ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | С.В. Щекин |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №6 |
| Организация навигации в многоэкранном приложении |
| по курсу: Программирование мобильных устройств |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

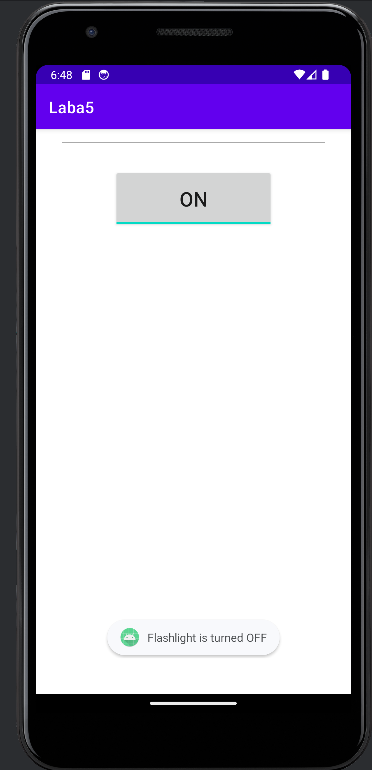
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4134к |  |  |  | Н.А. Костяков |
|  |  |  | подпись, дата |  | инициалы, фамилия |

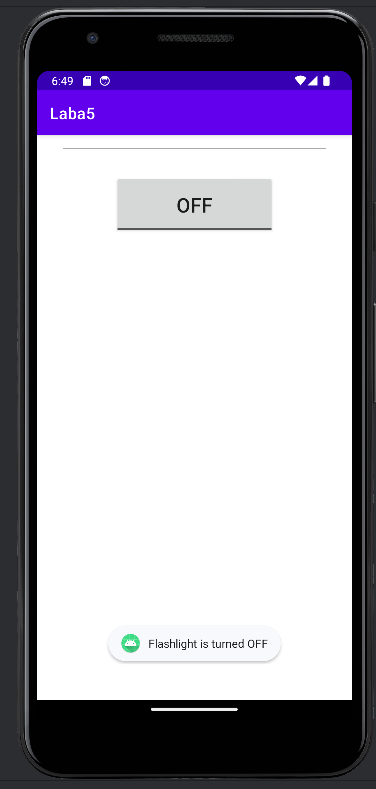
Санкт-Петербург 2024

Задание: Дополнить приложение, разработанное в рамках одной из предыдущих лабораторных работ, добавив в него работу с аппаратным обеспечение мобильного устройства (камера, светодиод подсветки, датчики и т.п.)

Цель работы: Получение навыков использования в разрабатываемых мобильных приложениия доступа к компонентам аппаратного обеспечения мобильных устройств.

## Ход Работы





## Листинг

Activity\_main

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout

    xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout\_width="match\_parent"

    android:layout\_height="match\_parent"

    android:orientation="vertical"

    tools:context=".MainActivity"

    tools:ignore="HardcodedText">

    <View

        android:layout\_width="match\_parent"

        android:layout\_height="1dp"

        android:layout\_marginStart="32dp"

        android:layout\_marginTop="16dp"

        android:layout\_marginEnd="32dp"

        android:background="@android:color/darker\_gray" />

    <ToggleButton

        android:id="@+id/toggle\_flashlight"

        android:layout\_width="200dp"

        android:layout\_height="75dp"

        android:layout\_gravity="center"

        android:layout\_marginTop="32dp"

        android:onClick="toggleFlashLight"

        android:textSize="25sp" />

</LinearLayout>

Main activity

package com.example.laba5;  
  
import android.content.Context;  
import android.hardware.camera2.CameraAccessException;  
import android.hardware.camera2.CameraManager;  
import android.os.Build;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Toast;  
import android.widget.ToggleButton;  
import androidx.annotation.RequiresApi;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
  
 private ToggleButton toggleFlashLightOnOff;  
 private CameraManager cameraManager;  
 private String getCameraID;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // Register the ToggleButton with specific ID  
 toggleFlashLightOnOff = findViewById(R.id.*toggle\_flashlight*);  
  
 // cameraManager to interact with camera devices  
 cameraManager = (CameraManager) getSystemService(Context.*CAMERA\_SERVICE*);  
  
 // Exception is handled, because to check whether  
 // the camera resource is being used by another  
 // service or not.  
 try {  
 // O means back camera unit,  
 // 1 means front camera unit  
 getCameraID = cameraManager.getCameraIdList()[0];  
 } catch (CameraAccessException e) {  
 e.printStackTrace();  
 }  
 }  
  
 // RequiresApi is set because, the devices which are  
 // below API level 10 don't have the flash unit with  
 // camera.  
 @RequiresApi(api = Build.VERSION\_CODES.*M*)  
 public void toggleFlashLight(View view) {  
 if (toggleFlashLightOnOff.isChecked()) {  
 // Exception is handled, because to check  
 // whether the camera resource is being used by  
 // another service or not.  
 try {  
 // true sets the torch in ON mode  
 cameraManager.setTorchMode(getCameraID, true);  
  
 // Inform the user about the flashlight  
 // status using Toast message  
 Toast.*makeText*(MainActivity.this, "Flashlight is turned ON", Toast.*LENGTH\_SHORT*).show();  
 } catch (CameraAccessException e) {  
 // prints stack trace on standard error  
 // output error stream  
 e.printStackTrace();  
 }  
 } else {  
 // Exception is handled, because to check  
 // whether the camera resource is being used by  
 // another service or not.  
 try {  
 // true sets the torch in OFF mode  
 cameraManager.setTorchMode(getCameraID, false);  
  
 // Inform the user about the flashlight  
 // status using Toast message  
 Toast.*makeText*(MainActivity.this, "Flashlight is turned OFF", Toast.*LENGTH\_SHORT*).show();  
 } catch (CameraAccessException e) {  
 // prints stack trace on standard error  
 // output error stream  
 e.printStackTrace();  
 }  
 }  
 }  
 // when you click on button and torch open and  
 // you do not close the torch again this code  
 // will off the torch automatically  
 @RequiresApi(api = Build.VERSION\_CODES.*M*)  
 @Override  
 public void finish() {  
 super.finish();  
 try {  
 // true sets the torch in OFF mode  
 cameraManager.setTorchMode(getCameraID, false);  
  
 // Inform the user about the flashlight  
 // status using Toast message  
  
 } catch (CameraAccessException e) {  
 // prints stack trace on standard error  
 // output error stream  
 e.printStackTrace();  
 }  
 }  
}