR\_OF\_DAY);  
 String weatherStr , temperature;  
  
  
 getRequest Request = new getRequest();  
 try {  
 weather = Request.getJSON(0, code);  
 // binding.textView14.setText(weather);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
  
 System.out.println(hour);  
 String Time = "白天";  
 if(hour >= 6 && hour <= 18)  
 {  
 //白天  
 weatherStr = weather[0];  
 temperature = weather[1]+"℃ ";  
 Time = "白天";  
 }  
 else  
 {  
 //黑夜  
 weatherStr = weather[2];  
 temperature = weather[3]+"℃ ";  
 Time = "夜间";  
 }  
 binding.textView4.setText(notificationsViewModel.city+" "+Time); //设置城市  
  
 binding.textView14.setText(weatherStr);  
 binding.textView15.setText(temperature);  
  
 //以下为加载 mipmap里面的图片  
  
 if(weatherStr.equals("雷阵雨"))  
 binding.imageView.setImageResource(R.mipmap.thunder);  
 else if("晴".equals(weatherStr))  
 binding.imageView.setImageResource(R.mipmap.sunny);  
 else if("小雨".equals(weatherStr))  
 binding.imageView.setImageResource(R.mipmap.samllrain);  
 else if("中雨".equals(weatherStr))  
 binding.imageView.setImageResource(R.mipmap.midrain);  
 else  
 binding.imageView.setImageResource(R.mipmap.cloud);  
  
  
 //明天天气  
 getRequest Request1 = new getRequest();  
 try {  
 weather = Request1.getJSON(1, code);  
 // binding.textView14.setText(weather);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
  
 //System.out.println(weather[2]);  
  
 String tomorrow = "明天 ";  
 tomorrow += weather[0];  
 tomorrow += "/";  
 tomorrow +=weather[2];  
 tomorrow += " ";  
 tomorrow += weather[1];  
 tomorrow += "℃~";  
 tomorrow += weather[3];  
 tomorrow += "℃ ";  
 binding.textView6.setText(tomorrow);  
  
 //后天天气  
 getRequest Request2 = new getRequest();  
 try {  
 weather = Request2.getJSON(2, code);  
 // binding.textView14.setText(weather);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
  
 String after\_tomorrow = "后天 ";  
 after\_tomorrow += weather[0];  
 after\_tomorrow += "/";  
 after\_tomorrow +=weather[2];  
 after\_tomorrow += " ";  
 after\_tomorrow += weather[1];  
 after\_tomorrow += "℃~";  
 after\_tomorrow += weather[3];  
 after\_tomorrow += "℃ ";  
 binding.textView7.setText(after\_tomorrow);  
  
 return root;  
 }  
  
 @Override  
 public void onDestroyView() {  
 super.onDestroyView();  
 binding = null;  
 }  
  
 public static void main(String[] args) {  
  
 }  
  
  
  
  
}

**getCityCode.java**

package com.example.pro2.ui.notifications;  
  
import org.json.JSONArray;  
import org.json.JSONObject;  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import java.net.HttpURLConnection;  
import java.net.URL;  
  
  
public class getCityCode {  
 public StringBuilder getReq (String urlToRead) throws IOException {  
 StringBuilder result = new StringBuilder();  
 URL url = new URL(urlToRead);  
 HttpURLConnection conn = (HttpURLConnection) url.openConnection();  
 conn.setRequestMethod("GET");  
 try (BufferedReader reader = new BufferedReader(  
 new InputStreamReader(conn.getInputStream()))) {  
 for (String line; (line = reader.readLine()) != null; ) {  
 result.append(line);  
 }  
 }  
 //System.out.println(result.toString());  
 // return result.toString();  
 return result;  
 }  
  
