

Bakalářská práce



České  
vysoké  
učení technické  
v Praze

**F3**

Fakulta elektrotechnická  
Katedra počítačů

## Slovníkový modul Vokabuláře webového pro Android

**Lukáš Toman**

Vedoucí: Ing. Ivo Malý, Ph.D.

Obor: Softwarové systémy

Studijní program: Otevřená informatika

Květen 2017

České vysoké učení technické v Praze  
Fakulta elektrotechnická  
Katedra počítačů

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Toman Lukáš

Studijní program: Otevřená informatika  
Obor: Softwarové systémy

Název tématu: Slovníkový modul Vokabuláře webového pro Android

Pokyny pro vypracování:

Proveďte analýzu slovníkového modulu Vokabuláře webového a navrhnete mobilní verzi slovníkového modulu jako samostatnou aplikaci. Návrh otestujte formou prototypů v uživatelských testech. Na základě prototypů implementujte mobilní aplikaci pro platformu Android. Při tvorbě se zaměřte jak na práci v online režimu tak i na možnosti zpřístupnění dat bez připojení k Internetu. Výslednou aplikaci otestujte pomocí vhodných uživatelských testů.

Seznam odborné literatury:

- [1] M. Jones, G. Marsden, Mobile Interaction Design, Wiley, 2006
- [2] B. Phillips, C. Stewart, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch Guides, 2015.
- [3] Vokabulář webový, <http://vokabular.ujc.cas.cz/>

Vedoucí: Ing. Ivo Malý, Ph.D.

Platnost zadání do konce letního semestru 2017/2018

prof. Dr. Michal Pěchouček, MSc.  
vedoucí katedry



prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 13.2.2017

## Poděkování

V první řadě bych chtěl poděkovat vedoucímu mé práce panu Ivovi Malému za trpělivost, pomoc a rady. V další řadě mé rodině za podporu při studiu.

## Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškerou použitou literaturu.

V Praze, 26. května 2017

## Abstrakt

Cílem této bakalářské práce je vytvoření mobilní aplikace slovníkového modulu Vokabuláře webového pro platformu android. Základem práce je analýza uživatelského prostředí již fungující webové aplikace Vokabuláře webového, vytvoření prototypu a následná implementace. Výsledná mobilní aplikace bude, po vzoru webové aplikace, umožňovat listování slovníkovými hesly, vyhledávání konkrétního slovníkového hesla a ukládání hesel pro budoucí zobrazení bez využití internetového připojení.

**Klíčová slova:** android, android aplikace, Vokabulář webový

**Vedoucí:** Ing. Ivo Malý, Ph.D.

## Abstract

The Bachelor Thesis deals with creation of mobile application - dictionary module of Vokabulář webový for Android platform. The basis of this paper is to analyse Vokabulář webový's web application's user interface, prototype creation and its implementation. Resulting mobile application will, as of web application, allow functions such as listing in dictionary entries, searching for specific entry and saving dictionary entries for future display without the need of Internet connection.

