

Gymnázium, Praha 6, Arabská 14

Programování

Aplikace na kontrolu financí



2017

Josef Vašíčka

Gymnázium, Praha 6, Arabská 14

Arabská 14, Praha 6, 160 00

Ročníková práce

Předmět: Programování

Téma: Aplikace na kontrolu financí

Autor: Josef Vašíčka

Třída: 4. E

Školní rok: 2019/ 2020

Vedoucí práce: Mgr. Jan Lána

Třídní učitel: Mgr. Jana Urzová, Ph.D.

Obsah

Prohlášení.....	1
Anotace.....	2
Abstract.....	3
Úvod.....	4
Zadání.....	5
Popis problematiky	6
Řešení zásadních problémů.....	7
Architektura projektu.....	8
Uživatelské rozhraní	8
MainActivity	8
Expenses_activity	8
Incomes_activity	9
History_data_activity.....	9
Detailed_data.....	9
Converter	10
Objekty reprezentující výdaje a příjmy	11
Expenses a Incomes	11
Použité technologie	12
Jazyk a IDE	12
Platforma aplikace	12
Použité vnější rozhraní	12
SQLite databáze	13
Použité knihovny.....	14
Knihovna MPAndroidChart.....	14
Knihovna Support	14
Závěr.....	15
Seznam Obrázků	16
Zdroje.....	17

Prohlášení

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

Dne _____ v _____

Podpis: _____ (Josef Vašíčka)

Anotace

Tato práce se zabývá vytvořením aplikace sloužící ke kontrole financí. Praktickou částí této práce je aplikace na mobilní zařízení, ve které uživatel může zaznamenávat své příjmy a výdaje. Všechny své výdaje a příjmy, které si uživatel zadá do aplikace si může najít a prohlédnout ve výpisu výdajů nebo v jednom z interaktivních grafů. Součástí aplikace je také převodník měn, který uživateli pomůže s převodem světových měn s aktuálními kurzy.

Abstract

This work is about the creation of an application, which can be used to control finances. The practical part of this work is an application for mobile devices, in which the user can record their incomes and expenses. All the expenses and income that the user enters into the application can be found and viewed in the list of expenses or in one of the interactive graphs. The application also includes a currency converter, which will help the user to convert world currencies with current exchange rates.

Úvod

Jedná se o aplikaci, která by měla uživateli pomoci s dohledem na jeho finance. Do aplikace je možné přidávat své příjmy a výdaje, ke kterým je potřeba doplnit dva údaje a to částku a kategorii (myšleno ve smyslu za co byla daná částka utracena či odkud pochází daný příjem). Aplikace dále pracuje se zadanými daty, ukládá je do databáze. Všechny svoje výdaje a příjmy může sledovat na grafech, které se v aplikaci vyskytují. Z grafů lze vyčíst za co uživatel procentuálně nejvíc utrací. V dalším grafu si uživatel může pomocí zadání měsíce a roku dohledat za co daný měsíc utrácel nejvíce. Lze také na grafu sledovat rozdíl výdajů a příjmů, jinými slovy graf ukazuje zda je uživatel s náklady v mínusu nebo naopak. Aplikace také obsahuje převodník měn, kde uživatel může najít aktuální kurzy světových měn a posléze mezi nimi převádět.

Zadání

Mobilní aplikace ohledně financí. Uživatel bude mít pod dohledem své výdaje, které se rádi do různých kategorií (podle toho na co daně peníze šli). Uživatel bude mít přístup k historii svých výdajů, bude moci pomoci shrnutích (měsíčních, ročních) vidět za co kolik utratil, případně z grafu vyčíst jak ztracené peníze v dané kategorii narůstají či klesají. Další funkce budou zcela informační např. Počítání dani, pomoci zadání několika údajů bude člověk schopen vidět kolik bude muset platit má daních. Dále zde uživatel nalezne aktuální kurzy měn.

Popis problematiky

Aplikace je určena k zaznamenávání výdajů a příjmů uživatele. Člověku tak může pomoci s kontrolováním toku jeho peněz. Lze zde najít hned několik způsobů, jak si daná data prohlédnout. Ať už v grafech či ve výpisu výdajů. Při prvním spuštění aplikace v novém měsíci se uživateli také zobrazí údaje z minulého měsíce, kolik utratil peněz v různých kategoriích. Aplikace má také funkci převodníku měn. Uživatel tak může pozorovat aktuální kurzy nejpoužívanějších měn a následně mezi nimi převádět. Celkově by aplikace měla zlehčit člověku dost často náročnou kontrolu nad jeho financemi.

