

Создание простого приложения для Android на языке Kotlin

Семченко Регина Викторовна

Приамурский государственный университет имени Шолом-Алейхема

студент

Еровлев Павел Андреевич

Приамурский государственный университет имени Шолом-Алейхема

студент

Аннотация

В данной статье описан процесс создания простого приложения на android устройства с помощью языка Kotlin. Практическим результатом имеется рабочее приложение, которое является счетчиком.

Ключевые слова: Kotlin, приложение, андроид, счетчик

Creating a simple Kotlin Android application

Semchenko Regina Viktorovna

Sholom-Aleichem Priamursky State University

student

Erovlev Pavel Andreevich

Sholom-Aleichem Priamursky State University

student

Abstract

This article describes the process of creating a simple application on an android device using the Kotlin language. The bottom line is a working application that is a counter.

Keywords: Kotlin, application, android, counter

Kotlin является официальным языком для создания Приложения для Android начиная с 2017 года. Kotlin – это выразительный и современный язык, который можно рассматривать как язык будущего для создания приложений Android по сравнению с Java.

Цель данной статьи создать Android приложение с Kotlin, простой счетчик с кнопками плюс и минус.

В.И.Макаров провел в своей статье анализ способов для создания пользовательского интерфейса при разработке приложений, которые применяются в среде разработки AndroidStudio [1]. Так же А.И.Долженко совместно с С.А.Глушенко провели анализ целесообразности разработки мобильного приложения для android устройств, а так же разработали

собственное приложение на android [2]. В статье Р.В.Мальчева и С.В.Кривошеева был выполнен анализ архитектурной системы ARM как аппаратной основы для создания симуляторов т/с [3].

Пользовательский интерфейс счетчика, сделанного с Kotlin, будет довольно простым с тремя компонентами: кнопка «плюс» для увеличения счетчика, кнопка минус, чтобы уменьшить счетчик, TextView для отображения значения счетчика(рис.1).

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/
  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="com.ssaurel.kotlincounter.MainActivity">

  <TextView
    android:id="@+id/value"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="100sp"
    android:textStyle="bold"
    android:text=""
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="@id/plusBtn"
    app:layout_constraintBottom_toBottomOf="@id/minusBtn"
  />

  <Button
    android:id="@+id/plusBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="50sp"
    android:text="+"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="50dp"
  />

  <Button
    android:id="@+id/minusBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="50sp"
    android:text="-"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginBottom="50dp"
  />

</android.support.constraint.ConstraintLayout>
```

Рисунок 1 – создание интерфейса

Теперь напишем код на Kotlin. Благодаря плагину «Kotlin Android Extensions» не нужно выполнять вызовы «findViewById ()» для получения ссылок на виды в функции «MainActivity».

Затем определяем свойство для хранения значения счетчика и устанавливаем «OnClickListener» на кнопки «плюс» и «минус», чтобы обновить значение счетчика, когда пользователь нажимает кнопки(рис.2).

```
package com.ssaurel.kotlincounter

import android.os.Bundle
import android.support.v7.app.AppCompatActivity
import kotlinx.android.synthetic.main.activity_main.*

class MainActivity : AppCompatActivity() {

    var id = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        value.setText("" + id)

        plusBtn.setOnClickListener {
            value.setText("" + ++id)
        }

        minusBtn.setOnClickListener {
            value.setText("" + --id)
        }
    }
}
```

Рисунок 2 – добавление функций

Это весь код приложения, следующим этапом может быть тестирование, через эмулятор в среде AndroidStudio(рис.3).



Рисунок 3 – проверка приложения

На этом создание приложение закончено, язык является новым и пока, что легко его использовать, а самое главное, что его можно начать изучать прямо сейчас, пока он еще молодой.

Практическим результатом является созданное приложение, считающее количество нажатий на «+» и вычитающее при нажатии на «-».

Библиографический список

1. Макаров В.И. Особенности разработки пользовательского интерфейса для android-приложений в среде разработки android studio// Современные научные исследования и инновации. 2017. № 7-5 (43). С. 47-55.
2. Долженко А.И., Глушенко С.А. Разработка мобильного приложения для ТСЖ на платформе android // Труды Международного симпозиума «Надежность и качество». 2014. №5. С. 14-20.
3. Мальчева Р.В., Кривошеева С.В. Разработка симуляторов транспортных средств с использованием операционной системы android // Автоматика. Вычислительная техника. 2012. №1. С. 24-30.