



Курсова работа по

Мобилни приложения

Тема: стандартен калкулатор

Изготвен от:

Даниел Делчев, фн:81211, Компютърни Науки,

3 курс

e-mail : danieldelchev95@abv.bg

Летен семестър 2017г

Проверил:

(ас. Христо Христов)

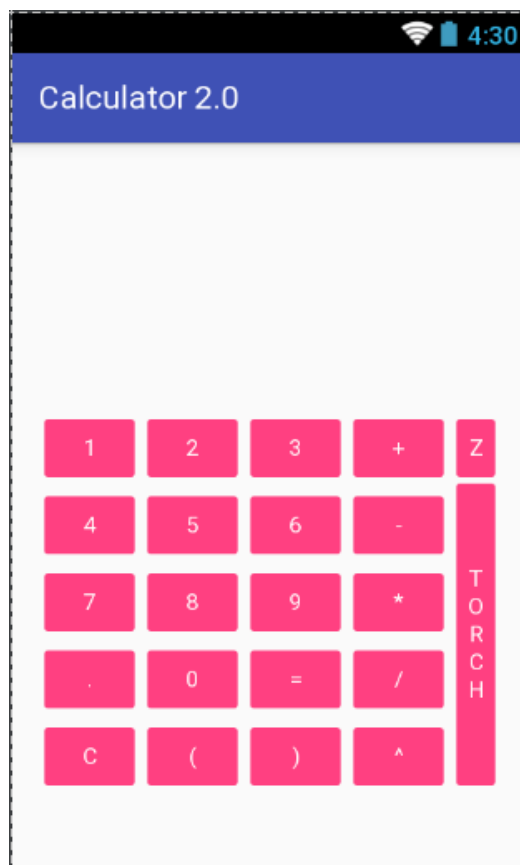
Описание на проекта:

Проекта представлява имплементация на стандартен калкулатор предоставящ възможност за изчисления , чрез стандартните аритметични операции събиране, изваждане, умножение , деление, степенуване (и съответно коренуване). Също така са взети под внимание редът на операциите, като освен това има възможност и за задаване на приоритет чрез скоби. Естествено, има проверка за коректност на входа, като при неправилен аритметичен израз, потребителя бива известен с подходящо съобщение , както и с вибрация на телефона. Освен това за да се демонстрира и работа със сензорите на телефона, е добавена и възможност за включване/изключване на фенера (светкавицата на камерата) на телефонът.

Github repository на проекта:

<https://github.com/DanielDelchev/mobile>

Интерфейс и възможности



Вход от потребителя се получава през бутони. При натискане на бутон, към аритметичния израз се добавя съответният символ на бутона. Бутонът “C” е за нулиране на израза, бутонът “Z” изстрива последно добавеният символ, а бутонът “=” заменя текущия израз с неговата стойност. Бутонът “TORCH” е за включване и изключване на светкавицата. Текущият израз бива изписван чрез текст в полето над бутоните.

Архитектура и реализация

Минималната версия за OS android за проекта е 4.1 Jelly Bean, избрана главно поради използването на физическо устройство за провеждане на тестов, по време на разработка. Изчисленията,

които се провеждат за оценяване стойността на изразите естествено се базира на RPN (reverse polish notation) с 2 стека – 1 за операнди и един за оператори. Тъй като алгоритъма се прилага главно върху изрази с бинарни операции, то за да се вгради унарен оператор – (негация на число) изрази от вида -4+6 биват представени вътрешно като 0-4+6.

Използвани са следните android API-s

```
import android.content.Context;
import android.content.pm.ActivityInfo;
import android.hardware.Camera;
import android.os.Vibrator;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.hardware.Camera.Parameters;
```

Управлението на приложението се съдържа в единствен клас, чиито член данни са низ – представящ текущия аритметичен израз, член данна за връзка с камерата на устройството, както и булева променлива указваща състоянието на светкавицата на камерата на устройството. Override-нати са методите onCreate, onResume, onBackPressed, за извеждане на кратки съобщения при съответните събития, като освен това при onCreate се заключава ориентацията на приложението в портретен режим.

Методът vibrate прави точно това. Използва се при некоректно зададен аритметичен израз за известяване на потребителя, заедно с кратко текстово съобщение. Методът clear нулира низа на израза, а методът update подновява състоянието на текстовото поле над бутоните на приложението за да бъдат отразени промените, ако има извършени такива. Всеки от методите започващи с представката dial добавя конкретен символ към низа на израза и е свързана с някой от бутоните. Методите apply, isop, isdigit, priority са помощни за

метода `calculate_expr`, който изчислява стойността на текущия аритметичен израз, и го заменя с нея. Методът `flash` в зависимост от текущото състояние на светкавицата на устройството, спира или включва фенерчето, като естествено се извършва и проверка за способност на устройството да използва фенерче.

Обосновка на избраните подходи

Тъй като калкулаторът е сравнително стандартен и не е нужно да се поддържат по-сложни аритметични операции, за удобство на представянето на изразите е избран низов формат, който бива превърнат до RPN (reverse polish notation). В случай, когато има не само бинарни операции, а произволни n -арни, обаче би било нужно да се заложи на дървовидно представяне или дори на стекови автомати. Тъй като няма кой знае каква полза от запаметяването на изчисления направени на калкулатор на мобилно устройство, вместо предоставяне на persistence, внимание бе обърнато на работата с подустройствата на мобилното устройство като светкавица и вибриращо устройство, като вибрации се ползват за известяване на некоректен вход, а за стартиране или спиране на фенерчето е предоставен бутон в интерфейсът на приложението.