Řešení zásadních problémů

V této práci byly zásadním problémem bylo vyřešit ukládání výdajů a příjmů, tak aby se dala při opětovném spuštění aplikace znovu načíst. Logicky prvním řešením je vytvoření databáze, do které se dané data budou ukládat. A toto řešení bylo zvoleno i pro tento problém tohoto projektu. Hned v zápětí následoval další problém, jak rozlišovat výdaje a příjmy v databázi, aby se pak dali jednotlivě vypisovat. Pro vyřešení tohoto problému slouží v aplikaci objekty Expense a Income, které mají parametry, které se ukládají do databází. Struktura těchto objektů je podrobně rozepasána v kapitole Architektura projektu. V programu také existuje objekt Balance, který reprezentuje zůstatek uživatele (rozdíl mezi výdaji a příjmy).

Konečné řešení těchto dvou problémů vypadá následovně. Tabulky u databází s výdaji a příjmy mají čtyři sloupce a to: id, category, amount a date. Názvy jednotlivých sloupců naznačují, která data jsou do daných sloupců ukládána: id (číslo řádků), category (kategorie Expense/ Income), amount (utracená/ přijetá částka) a date (datum přidání Expense/ Income). Tabulka, která obsahuje data o Balance (zůstatku), se skládá pouze ze dvou sloupců a to: balance_id (číslo řádků) a balance_amount (hodnota zůstatku). Do jednotlivých sloupců jsou ukládány parametry objektů Expense a Income. Při získávání dat jsou poté data z databází opětovně dosazována, jako parametry těchto objektů, tak aby se s nimi dalo v programu jednodušeji pracovat (vypisovat, měnit atd.).

Architektura projektu

Aplikace je tvořena osmi aktivitami, mezi kterými uživatel přepínat pomocí postraniho rozcestníku. Aktivita se dělí podle jejich vlastností. V aplikaci se nacházejí aktivity, v kterých může uživatel pozorovat své výdaje a příjmy v grafech. V dalších aktivitách může uživatel nalézt data ve formě čistého výpisu z databaze, či aktivitu v které nalezne převaděč měn. Dále se v programu nacházejí tzv. Pop upy (vyskakovací okna), které mají za úkol brát od uživatele data a vkládat je do databází. V poslední řadě se v aplikaci nacházejí pomocné třídy, které buď reprezentují nějaký objekt nebo pracují s databázemi.

Uživatelské rozhraní

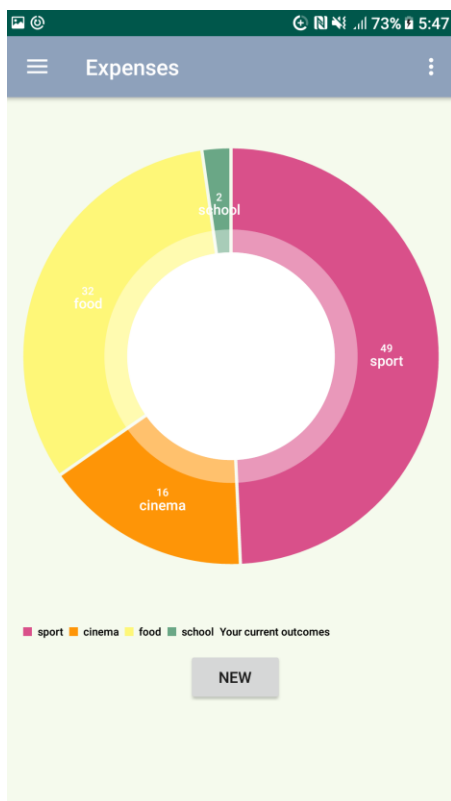
Uživatelské rozhraní aplikace bylo vytvořeno za pomoci XML (Extensible Markup Language). Každá aktivita má svůj vlastní layout (rozpoložení), tedy je k ní přiřazen XML soubor, v kterém je daný layout definovaný. V aplikaci se nacházejí přesně osm aktivit. Mezi aktivitami lze proklikávat díky postraničnímu navigačnímu pruhu.

