

Создание приложения камеры для Android смартфон

Ульянов Егор Андреевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье рассмотрен метод создания камеры для android устройств. Приложение будет разработано в среде разработки Android Studio на языке программирования JavaScript. Практическим результатом является рабочее мобильное приложение камеры.

Ключевые слова: Android, Android Studio, приложение, камера.

Creating camera applications for Android smartphones

Ulianov Egor Andreevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article describes the method of creating a camera for android devices. The application will be developed in the Android Studio development environment in the JavaScript programming language. The bottom line is a working mobile application.

Keywords: Android, Android Studio, application, camera

На сегодняшний день камера является одним из самых часто используемых приложений на смартфонах, большинство людей используют камеру своего смартфона для публикации фото в различных социальных сетях.

Цель данной статьи создать рабочее мобильное приложение с возможностью сделать фотографию или снять видео, и отвечающее всем критериям камеры смартфона, таким как просмотр сделанных ранее фотографии, просмотр видео, с возможностью поставить его на паузу или перемотать. Приложение будет создано на языке программирования JavaScript в среде разработки Android Studio.

Исследованиями в области разработки мобильных приложений занимались многие российские и зарубежные исследователи. А.С. Винокуров, Р.И. Баженов [1] рассмотрели разработку приложений для мобильных устройств. С.К. Заманова, Г.Е. Сейдахметова, Г.Г. Масимова, А.Е. Манатова[2]. Они изучили современные подходы к разработке мобильных приложений. Также они рассмотрели разработку приложения в

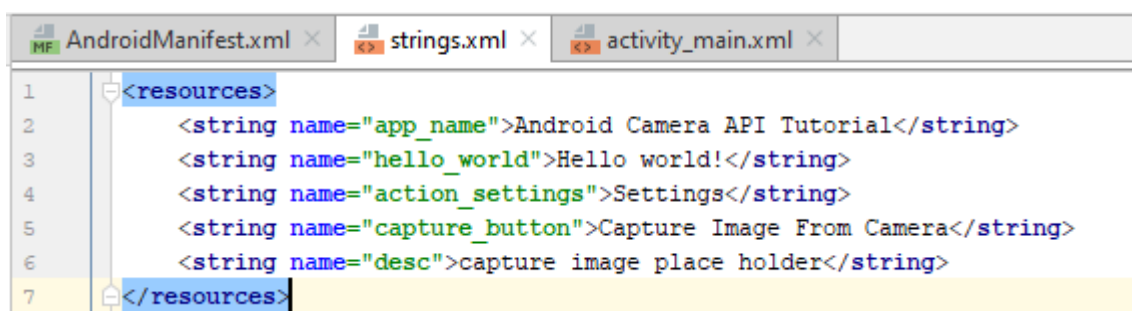
среде Rad Studio XE7. Е.Н.Амиргалиев и др. разработал свою собственную модель отправки информационных сообщений для мобильных операционных систем [3].

Так как нужен доступ для использования некоторого аппаратного ресурса в устройстве, например, камеры, будем использовать разрешения для Android. Обновленная версия «Manifest.xml» файла показана ниже (рис.1).

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="inducesmile.com.androidcameraapitutorial" >
4     <uses-permission android:name="android.permission.CAMERA" />
5     <uses-feature android:name="android.hardware.camera" />
6     <uses-feature android:name="android.hardware.camera.autofocus" />
7     <application
8         android:allowBackup="true"
9         android:icon="@mipmap/ic_launcher"
10        android:label="@string/app_name"
11        android:theme="@style/AppTheme" >
12        <activity
13            android:name=".MainActivity"
14            android:label="@string/app_name" >
15            <intent-filter>
16                <action android:name="android.intent.action.MAIN" />
17
18                <category android:name="android.intent.category.LAUNCHER" />
19            </intent-filter>
20        </activity>
21    </application>
22
23 </manifest>
```

Рисунок 1 – Добавление разрешений

Добавляем данные строки в файл string.xml. Обновленная версия файла показана ниже (рис.2).



```
AndroidManifest.xml x strings.xml x activity_main.xml x
1 <resources>
2     <string name="app_name">Android Camera API Tutorial</string>
3     <string name="hello_world">Hello world!</string>
4     <string name="action_settings">Settings</string>
5     <string name="capture_button">Capture Image From Camera</string>
6     <string name="desc">capture image place holder</string>
7 </resources>
```

Рисунок 2 – обработка запросов

Теперь продолжим разработку основного макета приложения. В файле «activity_main.xml» будем помещать код пользовательского интерфейса приложения. Открываем файл «activity_main.xml» в проекте, который хотим использовать. Добавляем три различных элемента управления View в файл макета. «FlameLayout» будет содержать предварительный просмотр камеры. «ButtonView» будет использоваться для захвата изображения камеры.

«ImageView» будет содержать захваченное изображение из предварительного просмотра камеры (рис.3).

