Министерство науки и высшего образования Российской Федерации

Федерального государственного бюджетное образовательное учреждение высшего образования

«Российской экономический университет имени Г. В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ.**

ПРАКТИЧЕСКИЕ РАБОТЫ № 15

ДИСЦИПЛИНА: «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Тема: Работа с датчиками и создание гироскопа.

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

Листов:

|  |  |  |
| --- | --- | --- |
| Выполнил студент |  | Проверил преподаватель |
| Группа П50-3-18 |  | \_\_\_\_\_\_\_\_\_\_\_А.О.Лясников |
| Пахомов Даниил Александрович |  | «\_\_\_» \_\_\_\_\_\_\_\_2020 года |

Москва 2021

ПРАКТИЧЕСКАЯ РАБОТА №15

Работа с Widget

Цель: создать программу, которая будет взаимодействовать с widget.

1. AndroiManifest

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <manifest xmlns:android="http://schemas.android.com/apk/res/android"  package="com.example.widget">   <uses-permission android:name="android.permission.CAMERA" />  <uses-permission android:name="android.permission.FLASHLIGHT" />   <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.Widget">  <receiver android:name=".ToastWidget">  <intent-filter>  <action android:name="android.appwidget.action.APPWIDGET\_UPDATE" />  </intent-filter>   <meta-data  android:name="android.appwidget.provider"  android:resource="@xml/toast\_widget\_info" />  </receiver>  <receiver android:name=".CameraWidget">  <intent-filter>  <action android:name="android.appwidget.action.APPWIDGET\_UPDATE" />  </intent-filter>   <meta-data  android:name="android.appwidget.provider"  android:resource="@xml/camera\_widget\_info" />  </receiver>   <activity android:name=".MainActivity">  <intent-filter>  <action android:name="android.intent.action.MAIN" />   <category android:name="android.intent.category.LAUNCHER" />  </intent-filter>  </activity>  </application>  </manifest> |

1. MainActivity

|  |
| --- |
| package com.example.widget  import android.Manifest import android.content.pm.PackageManager import androidx.appcompat.app.AppCompatActivity import android.os.Bundle import android.widget.Toast import androidx.core.app.ActivityCompat import androidx.core.content.ContextCompat  class MainActivity : AppCompatActivity() {  override fun onCreate(savedInstanceState: Bundle?) {  super.onCreate(savedInstanceState)  setContentView(R.layout.activity\_main)  // условие на проверку есть ли разрешение на камеру  if (ContextCompat.checkSelfPermission(this, Manifest.permission.CAMERA) != PackageManager.PERMISSION\_GRANTED) {  Toast.makeText(this, "Нет разрешения на камеру", Toast.LENGTH\_SHORT).show()  ActivityCompat.requestPermissions(this, arrayOf(Manifest.permission.CAMERA), 111)  }  }   override fun onRequestPermissionsResult(requestCode: Int, permissions: Array<out String>, grantResults: IntArray)  {  when(requestCode){  111->{  if(grantResults.isNotEmpty() && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {  Toast.makeText(this, "YES!", Toast.LENGTH\_SHORT).show()  }else{  Toast.makeText(this, "NO!", Toast.LENGTH\_SHORT).show()  }  }  }  } } |