MainActivity

Jedná se o základní aktivitu, která se automaticky spustí, jako první po spuštění aplikace. V aktivitě, kterou v bočním průzkumníku nalezneme pod názvem Home, se nachází tlačítko. Na daném tlačítku se nachází textové pole na, kterém se zobrazuje momentální zůstatek uživatele. Kliknutím na dané tlačítko se na obrazovce objeví vyskakovací okno s grafem, z kterého lze vyčíst, jak se po jednotlivých vkladech/ výdajích měnil zůstatek. V této aktivitě se dále nacházejí obrázkové tlačítka jejichž prostřednictvím se uživatel může dostat na další aktivity.

Expenses activity

V aplikaci ji uživatel nalezne pod jménem Expenses. Jedná se o aktivitu, kde nachází tzv. koláčový graf, ze kterého lze vyčíst poměr utracených částek v daných kategoriích. V grafu jsou políčka zastupující jednotlivé kategorie, každá má svoji vlastní barvu. Na políčku se dále nachází číslo značící, kolik z celkové množství utracených peněz spadá do dané kategorie. V Expenses_activity se dále nachází tlačítko. Po kliknutí na tlačítko se objeví vyskakovací okno, kde uživatel může přidat nový výdej. Do dvou textových polí zadá uživatel peněžní částku a kategorii. Kategorii si také může vybrat tzv. spinner, ve kterém uživatel najde všechny své dosavadní použité kategorie. Pomocí tlačítka pak uživatel výdaj přidá do databáze a do grafu.



Obrázek 1: Expenses_activity

Obrázek 2: Expenses vyskakovací okno

Incomes activity

V aplikaci ji uživatel nalezne pod jménem Incomes. Jedná se o aktivitu, kde nachází tzv. koláčový graf, ze kterého lze vyčíst poměr přijmutých částek z daných kategorií. V grafu jsou políčka zastupující jednotlivé kategorie, každá má svoji vlastní barvu. Na políčku se dále nachází číslo značící, kolik z celkové množství přijatých peněz spadá do dané kategorie. V Incomes_activity se dále nachází tlačítko. Po kliknutí na tlačítko se objeví vyskakovací okno, kde uživatel může přidat nový příjem. Do dvou textových polí zadá uživatel peněžní částku a kategorii. Pomocí tlačítka pak uživatel vklad přidá do databáze a do grafu.

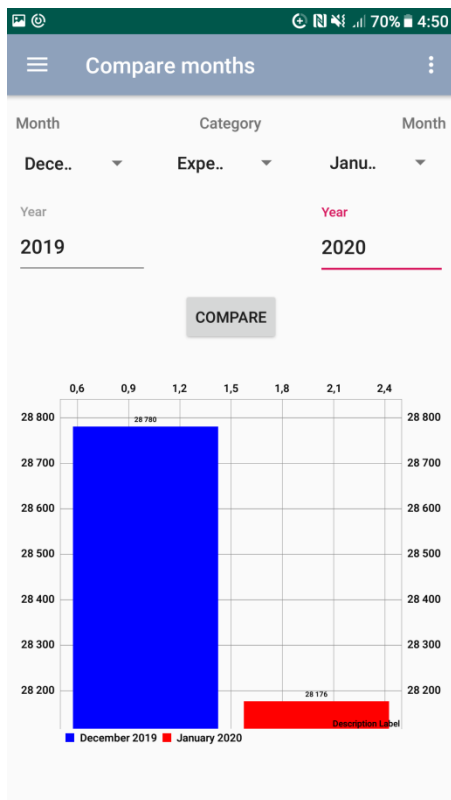
History data activity

V aplikaci ji uživatel nalezne pod jménem Data. Jedná se o aktivitu, v které si uživatel pomocí spinner vybere měsíc a do textového pole napíše rok a na obrazovce se mu zobrazí všechny výdaje v daném měsíci. K výdajům je připsáno přesné datum, částka a kategorie.

