|  |  |  |
| --- | --- | --- |
| Logo__SSPU_2016_Barva | | |
| **Závěrečná studijní práce**  **dokumentace** | | |
| **Mobilní informační aplikace** | | |
| Michael Hluchník | | |
| [místo pro vložení obrázku] | | |
|  | |  |
| **Obor:** | 18-20-M/01 INFORMAČNÍ TECHNOLOGIE  se zaměřením na počítačové sítě a programování | |
| **Třída:**  **Školní rok:** | IT4  2018/2019 | |

#### Poděkování

* *poděkování (například vedoucímu práce).*

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité   
informační zdroje.

Souhlasím, aby tato studijní práce byla použita k výukovým účelům na Střední průmyslové   
a umělecké škole v Opavě, Praskova 399/8.

V Opavě 31. 12. 2016

*podpis autora práce*

**ANOTACE**

Cílem projektu bylo vytvořit mobilní aplikaci vypisující informace o daných předmětech ze hry Warframe. Zajišťuje jednoduché a přehledné vypisování dat, aby uživatel neměl problém najít jakýkoliv předmět ve hře. Aplikace byla vyvinuta pomocí jazyka JAVA ve vývojovém prostředí Android studio. Data jsou brána z internetových stránek v podobě JSON souborů. Prioritou při tvorbě aplikace byl výsledný vzhled aplikace, aby byla přehledná a jednoduše ovladatelná.

**Klíčová slova:** mobilní aplikace, android studio, JSON soubory, JAVA

OBSAH

[Poděkování 2](#_Toc532727048)

[Úvod 5](#_Toc532727049)

[1 Teoretická a metodická východiska 6](#_Toc532727050)

[1.1 Definice pojmů 6](#_Toc532727051)

[2 Využité technologie 8](#_Toc532727052)

[2.1 Android studio 8](#_Toc532727053)

[2.2 JAVA 8](#_Toc532727054)

[2.3 Soubory JSON 8](#_Toc532727055)

[2.4 Android OS 9](#_Toc532727056)

[2.5 Sdk 9](#_Toc532727057)

[3 Způsoby řešení a použité postupy 10](#_Toc532727058)

[3.1 Začátky v Android studiu 10](#_Toc532727059)

[3.1.1 Nastavení pro vývojáře v mobilu 10](#_Toc532727060)

[3.2 Práce s Activity 10](#_Toc532727061)

[3.2.1 Activity 10](#_Toc532727062)

[3.2.2 Vytvoření aktivity 10](#_Toc532727063)

[3.2.3 Přepínaní mezi aktivitami 11](#_Toc532727064)

[3.2.4 Posílání proměnné mezi aktivitami 11](#_Toc532727065)

[3.3 JSON parsing 11](#_Toc532727066)

[3.3.1 AsyncTask 12](#_Toc532727067)

[3.3.2 Přístup k internetu 12](#_Toc532727068)

[3.3.3 Parsování 12](#_Toc532727069)

[4 Výsledky řešení, výstupy, uživatelský manuál 13](#_Toc532727070)

[4.1 Ukázka kódu 13](#_Toc532727071)

[Závěr 14](#_Toc532727072)

[Seznam použitýCH INFORMAČNÍCH ZDROJů 15](#_Toc532727073)

[Seznam příloh 17](#_Toc532727074)

Úvod

Text úvodu

# Teoretická a metodická východiska

## Definice pojmů

**Software:**

* + Je v informatice sada všech počítačových programů používaných v [počítači](https://cs.wikipedia.org/wiki/Po%C4%8D%C3%ADta%C4%8D), které provádějí nějakou činnost. Software lze rozdělit na [systémový software](https://cs.wikipedia.org/wiki/Syst%C3%A9mov%C3%BD_software), který zajišťuje chod samotného počítače a jeho styk s okolím a na aplikační software, se kterým buď pracuje uživatel počítače, nebo zajišťuje řízení nějakého stroje.

**Stream dat:**

* + Tok dále nespecifikovaných dat

**Debugger:**

* + Je softwarový nástroj, který se používá pro hledání chyb při vývoji software ve fázi ladění. Většinou je možné zobrazit [zdrojový kód](https://cs.wikipedia.org/wiki/Zdrojov%C3%BD_k%C3%B3d) laděného programu, takže je ihned možné vidět místo, kde se objevila [programátorská chyba](https://cs.wikipedia.org/wiki/Program%C3%A1torsk%C3%A1_chyba).

**Parsing:**

* + Neboli rozdělení celku na několik částí.