**Keywords:** android, android application, Vokabulář webový

# Obsah

<b>1 Úvod</b>	<b>1</b>	<b>6 Implementace</b>	<b>19</b>
1.1 Vokabulář webový	1	6.1 MainActivity	19
1.2 Modul Slovníky	1	6.2 ImageFullScreen	19
1.3 Cíl práce	2	6.3 Adapters	19
1.4 Motivace	2	6.3.1 DataBindingAdapter	19
1.5 Obsah práce	2	6.3.2 DictionaryAdapter	19
<b>2 Analýza</b>	<b>3</b>	6.3.3 HintsAdapter	19
2.1 Analýza Vokabuláře webového	3	6.3.4 HistoryAdapter	20
2.2 Analýza modulu slovníky	4	6.3.5 ListingAdapter	20
2.2.1 Informace	4	6.3.6 ViewPagerAdapter	20
2.2.2 Seznam	4	6.4 DataProviders	20
2.2.3 Listování	5	6.4.1 DataProvider	20
2.2.4 Vyhledávání	5	6.4.2 CacheDataProvider	20
2.3 Analýza požadavků	6	6.4.3 ServerDataProvider	20
2.3.1 Funkční požadavky	6	6.5 Fragments	20
2.3.2 Nefunkční požadavky	6	6.5.1 DictionariesFragment	20
2.4 Problémy	6	6.5.2 DictionariesDialogFragment	21
2.4.1 Komunikace se serverem	6	6.5.3 InformationsFragment	21
2.4.2 Zobrazení seznamu hesel a statí	6	6.5.4 ListingFragment	21
2.4.3 Zobrazování obrázků	6	6.5.5 SearchFragment	21
2.4.4 Přecházení na dané heslo/písmeno	6	6.5.6 SearchResultFragment	21
2.4.5 Ukládání hesel se statěmi	7	6.5.7 SearchTabFragment	21
2.5 Android	7	6.6 Holders	22
2.5.1 Material design	8	6.6.1 HeadwordHolder	22
2.5.2 Struktura mobilní aplikace	8	6.7 Listeners	22
2.5.3 Horní panel	8	6.7.1 DictionarySetChangedListener	22
2.5.4 Vlastní obsah	8	6.7.2 EndlessRecyclerViewScrollListener	22
2.5.5 Spodní lišta	9	6.7.3 OnHeadwordListCompleteListener	22
2.5.6 Postraní navigace	9	6.8 Requests	22
2.5.7 Android aplikace	9	6.8.1 Envelope	22
<b>3 Podobné aplikace</b>	<b>10</b>	6.9 Observers	23
3.1 IDOS	10	6.9.1 DictionariesWithCategoriesResponseObserver	23
3.2 Google maps	10	6.9.2 DictionaryEntryFromSearchResponseObserver	23
3.3 Anglicko-český offline slovník	11	6.9.3 DictionaryEntryResponseObserver	23
3.4 Slovník	12	6.9.4 DictionaryImageResponseObserver	23
<b>4 Návrh uživatelského rozhraní</b>	<b>13</b>	6.9.5 HeadwordListResponseObserver	23
4.1 Informace	13	6.9.6 HeadwordRowEntryResponseObserver	23
4.2 Listování	13	6.9.7 TypeheadHeadwordsObserver	23
4.3 Vyhledávání	15		
4.4 Slovníky	15		
<b>5 Testování prototypů</b>	<b>17</b>		
5.1 Zadání	17		
5.2 Výsledek testování	17		

6.10 Models .....	23
<b>7 Závěr</b>	<b>25</b>
<b>A Literatura</b>	<b>26</b>
<b>B Obsah CD</b>	<b>27</b>

## Obrázky

2.1 Ukázka stránky s informacemi . . .	4
2.2 Ukázka stránky se seznamem slovníků . . . . .	4
2.3 Ukázka stránky s listováním . . . . .	5
2.4 Ukázka stránky s vyhledáváním slova “pes” . . . . .	5
2.5 OS Android[3] . . . . .	7
2.6 Struktura app baru[3] . . . . .	8
2.7 Spodní lišta[3] . . . . .	9
3.1 Ukázka aplikace “IDOS” . . . . .	10
3.2 Ukázka aplikace “Google Maps”	11
3.3 Ukázka aplikace “Anglicko-český offline slovník” . . . . .	11
3.4 Ukázka aplikace “Slovník” . . . . .	12
4.1 Návrh obrazovky Informace . . . . .	13
4.2 Návrh listování v podobě seznamu a karet . . . . .	14
4.3 Návrh listování v podobě rozbalovacích karet . . . . .	14
4.4 Návrh listování v podobě rozbalovacích karet . . . . .	15
4.5 Návrh obrazovky s výsledky fulltextového hledání slova “pes” . .	16

## Tabulky

5.1 Výsledky podle uživatele 1 . . . . .	18
5.2 Výsledky podle uživatele 2 . . . . .	18
5.3 Výsledky podle uživatele 3 . . . . .	18

# Kapitola 1

## Úvod

### 1.1 Vokabulář webový

Vokabulář webový je webová aplikace, která zpřístupňuje textové, obrazové a zvukové zdroje k poznání historické češtiny. Tvůrcem a provozovatelem je Ústav pro jazyk český AV ČR za přispění fakulty elektrotechnické ČVUT v Praze. Jsou zde umístěny různorodé zdroje a skládá se z několika modulů. Jeho databáze je průběžně doplňována o nové zdigitalizované zdroje.