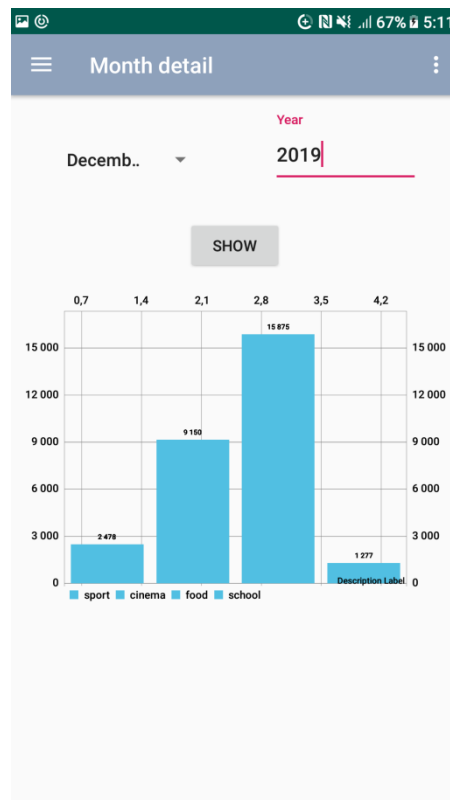
Detailed data

V aplikaci ji uživatel nalezne pod jménem Detailed data. Tato aktivita slouží jako rozcestník mezi dalšími dvěmi kategoriemi. Nacházejí se zde dvě tlačítka: Compare Months a Month Detail. Po zmáčknutí tlačítka Compare Months se objeví aktivita Detailed_com_data. V této aktivitě si uživatel pomocí zadání roků a měsíců, může porovnat výdaje v jednotlivých kategoriích za dané měsíce. Porovnání je

znázorněno ve sloupcovém grafu. Po zmáčknutí tlačítka Month Detail se objeví aktivita Detailed_data_month. V této aktivitě se po zadání roku a měsíce objeví sloupcový graf se sloupci pro částky u tracených v jednotlivých kategoriích za daný měsíc.



Obrázek 3: Detailed_com_data



Obrázek 4: Detailed_data_month

Converter

V aplikaci ji uživatel nalezne pod jménem Currency converter. Jedná se o aktivitu, která slouží k převodu mezi uživatelem zvolenými měnami. V aplikaci lze najít tyto měny: USD (americký dolar), Eur (euro), JPY (japonský jen), GBP (britská libra), AUD (australský dolar) a CZK (česká koruna). Převádění je velmi jednoduché, uživatel do textového pole zadá částku a poté si ve spinnerech vybere dané dvě měny. Také zde lze najít přehled kurzů všech výše zmíněných měn vůči dolaru.

Objekty reprezentující výdaje a příjmy

Expenses a Incomes

Expenses a Incomes jsou objekty, které v aplikaci zastupují výdaje a příjmy uživatele. Oba tyto objekty mají tři parametry a to amount (utracená či přijatá částka), date (datum kdy byl výdaj/ příjem vložen do aplikace) a category (kategorie/„oblast“ v které uživatel peníze utratil či odkud mu peníze přišly). Pro vklad výdajů slouží Expenses_activity, v které se nachází graf, z kterého uživatel může vyčíst poměr mezi výdaji v jednotlivých kategoriích. Pomocí zmačknutí tlačítka se zobrazí vyskakovací okno, v kterém uživatel může vložit nový Expense. Pro Income to funguje obdobně. V aktivitě Income_activity se pomocí tlačítka dostane uživatel do vyskakovacího okna, kde může zadat nový příjem. Objekty Expense a Income jsou posléze ukládány do databází, respektive jsou tam ukládány jejich atributy. Objekty lze načíst pomocí načtení jejich atributů z databáze a jejich následné přidělení k daným objektům. Detailnější informace o práci s databázemi se nacházejí v kapitole Použité technologie: SQLite databáze.

Použité technologie

Jazyk a IDE

Aplikace byla vytvořena ve vývojovém prostředí Android Studio, které je založeno na IntelliJ IDEA. Vzhledem k vybranému prostředí se nabízely tři možnosti programovacích jazyků v kterých mohl být program psán, a to Java, Kotlin a C++. Vzhledem k autorově znalosti jednotlivých jazyků byla vybrána Java, která se stala programovacím jazykem, v kterém byla napsána tato práce.

Platforma aplikace

Aplikace je určena na mobilní zařízení s operačním systémem Android. Nejnižší verze Androidu, kterou aplikace podporuje, je verze 7.0 Nougat. Tím pádem aplikaci lze spustit na zařízení, které má tuto či novější verzi.

Použité vnější rozhraní

V aplikaci bylo použito API ExchangeRate-API. Bylo využito na získávání aktuálních kurzů měn. API vrátí JSON objekt, v kterém se nachází aktuální kurzy měn vůči tak zvané základní měně (v případě této práce se jedná o americký dolar).