1. CameraWidget

|  |
| --- |
| package com.example.widget  import android.Manifest import android.app.PendingIntent import android.appwidget.AppWidgetManager import android.appwidget.AppWidgetProvider import android.content.Context import android.content.Intent import android.content.pm.PackageManager import android.hardware.camera2.CameraManager import android.util.Log import android.view.View import android.widget.RemoteViews import android.widget.Toast import android.widget.ToggleButton import androidx.core.content.ContextCompat  const val widget: String = "But" var poweroff: Boolean = false class CameraWidget : AppWidgetProvider() {  override fun onUpdate(  context: Context,  appWidgetManager: AppWidgetManager,  appWidgetIds: IntArray  ) {  // There may be multiple widgets active, so update all of them  for (appWidgetId in appWidgetIds) {  updateAppWidget(context, appWidgetManager, appWidgetId)  }  }   override fun onEnabled(context: Context) {  // Enter relevant functionality for when the first widget is created  }   override fun onDisabled(context: Context) {  // Enter relevant functionality for when the last widget is disabled  }   override fun onReceive(context: Context?, intent: Intent?) {  if (!poweroff) {   if (!context!!.packageManager.hasSystemFeature(PackageManager.FEATURE\_CAMERA\_FLASH)) {  Toast.makeText(context, "Нет камеры", Toast.LENGTH\_SHORT).show()  return  }   if ((ContextCompat.checkSelfPermission(context,  Manifest.permission.CAMERA) != PackageManager.PERMISSION\_GRANTED)) {  Toast.makeText(context, "Нет разрешения на камеру", Toast.LENGTH\_SHORT).show()  return  }   if ((context!!.getSystemService(Context.CAMERA\_SERVICE) as CameraManager) == null) {  Toast.makeText(context, "Нет камеры", Toast.LENGTH\_SHORT).show()  return  }  (context!!.getSystemService(Context.CAMERA\_SERVICE) as CameraManager).  setTorchMode((context!!.getSystemService(Context.CAMERA\_SERVICE) as CameraManager).cameraIdList[0], true  )  poweroff = true   } else {  (context!!.getSystemService(Context.CAMERA\_SERVICE) as CameraManager).  setTorchMode((context!!.getSystemService(Context.CAMERA\_SERVICE) as CameraManager).cameraIdList[0], false)  poweroff = false  }   updateAppWidget(context!!,AppWidgetManager.getInstance(context),intent!!.getIntExtra("appWidgetId", 0))   super.onReceive(context, intent)  } }  internal fun updateAppWidget(context: Context,appWidgetManager: AppWidgetManager,appWidgetId: Int) {   val intent: Intent = Intent(context, CameraWidget::class.java)  val views = RemoteViews(context.packageName, R.layout.camera\_widget)   intent.action = widget  intent.putExtra("appWidgetId", appWidgetId)  views.setOnClickPendingIntent(R.id.btn\_flashlight, PendingIntent.getBroadcast(context, 0, intent, 0))   appWidgetManager.updateAppWidget(appWidgetId, views) } |

1. ToastWidget

|  |
| --- |
| package com.example.widget  import android.app.PendingIntent import android.appwidget.AppWidgetManager import android.appwidget.AppWidgetProvider import android.content.Context import android.content.Intent import android.widget.RemoteViews import android.widget.Toast  */\*\*  \* Implementation of App Widget functionality.  \*/* class ToastWidget : AppWidgetProvider() {  override fun onUpdate(context: Context,appWidgetManager: AppWidgetManager,appWidgetIds: IntArray) {  // Там может быть несколько активных виджетов, поэтому обновите их все  for (appWidgetId in appWidgetIds) {  *updateAppWidget1*(context, appWidgetManager, appWidgetId)  }  }   override fun onEnabled(context: Context) {  // Введите соответствующую функциональность для момента создания первого виджета  }   override fun onDisabled(context: Context) {  // Введите соответствующую функциональность для тех случаев, когда последний виджет отключен  }   override fun onReceive(context: Context?, intent: Intent?) {  super.onReceive(context, intent)   Toast.makeText(context, "hehdl", Toast.*LENGTH\_SHORT*).show()  *updateAppWidget*(context!!,AppWidgetManager.getInstance(context),intent!!.getIntExtra("appWidgetId", 0))  } }  internal fun updateAppWidget1(  context: Context,  appWidgetManager: AppWidgetManager,  appWidgetId: Int ) {  val intent: Intent = Intent(context, ToastWidget::class.*java*)  val widgetText = context.getString(R.string.*appwidget\_text*)  // Создание объекта RemoteViews  val views = RemoteViews(context.*packageName*, R.layout.*toast\_widget*)  intent.putExtra("appWidgetId", appWidgetId)  //обработчик нажатии кнопки  views.setOnClickPendingIntent(R.id.*dialog\_button*, PendingIntent.getBroadcast(context, 0,intent, 0))   // Поручите менеджеру виджетов обновить виджет  appWidgetManager.updateAppWidget(appWidgetId, views) } |

1. Activity\_main

|  |
| --- |
| <?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">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Hello World!"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />  </androidx.constraintlayout.widget.ConstraintLayout> |

1. Toast\_widget

|  |
| --- |
| <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="?attr/appWidgetBackgroundColor"  android:padding="@dimen/widget\_margin"  android:theme="@style/ThemeOverlay.Widget.AppWidgetContainer">   <Button  android:id="@+id/btn\_flashlight"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"/> </RelativeLayout> |

1. Camera\_widget

|  |
| --- |
| <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:background="?attr/appWidgetBackgroundColor"  android:padding="@dimen/widget\_margin"  android:theme="@style/ThemeOverlay.Widget.AppWidgetContainer">   <Button  android:id="@+id/dialog\_button"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  /> </RelativeLayout> |

Вывод: научился создавать программу, которая может взаимодействовать с Widget.