### 1.2 Modul Slovníky

V oddílu Slovníky vokabuláře webového jsou poskytovány zájemcům o historickou češtinu informace o její slovní zásobě. Tvoří jej různorodé lexikální zdroje, které umožňují jednotné vyhledávání a listování, tj. procházení slovníkovými zdroji „po listech“.

Základ oddílu tvoří tato novodobá lexikografická díla pojednávající zejména o staročeské slovní zásobě: Elektronický slovník staré češtiny (ESSČ), Malý staročeský slovník (MSS), pracovní heslář k lístkové kartotéce Staročeského slovníku (HesStčS), Slovník staročeský Jana Gebauera (GbSlov), Staročeský slovník (StčS), Slovníček staré češtiny Františka Šimka (ŠimekSlov) a Index Slovníku staročeských osobních jmen Jana Svobody (IndexSvob).

Jsou zde však též dostupné elektronické verze historických slovníků a podobných lexikografických příruček z období 16. až 19. století. Jedná se např. o Česko-německý slovník Jana Václava Pohla, jehož elektronickou edici vytvořil se svými kolegy prof. Tilman Berger z univerzity v Tubinkách; dále Dodavky ke slovníku Josefa Jungmanna od F. L. Čelakovského; Slovník český Jana Františka Josefa Ryvolvy a Deutsch-böhmisches Wörterbuch Josefa Dobrovského, Thesaurus Linguae Bohemicae Václava Jana Rosy a další slovníky, jejichž elektronické verze vznikají v oddělení vývoje jazyka ÚJČ a které primárně poskytují transliterovaný text, doplněný v některých případech o transkripci české jazykové části.

Formou digitalizovaných obrazů doplněných o metainformace (soupis hesel a podheslí) je zveřejněn Slovník česko-německý Josefa Jungmanna. Jedná se o fotokopie archivního exempláře, v němž sám autor označoval opravy



a doplňky textu pro zamýšlené druhé vydání slovníku. Digitální kopii lze prohlížet dvěma způsoby: pomocí listování a vyhledávání v heslových slovech a podheslích, a to podle transliterované i transkribované podoby.[1]

### ■ 1.3 Cíl práce

Cílem této práce je vytvořit fungující a otestovanou mobilní aplikaci pro platformu Android, které bude poskytovat přehledné uživatelské rozhraní a v co největší míře splní požadavky na funkcionalitu celé aplikace.

### ■ 1.4 Motivace

Hlavním důvodem pro vytvoření této aplikace byla snaha o navržení takového uživatelského rozhraní, ve kterém budou moci uživatelé mobilních zařízení pracovat stejně jako s plnohodnotnou webovou aplikací, které ale bude uzpůsobeno limitům vztahujícím se k mobilním zařízením.

### ■ 1.5 Obsah práce

Práce se zabývá analýzou, návrhem, implementací a testováním mobilní aplikace Vokabuláře webového pro operační systém android. Kapitola 2 se zaměřuje na analýzu celého projektu, analýzou vokabuláře webového se vytvoří požadavky na aplikaci pro mobilní zařízení a zjistí se problémy, které budou potřeba vyřešit. Kapitola 3 se věnuje již hotovým aplikacím na Android, které řeší problémy nalezené při analýze, nebo které se funkcionalitou podobají vyvíjené aplikaci. Kapitola 4 se zabývá návrhem uživatelského rozhraní a vytvořením prototypů. Kapitola 5 se věnuje testování prototypů. V kapitole 6 dojde na samotnou implementaci aplikace s popisem jednotlivých funkcí použitých při implementaci.