**String:**

* + Neboli řetězec. V programátorských jazycích se často jedná o datový typ, který označuje posloupnost znaků.
* Teoretická část (též metodologická) obsahuje dosavadní poznatky k danému problému, definici pojmu, formulaci hypotéz, výběr metod, které použijete, a důvody pro jejich použití.
* nadpis volte podle zaměření své práce – např.:
  + Zpracování zvuku na počítači
  + Tvorba elektronických kurzů v prostředí Moodle
  + Drupal 7 – tvorba modulů
  + LDAP a adresářové služby
* využívejte správných citací z odborné literatury, případně internetových zdrojů

# Využité technologie

## Android studio

Android studio je vývojové prostředí založené na Intellij ADEA které podporuje programovací jazyky jako je JAVA a Groovy. Také podporuje programovací jazyky C++ a Kotlin. Pro tento projekt jsem si vybral jazyk JAVA. Vytvořenou firmou Google pro platformy Windows, Mac OS X a Linux. Android studio obsahuje mnoho různých funkcí, které usnadňují programování. Mezi užitečné nástroje patří například debugger, kterým si můžeme otestovat vytvořený program před tím, než jej nahrajeme do procesoru.

Toto vývojové prostředí jsem si vybral kvůli jednoduchosti a taky kvůli tomu, že se mi prostě líbilo. Byla to moje první zkušenost s tímto vývojovým prostředím a můžu doporučit.

## JAVA

Je objektově orientovaný [programovací jazyk](https://cs.wikipedia.org/wiki/Programovac%C3%AD_jazyk), který vyvinula firma Sun Microsystems. Java nejpopulárnější programovací jazyk. Díky své [přenositelnosti](https://cs.wikipedia.org/wiki/P%C5%99enositelnost) je používán pro [programy](https://cs.wikipedia.org/wiki/Po%C4%8D%C3%ADta%C4%8Dov%C3%BD_program), které mají pracovat na různých systémech počínaje čipovými kartami přes [mobilní telefony](https://cs.wikipedia.org/wiki/Mobiln%C3%AD_telefon) a různá zabudovaná zařízení aplikace pro desktopové počítače až po rozsáhlé distribuované systémy pracující na řadě spolupracujících počítačů rozprostřené po celém světě. Je také multiplatformní.

## Soubory JSON

JSON (JavaScript Object Nation) je způsob zápisu dat nezávislý na počítačové platformě. Je určený pro přenos dat, která mohou být organizována v polích nebo agregována v objektech. Používají se pro přenos dat. Je člověkem čitelný a jednoduše editovatelný.

Tento typ souboru jsem zvolil pro jednoduchou manipulaci s daty a jednoduchosti čtení v jazyce JAVA.

## Android OS

Je [mobilní operační systém](https://cs.wikipedia.org/wiki/Mobiln%C3%AD_opera%C4%8Dn%C3%AD_syst%C3%A9m) založený na [jádře Linuxu](https://cs.wikipedia.org/wiki/Linux_(j%C3%A1dro)), který je dostupný jako otevřený software. Je používán na smartphonech, tabletech, [chytrých televizích](https://cs.wikipedia.org/wiki/Smart_TV) a dalších zařízeních. Jeho vývoj vede firma [Google](https://cs.wikipedia.org/wiki/Google) pod hlavičkou konsorcia firem Open Handset Alliance. Má mnoho verzí, podle kterých se odvádí název. Systém Android má největší zastoupení na světě mezi všemi operačními systémy.

Vytváření aplikace pro tento operační systém jsem si vybral z důvodu vlastnictví mobilu, který užívá tento operační systém.

## Sdk

Software development kit (sdk)je typická sada vývojových nástrojů umožňující vytváření aplikací pro jisté softwarové balíčky, frameworky, platformy, počítačové systémy, herní konzole, operační systémy nebo podobnou platformu. Pro vytvoření vlastní aplikace se bez toho neobejdete. Většinou jsou přiděleny k přímo k vývojovému prostředí.

# Způsoby řešení a použité postupy

## Začátky v Android studiu

Nejprve jsem se musel naučit používat android studio. Jak vytvářet nový projekt jak ho debuggovat a přímo spouštět. K tomu jsem potřeboval emulátor nebo zařízení, které používalo operační systém android. To jsem dokázal až po zhlédnutí několika videí.