 /\*\*  
 \* 通过城市名称获取城市地理编码  
 \* @param CityName  
 \* @return  
 \* @throws Exception  
 \*/  
 public String getCode(String CityName) throws Exception  
 {  
 //System.out.println(getHTML("https://restapi.amap.com/v3/weather/weatherInfo?city=110101&key=7e5182d424aed687b5157f62eae915f2"));  
 String Url = "https://restapi.amap.com/v3/geocode/geo?key=7e5182d424aed687b5157f62eae915f2&city=";  
 Url += CityName;  
 Url += "&address=";  
 Url += CityName;  
 // StringBuilder str = GetReq("https://restapi.amap.com/v3/weather/weatherInfo?city=110101&key=7e5182d424aed687b5157f62eae915f2&extensions=all");  
 StringBuilder str = getReq(Url);  
 System.out.println(str.toString());  
 String s = str.toString();  
  
 JSONObject jsonObject = new JSONObject(s);  
 // System.out.println(jsonObject);  
 // System.out.println(jsonObject.getJSONObject("forecasts").getJSONObject("0").getJSONObject("casts").getJSONObject("0"));  
  
 JSONArray jsonArray = (JSONArray) jsonObject.getJSONArray("geocodes");  
 JSONObject JSONObject1 = jsonArray.getJSONObject(0);  
 //System.out.println(JSONObject1);  
 //JSONObject codeJson = JSONObject1.getJSONObject("adcode");  
 String ans = (String) JSONObject1.get("adcode");  
 System.out.println(ans);  
 //String ans = codeJson.get("adcode");  
 return ans;  
 }  
}

**getRequest.java**

package com.example.pro2.ui.notifications;  
  
//import android.util.JsonReader;  
//import org.json.JSONObject;  
//import com.alibaba.fastjson.JSONObject;  
//import com.alibaba.fastjson.JSONArray;  
import org.json.JSONArray;  
import org.json.JSONObject;  
  
//import com.alibaba.fastjson.JSONObject;  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import java.net.HttpURLConnection;  
import java.net.MalformedURLException;  
import java.net.URL;  
  
public class getRequest {  
 public StringBuilder GetReq (String urlToRead) throws IOException {  
 StringBuilder result = new StringBuilder();  
 URL url = new URL(urlToRead);  
 HttpURLConnection conn = (HttpURLConnection) url.openConnection();  
 conn.setRequestMethod("GET");  
 try (BufferedReader reader = new BufferedReader(  
 new InputStreamReader(conn.getInputStream()))) {  
 for (String line; (line = reader.readLine()) != null; ) {  
 result.append(line);  
 }  
 }  
 //System.out.println(result.toString());  
 // return result.toString();  
 return result;  
 }  
  
 /\*\*  
 \* 通过城市名称及day获取天气信息  
 \* @param day  
 \* @param cityCode  
 \* @return  
 \* @throws Exception  
 \*/  
 //传入 预报日期 第0-3天 ， 以及城市的高德天气代码  
 public String[] getJSON(int day,String cityCode) throws Exception  
 {  
 //System.out.println(getHTML("https://restapi.amap.com/v3/weather/weatherInfo?city=110101&key=7e5182d424aed687b5157f62eae915f2"));  
 String Url = "https://restapi.amap.com/v3/weather/weatherInfo?city=";  
 Url = Url + cityCode;  
 Url = Url + "&key=7e5182d424aed687b5157f62eae915f2&extensions=all";  
 // StringBuilder str = GetReq("https://restapi.amap.com/v3/weather/weatherInfo?city=110101&key=7e5182d424aed687b5157f62eae915f2&extensions=all");  
 StringBuilder str = GetReq(Url);  
 System.out.println(str.toString());  
 String s = str.toString();  
 //JSONObject jsonObject = JSONObject.parseObject(s);  
 // System.out.println(jsonObject.getJSONObject("lives").get("weather"));  
 JSONObject jsonObject = new JSONObject(s);  
 System.out.println(jsonObject);  
 // System.out.println(jsonObject.getJSONObject("forecasts").getJSONObject("0").getJSONObject("casts").getJSONObject("0"));  
  