## Kapitola 2

### Analýza

#### 2.1 Analýza Vokabuláře webového

Celý systém Vokabuláře webového se skládá z několika modulů:

**Slovníky** Slouží k vyhledávání informací o slovní zásobě historické češtiny.[1]

**Edice** Jsou zde prezentovány prameny zaznamenané starší češtinou jako souvislý text společně s textově-kritickým komentářem.[1]

**Korpusy** Slouží k prohledávání starých českých textů.[1]

**Mluvnice** Prezentuje digitalizované verze mluvnice a obdobných příruček z 16. až počátku 20. století. Mluvnice slouží nejen ke studiu dobového stavu a proměn českého jazykového systému, ale též k bádání o vývoji českého mluvnictví.[1]

**Odborná literatura** Jsou zde zveřejněny digitalizované verze odborných textů, které se věnují problematice historické češtiny.[1]

**Bibliografie** Nabízí vyhledávání v bibliografických záznamech odborné literatury k problematice staršího českého jazyka a literatury.[1]

**Kartotéky** Zpřístupňují digitalizovanou podobu dvou kartoték Jana Gebauera: kartotéky excerpce ze staročeské literatury a kartotéky pramenů k této excerpce.[1]

**Pomůcky** Nabízí softwarové nástroje a pomůcky pro práci s digitalizovanými zdroji. Tyto nástroje si mohou zájemci zdarma stáhnout a používat, a to včetně zdrojových kódů.[1]

Serverová část Vokabuláře webového funguje na protokolu SOAP. SOAP je protokol vytvořený pro výměnu strukturovaných informací. Využívá XML technologie k definování rozšiřitelného rozhraní, které určuje strukturu zpráv, které mohou být posílány prostřednictvím jednoho z transportních protokolů(html, smtp, tcp, udp a další) .[2] Systém autorizace a autentikace Vokabuláře webového umožňuje spravovat jaké zdroje má daný uživatel právo zobrazovat. Pro nepřihlášené uživatel jsou dostupné jen základní zdroje.

## 2.2 Analýza modulu slovníky

### 2.2.1 Informace

Úvodní obrazovka slovníkového modulu zobrazuje základní informace o něm. Jsou zde odkazy na vybrané slovníky a na další funkce modulu jako je listování a vyhledávání (Obrázek [2.1]).

Obrázek 2.1: Ukázka stránky s informacemi

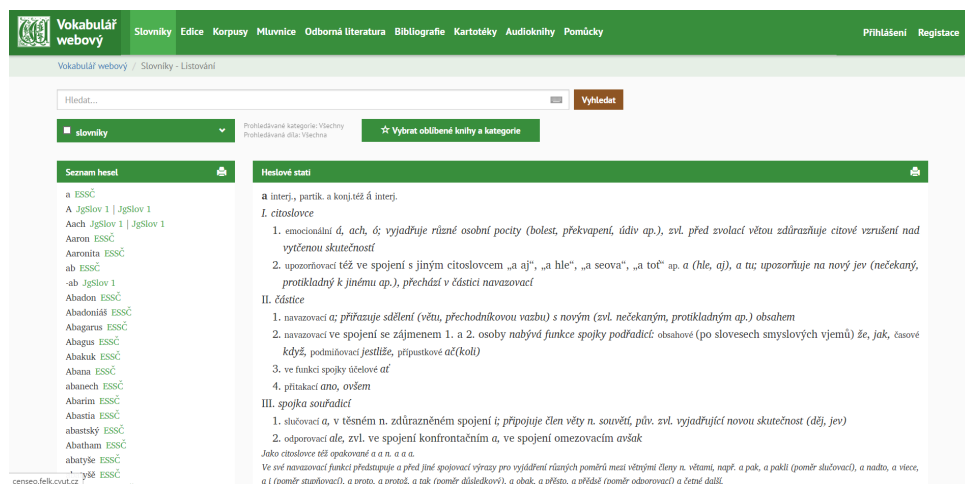
### 2.2.2 Seznam

V oddílu slovníky jsou uživateli zobrazena jednotlivá díla, která jsou k dispozici. Uživatel má možnost je zde řadit, vyhledávat konkrétní slovník a zobrazovat si informace o daném slovníku (Obrázek [2.2]).