### Nastavení pro vývojáře v mobilu

Abych mohl vyvíjet aplikaci pro můj vlastní telefon, musel jsem si ho takzvaně přepnout do režimu „Pro vývojáře“. Toto mi umožnilo instalovat a zkoušet moji aplikaci v mém mobilu. Docílil jsem k tomu tak že v nastavení v informacích o zařízení jsem 6x rychle za sebou po klikal na číslo sestavení

## Práce s Activity

### Activity

Activity je to co uživatel vidí. Je tvořena dvěma soubory. Třídou, která vykonává úkony a provádí operace. Layoutem, který se zabývá grafickou stránkou. Activita jako celek komunikuje s uživatelem a vytváří okna. Většinou jsou aktivity prezentovány jako celá obrazovka, ale můžou mít i jiné použití, jako třeba plovoucí obrazovka či přímo vložené do jiné aktivity.

### Vytvoření aktivity

Vytvoření nové aktivity je jednoduchá záležitost. Stačí pravým kliknutím myši na danou záložku vybrat „nový“ > „activity“ a potom si vybereme, zda chceme prázdnou či nějak předem upravenou.

### Přepínaní mezi aktivitami

Toto je jedna z nejdůležitějších akcí ve všech projektech, které se týkají mobilních aplikací. Její syntaxe je velice jednoduchá a rychlá. Nejprve ji musíme inicializovat.

Intent (název) = new Intent(getApplicationContext(), (název třídy aktivity);

A poté spustit.

startActivity(název);

### Posílání proměnné mezi aktivitami

I když toto taktéž není náročná činnost, musíme si u ní dávat pozor na správné indexovaní proměnných. Tato činnost má dvě fáze. Nejprve si v activitě ze které chceme posílat vložit proměnou do něčeho co můžeme nazvat dopis.

(název).putExtra(„jméno“, proměnná);

Ve druhé activitě tento dopis vybalíme. Samozřejmě můžeme posílat jakýkoliv typ proměnných, avšak pak budeme muset přizpůsobit i náš kód. A to změnou v názvu a atributech funkcí .getExtra()

Intent (název) = getIntent();

Proměnná = (název).getExtra(„jméno“);

## JSON parsing

Jazyk JAVA má knihovny určené k parsování JSON souborů a proto jsem si myslel, že to bude jednoduchá a rychlá záležitost. Mýlil jsem se. Musel jsem vyřešit problematiku kdy, a kde bych to měl parsovat, aby se mi to už zobrazilo přesně tam, kde jsem potřeboval a kdy jsem potřeboval. Nakonec jsem zvolil asynchronní funkce. Své JSON soubory jsme měl uložené na internetových stránkách, tak jsem ještě potřeboval vyřešit tuto problematiku.

### AsyncTask

Generická abstraktní třída AsyncTask umí asynchonně spustit jiný kód. Užitečná je zejména v tom, že nemusíte ručně vytvářet nová vlákna a starat se o ně. Využívá se většinou při náročnějších operacích. Využívám tyto funkce.

Funkce onPreExecute() se volá před započetím práce asynchronního úkolu, zde vytvářím progres dialogu, který informuje uživatele o probíhající činnosti. Ve funkci doInBackground probíhá výpočet. A ve funkci onPostExecute() se zpracovává vrácený výsledek. AsyncTask se spouští zavoláním funkce execute() s parametry.

### Přístup k internetu

Aby naše aplikace mohla přistupovat k internetu, a využívat informace z něho získané musíme si toto povolit v souboru AndroidManifest.xml. Příkazem:

<uses-permission android:name=“android.permission.INTERNET“/>

Poté si vytvoříme třídu, která nám zajistí ověření internetové adresy a zajistí nám přijetí dat z JSON souboru v podobně stringu.

### Parsování

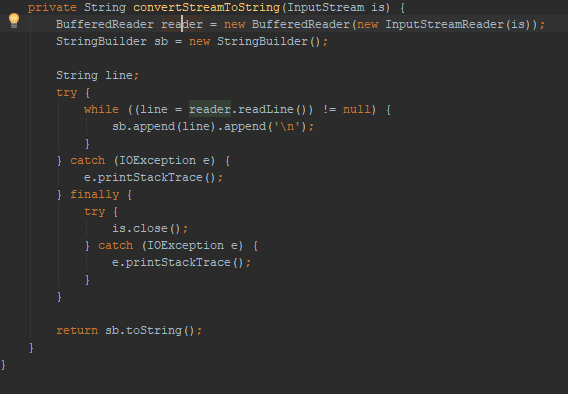
V asynchroní funkci „DoInBackground“ si inicializujeme třídu, která nám zajišťuje komunikování s internetovou adresou. Pokud máme více stránek s JSON soubory můžeme si je posílat do této třídy postupně pomocí switche či jiného přepínače.