 JSONArray jsonArray = (JSONArray) jsonObject.getJSONArray("forecasts");  
 JSONObject JSONObject1 = jsonArray.getJSONObject(0);  
 System.out.println(JSONObject1);  
 JSONArray jsonArray1 = (JSONArray) JSONObject1.getJSONArray("casts");  
 JSONObject JSONObject2 = jsonArray1.getJSONObject(day);  
 System.out.println(JSONObject2);  
 System.out.println(JSONObject2.get("dayweather"));  
  
 String[] ans = new String[4];  
 ans[0] = (String) JSONObject2.get("dayweather");  
 ans[1] = (String) JSONObject2.get("daytemp");  
 ans[2] = (String) JSONObject2.get("nightweather");  
 ans[3] = (String) JSONObject2.get("nighttemp");  
 return ans;  
 }  
}//

**Fragment\_notifications.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:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".ui.notifications.NotificationsFragment">  
  
 <LinearLayout  
 android:orientation="vertical"  
 android:layout\_width="411dp"  
 android:layout\_height="734dp" tools:layout\_editor\_absoluteX="0dp"  
 app:layout\_constraintBottom\_toTopOf="@+id/text\_notifications" tools:ignore="MissingConstraints,NotSibling">  
 <TextView  
 android:id="@+id/text\_notifications"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="8dp"  
 android:layout\_marginTop="8dp"  
 android:layout\_marginEnd="8dp"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 />  
 <LinearLayout  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="93dp">  
 <ImageView  
 android:src="@mipmap/rain"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent" android:id="@+id/imageView" android:layout\_weight="1"/>  
 <TextView  
 android:text="晴天"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent" android:id="@+id/textView14" android:layout\_weight="1"  
 android:textSize="40dp"  
 android:gravity="center"  
 />  
 <TextView  
 android:text="27℃ "  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent" android:id="@+id/textView15" android:layout\_weight="1"  
 android:textSize="40dp"  
 android:gravity="center"  
 />  
 </LinearLayout>  
  
 <TextView  
 android:text="北京 白天"  
 android:layout\_width="match\_parent"  
 android:layout\_height="78dp" android:id="@+id/textView4"  
 android:textSize="35dp"  
 android:gravity="center"  
 />  
 <androidx.cardview.widget.CardView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:cardBackgroundColor="#ccc"  
 app:cardCornerRadius="@dimen/activity\_horizontal\_margin"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <TextView  
 android:text="明天 大雨/中雨 27℃~25℃ "  
 android:layout\_width="414dp"  
 android:layout\_height="85dp" android:id="@+id/textView6"  
 android:textSize="25dp"  
 android:gravity="center"  
 android:background="@mipmap/c"  
 />  
  
  
 </androidx.cardview.widget.CardView>  
 <TextView  
 android:text=""  
 android:layout\_width="match\_parent"  
 android:layout\_height="48dp" android:id="@+id/textView10"/>  
 <androidx.cardview.widget.CardView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:cardBackgroundColor="#ccc"  
 app:cardCornerRadius="@dimen/activity\_horizontal\_margin"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent">  
  
 <TextView  
 android:text="后天 大雨/中雨 27℃~28℃ "  
 android:layout\_width="match\_parent"  
 android:layout\_height="82dp" android:id="@+id/textView7"  
 android:textSize="25dp"  
 android:gravity="center"  
 android:background="@mipmap/e"  
 android:alpha="0.8"  
 />  
  
  
  
  
  
  
 </androidx.cardview.widget.CardView>  
 <LinearLayout  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="308dp">  
 </LinearLayout>  
 <TextView  
 android:text="TextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" android:id="@+id/textView11"/>  
 <TextView  
 android:text="TextView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" android:id="@+id/textView3"/>  
  
  
 </LinearLayout>  
</androidx.constraintlayout.widget.ConstraintLayout>