Obrázek 2.2: Ukázka stránky se seznamem slovníků

### 2.2.3 Listování

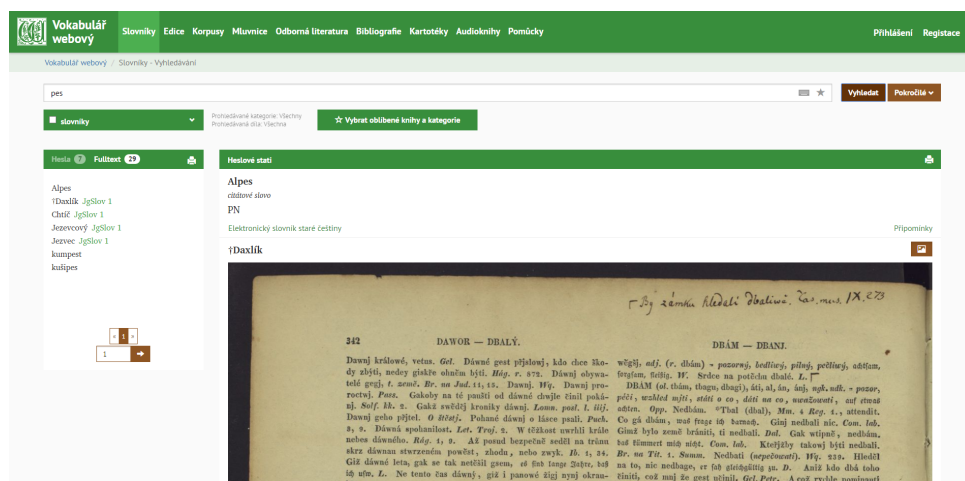
Při listování si uživatel vybere ze seznamu slovníky, kterými chce listovat a poté mu systém vypíše rozstránkový seznam hesel a heslových statí. Uživatel může také vyhledávat konkrétní heslo. Po zadání hesla systém zobrazí stránku, na které se heslo nachází (Obrázek [2.3]).



Obrázek 2.3: Ukázka stránky s listováním

### 2.2.4 Vyhledávání

U vyhledávání uživatel zadá slovo, které chce najít a systém mu poté vypíše všechna hesla nebo statě, která dané slovo obsahují. Funguje zde i pokročilé vyhledávání, kde uživatel může upřesnit podmínky pro oblasti, ve kterých se hledá (Obrázek [2.4]).



Obrázek 2.4: Ukázka stránky s vyhledáváním slova “pes”

## ■ 2.3 Analýza požadavků

### ■ 2.3.1 Funkční požadavky

- 01 Aplikace bude zobrazovat informace o modulu slovníky
- 02 Aplikace bude umožňovat listovat hesly v obou směrech
- 03 Aplikace bude umožňovat při listování přejít na určité písmeno/heslo
- 04 Aplikace bude umožňovat vyhledávání konkrétního hesla
- 05 Aplikace bude umožňovat vyhledávání výrazu ve statích
- 06 Aplikace bude umožňovat ukládání vybraných hesel se statěmi
- 07 Aplikace bude umožňovat vybírání slovníků, ze kterých se mají hesla zobrazit

### ■ 2.3.2 Nefunkční požadavky

- Minimální verze Android API bude 15
- Aplikaci bude možné používat i bez připojení k internetu

## ■ 2.4 Problémy

### ■ 2.4.1 Komunikace se serverem

Rozhraní serveru funguje na SOAP protokolu. OS Android nemá nativní podporu tohoto protokolu.

### ■ 2.4.2 Zobrazení seznamu hesel a statí

Seznam hesel a zobrazení statí je ve Vokabuláři oddělené (Obrázek [2.3]). Toto na malých obrazovkách mobilních zařízení není možné.

### ■ 2.4.3 Zobrazování obrázků

Jednotlivé statě mohou být v podobě obrázku, který na mobilních zařízeních může být nečitelný, proto se bude muset dát přiblížovat.