SQLite databáze

V aplikaci se nacházejí tři databáze. Do jednotlivých databází jsou ukládány data o výdajích, příjmech a zůstatku. Databáze byly vytvořeny pomocí SQLite. Jedná se o relační databázový systém (databázi založená na tabulkách). K usnadnění práce s datbázemi se v aplikaci nacházejí třídy DHelper, pomocí jehož metod jsou do databází data zapisována, vypisována a tříděna. Databáze jsou vytvořeny pomocí SQL příkazů, které jsou v podobě Stringu vloženy, jako parametr funkce execSQL, která zpustí daný SQL příkaz (nesmí se jednat o příkaz, který vrací jakékoliv data např.: SELECT). K zisku či k přepisu dat následně slouží objekt SQLiteDatabase

```
public class DHelper2 extends SQLiteOpenHelper {

    private static final String TABLE_NAME = "outcomesHistory";
    private static final String col1 = "id";
    private static final String col2 = "category";
    private static final String col3 = "amount";
    private static final String col4 = "date";

    public DHelper2(Context context) {
        super(context, TABLE_NAME, factory: null, version: 1);
    }

    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase) {
        String createTable = "CREATE TABLE " + TABLE_NAME + "(" + col1 + " INTEGER PRIMARY KEY, " + col2 + " TEXT, " + col3 + " TEXT, " + col4 + " TEXT" + ")";
        sqLiteDatabase.execSQL(createTable);
    }
}
```

Obrázek 5: Třída DHelper

```
public void addData(Expense outcome) {

    SQLiteDatabase sqLiteDatabase = this.getWritableDatabase();
    ContentValues contentValues = new ContentValues();
    contentValues.put(col2, outcome.getCategory());
    contentValues.put(col3, outcome.getAmount());
    contentValues.put(col4, outcome.getDate());

    sqLiteDatabase.insert(TABLE_NAME, nullColumnHack: null, contentValues);
    sqLiteDatabase.close();
}
```

Obrázek 6: Metoda addData třídy DHelper

Použité knihovny

Knihovna MPAndroidChart

Jedná se o knihovnu pro Android od autora Philippa Jahody. Pomocí této knihovny jsou v aplikaci tvořeny grafy. Knihovna obsahuje hned několik druhů grafů, v této práci byly použity LineChart, PieChart, BarChart. Některé druhy grafy z této knihovny jsou interaktivní, podporují gesta: lze přiblížit hodnoty, u PieChartu lze rotovat celý graf. V aplikaci je knihovna využita na zobrazování jednoduchých dat v grafech.

Knihovna Support

Jedná se o knihovnu, která obsahuje další podknihovny poskytující zpětně kompatibilní verze rozhraní API pro platformu Android. Také obsahuje funkce, které jsou k dispozici pouze díky API této knihovny. Každá podknihovna knihovny Support je zpětně kompatibilní s kteroukoliv verzí Android API. Knihovny z toho setu jsou v aplikaci použity na designové prvky, jako jsou např.: textové pole a další různá zobrazení.

Závěr

V tomto ročníkovém projektu se mi podařilo sprovoznit vše, až na jednu výjimku, co měl program podle zadání obsahovat. Tou výjimkou je počítání daní, kdy jsem přecenil své schopnosti, nejedná se o nic náročného na programování, ale jedná se o samotné počítání daní, které je náročné na teorii. Jinak jsem s programem spokojený. Do budoucna bych chtěl zapracovat na designu celého programu. Způsob kterým by se aplikace mohla zlepšit je načítání dat o výdajích a příjmech rovnou z elektronických bankovních výpisů.

Seznam Obrázků

Obrázek 1: Expenses_activity	9
Obrázek 2: Expenses vyskakovací okno.....	9
Obrázek 3: Detailed_com_data.....	10
Obrázek 4: Detailed_month_data.....	10
Obrázek 5: Třída DHelper.....	13
Obrázek 6: Metoda addData třídy DHelper.....	13

Zdroje

Oficiální dokumentace Android:

<https://developer.android.com/docs>

Oficiální dokumentace knihovny MPAndroidChart:

<https://weeklycoding.com/mpandroidchart-documentation/>

Oficiální dokumentace Support Library:

<https://developer.android.com/topic/libraries/support-library>

Oficiální dokumentace SQLite:

<https://www.sqlite.org/docs.html>

ExchangeRate-API:

<https://www.exchangerate-api.com/>

Joe. 2018. Android Chart Example APP using MPAndroidChart. *Javapapers.com*.

[Online] 15. listopad 2018. [Citace: 14. dubna 2020]

<https://javapapers.com/android/android-chart-example-app-using-mpandroidchart/>

Autor: neuveden. Rok: neuveden. Android – SQLite Database. *tutorialspoint.com*.

[Online] neuvedeno. [Citace: 10. dubna 2020]

https://www.tutorialspoint.com/android/android_sqlite_database.htm