Poté si pomocí předdefinovaných funkcí v jazyce JAVA na parsovaní JSON souboru jako jsou JSONArray(); JSONObject(); a opakovače for(), rozdělíme tento string podle toho, jak potřebujeme.

Nakonec si tyto data přepošleme kde potřebujeme ve funkci onPostExecute().

# Výsledky řešení, výstupy, uživatelský manuál

## Ukázka kódu



Na tomto obrázku je zachycena část kódu, která se stará o konvertování již získaného streamu dat z JSON souboru na proměnou string.

* výčet splněných a nesplněných cílů, obrázky (schémata, vzorce apod.) z finálního provedení, prokázání funkčnosti, výsledné parametry výrobku apod.
* podle zaměření a charakteru práce je třeba volit vhodný nadpis pro tuto kapitolu, je samozřejmě možné i rozdělení na více kapitol (např. Uživatelské rozhraní internetové aplikace; Administrace internetové aplikace…)

# Závěr

Cílem tohoto projektu bylo vytvoření funkční mobilní informační aplikace. Určené cíle jako zobrazení dat z JSON souborů, které byly převzaty z internetu a jejich přehledné vypisování bylo splněno.

Do budoucna by bylo možné přidat přihlašovací systém, který by byl propojen s přihlašovacím systémem hry Warframe.

Tuto aplikaci jsem navrhl pro vlastní potřebu abych měl rychlejší a jednoduší nalezení daných informací, když je budu potřebovat.

* povinná část,
* shrnuje výsledky, hodnotí splnění cíle práce, uvádí možnost uplatnění řešení v praxi a nastínění případných dalších budoucích vylepšení
* kapitola se nečísluje (stejné jako úvod)

Seznam použitýCH INFORMAČNÍCH ZDROJů

[1] BOHMAN, Ludvík. Zákon o pojistné smlouvě. Praha: Linde Praha a. s., 2004. 381 s. ISBN80-7201-504-4

[2] DUCHÁČKOVÁ, Eva. Principy pojištění a pojišťovnictví. 3. aktualizované vydání. Praha: Ekopress 2009. 224 s. ISBN 978-80-86929-51-4

[3] KUBALA, Petr. Planetární dvojcata - Věda a technika (Český rozhlas) [online].   
Č. 2000-2008, poslední revize 19. 3. 2008 [cit. 2008-03-20].  
<http://www.rozhlas.cz/veda/vesmir/\_zprava/435849>.

[4] KULDOVÁ, O., FLEISCHMANNOVÁ, E. Metodická příručka k technice administrativy a obchodní korespondence. 1.vyd. Praha: Fortuna 1998. 111 s.   
ISBN 80-7168-574-7. Kapitola 6, Metody nácviku psaní hmatovou metodou,   
s. 28-29.

[5] VLACH, J. JE Temelín a zásobování teplem. Energetika, 2001, roč. 51, č. 3, s. 84 -85. ISSN 0375-8842.

[6] BUTTERFIELD, B.< <https://www.youtube.com/watch?v=dFlPARW5IX8>>,< https://www.youtube.com/watch?v=6ow3L39Wxmg&t>,< https://www.youtube.com/watch?v=rdGpT1pIJlw&t>

[7] PANWAR, A. < <https://www.youtube.com/watch?v=Vcn4OuV4Ixg&t>>

[8] Autor neznámý < <https://www.tutorialspoint.com/android/android_json_parser.htm>>

[9] Autor neznámý < <https://cs.wikipedia.org/wiki/Hlavn%C3%AD_strana>>

[10] Autor neznámý < https://stackoverflow.com/>

Seznam příloh

č. 1 Titulní list

č. 2 Čestné prohlášení

č. 3 Poděkování

Nepovinná část – pokud nemáte žádné přílohy ke své práci, tuto část odstraňte!

* Přílohy se zařazují na konec práce.
* Jsou to texty, obrázky, grafy, tabulky, které by přímo v textu byly zbytečně detailní, ale mají být po ruce k dokreslení východisek i výsledku řešení.
* Jsou číslovány a v textu se na ně může odkazovat.
* Před první přílohu se umisťuje seznam příloh.
* Každá příloha je označena číslem - např. Tabulka č.. 1, Schéma č. 2, Obrázek č. 3.
* Každá tabulka by měla mít i vlastní název, který stručně vystihuje její obsah.
* (Tabulka č. 1 Zakázky stavebních prací v roce 2009-2010).
* Pokud je z tabulky vytvořen graf, umístíme jej na stejné stránce jako tabulku.

**Příloha č. 1: Titulní list**