```
<RelativeLayout xmlns="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:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MainActivity"
    android:baselineAligned="false">

    <FrameLayout
        android:id="@+id/camera_preview"
        android:layout_width="match_parent"
        android:layout_height="300dp"/>

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/capture_button"
        android:id="@+id/button"
        android:layout_below="@+id/camera_preview"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="15dp" />

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/captured_image"
        android:layout_below="@+id/button"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="15dp"
        android:contentDescription="@string/desc" />

</RelativeLayout>
```

Рисунок 3 – Добавление элементов управления View

Для предварительного просмотра камеры создадим собственный класс «SurfaceView», который будет привязан к «FrameLayout» как дочерний элемент макета.

Класс будет реализовывать «SurfaceHolder.Callback», и три метода в этом классе переопределяются функциями «SurfaceCreated», «SurfaceChanged», «SurfaceDestroyed»

Инициализируем класс «Camera». В методе «SurfaceCreated» объект камеры открывается и начинается предварительный просмотр камеры. В методе «surfaceDestroyed» предварительный просмотр камеры останавливается, и объект камеры освобождается, чтобы другие приложения могли использовать камеру (рис.4).

```
package inducesmile.com.androidcameraapitutorial;

import android.hardware.Camera;
import android.content.Context;
import android.view.SurfaceHolder;
import android.view.SurfaceView;

import java.io.IOException;

public class ImageSurfaceView extends SurfaceView implements SurfaceHolder.Callback {

    private Camera camera;
    private SurfaceHolder surfaceHolder;

    public ImageSurfaceView(Context context, Camera camera) {
        super(context);
        this.camera = camera;
        this.surfaceHolder = getHolder();
        this.surfaceHolder.addCallback(this);
    }

    @Override
    public void surfaceCreated(SurfaceHolder holder) {
        try {
            this.camera.setPreviewDisplay(holder);
            this.camera.startPreview();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    @Override
    public void surfaceChanged(SurfaceHolder holder, int format, int width, int height) {

    }

    @Override
    public void surfaceDestroyed(SurfaceHolder holder) {
        this.camera.stopPreview();
        this.camera.release();
    }
}
```

Рисунок 4 – Функция камеры

В файле «MainActivity.java» получены экземпляры элементов управления «View». Объект «ImageSurfaceView» был создан и передан как дочерний вид объекту «FrameLayout». Создадим внутренний класс «PictureCallback» и переопределим метод класса «onPictureTaken» класса.

Слушатель события щелчка прикреплен к кнопке. Когда кнопка нажата, она выполнит метод «takePicture» класса «Camera». Объект класса «PictureCallback» передается как один из параметров, требуемых методом (рис.5).

```
public class MainActivity extends ActionBarActivity {

    private ImageSurfaceView mImageSurfaceView;
    private Camera camera;

    private FrameLayout cameraPreviewLayout;
    private ImageView capturedImageHolder;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);

        cameraPreviewLayout = (FrameLayout)findViewById(R.id.camera_preview);
        capturedImageHolder = (ImageView)findViewById(R.id.captured_image);

        camera = checkDeviceCamera();
        mImageSurfaceView = new ImageSurfaceView(MainActivity.this, camera);
        cameraPreviewLayout.addView(mImageSurfaceView);

        Button captureButton = (Button)findViewById(R.id.button);
        captureButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                camera.takePicture(null, null, pictureCallback);
            }
        });
    }

    private Camera checkDeviceCamera(){
        Camera mCamera = null;
        try {
            mCamera = Camera.open();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return mCamera;
    }

    PictureCallback pictureCallback = new PictureCallback() {
        @Override
        public void onPictureTaken(byte[] data, Camera camera) {
            Bitmap bitmap = BitmapFactory.decodeByteArray(data, 0, data.length);
            if(bitmap==null){
                Toast.makeText(MainActivity.this, "Captured image is empty", Toast.LENGTH_LONG).show();
                return;
            }
            capturedImageHolder.setImageBitmap(scaleDownBitmapImage(bitmap, newWidth: 300, newHeight: 200 ));
        }
    };
};
```

Рисунок 5 – создание функций

Разработка приложений для android устройств не является сложным процессом. Достаточно иметь хорошую среду разработки и небольшое знание языка.

Практическим результатом данной статьи получилось рабочее приложение камеры.

Библиографический список

1. Винокуров А.С., Баженов Р.И. Разработка мобильного приложения информационного сайта для абитуриентов и первокурсников университета // Современные научные исследования и инновации. 2015. № 7-2 (51). С. 54-62.
2. Заманова С.К., Сейдахметова Г.Е., Масимова Г.Г., Манатова А.Е. Разработка мобильного приложения в среде Rad Studio XE7 // Труды Международного симпозиума «Надежность и качество». 2015. №1. С. 237-

240.

3. Amirgaliyev E.N., Kalizhanova A.U., Kozbakova A.KH. Development of applications to mobile devices in Android platform // Труды Международного симпозиума «Надежность и качество». 2015. №1. С. 240-242.