### ■ 2.4.4 Přecházení na dané heslo/písmeno

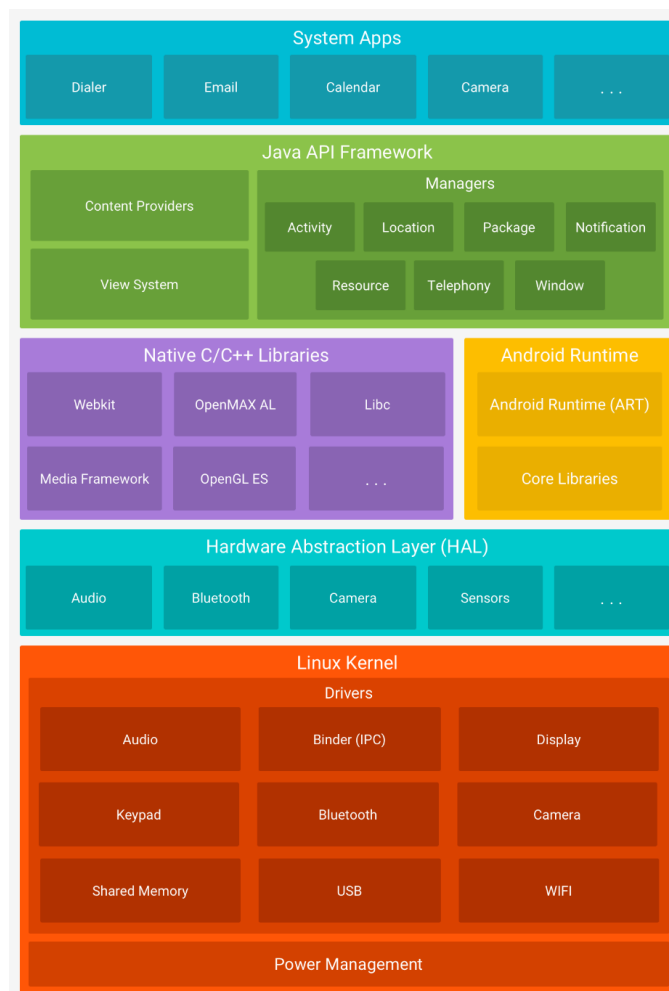
Při listování bude aplikace umožňovat zadání písmena/hesla, na které poté seznam hesel odroluje a načítat i hesla, která jsou “nad” tímto heslem.

### ■ 2.4.5 Ukládání hesel se statěmi

Uživatel si bude moci uložit dané heslo se statí pro budoucí zobrazení bez přístupu k internetu.

## ■ 2.5 Android

[3] Android je open-source operační systém založený na Linuxovém jádru vyvíjený společností Google. Je dostupný převážně pro mobilní zařízení s dotykovou obrazovkou, hlavně pro mobilní telefony a tablety, ale v poslední době i pro chytré hodinky a televize. Na trhu s mobilními telefony zaujímá první místo s přibližně 85% podílem. Celý operační systém se skládá z několika vrstev. 2.5



**Obrázek 2.5:** OS Android[3]

### ■ 2.5.1 Material design

Material design je vizuální jazyk vytvořený společností Google, který je vytvořen za účelem sjednocení uživatelského zážitku napříč různými platformami, velikostmi obrazovek apod.

### ■ 2.5.2 Struktura mobilní aplikace

Obrazovka mobilní aplikace podle Material designu je rozložena na několik oddílů:

- Horní panel, tzv. App bar
- Vlastní obsah
- Spodní lišta
- Levá a pravá navigace

### ■ 2.5.3 Horní panel

Horní panel, neboli app bar nebo action bar, je speciální panel sloužící k navigaci, hledání a akcím. Základ app baru tvoří zleva: ikona navigace, může to být prvek pro otevírání levé postraní navigace, šipka pro navigaci „vzhůru“ hierarchií aplikace nebo může být zcela vynechána. V této aplikaci bude sloužit pro otevírání levé postraní navigace, která bude sloužit jako menu aplikace. Dále je zde titulek, který reflektuje aktuální stránku, může to být název aplikace, stránky nebo filtr stránky. V pravé části jsou ikony sekundárních akcí a ikona pro otevření menu, které obsahuje ostatní sekundární akce. V aplikaci se bude přes toto menu zobrazovat dialogové okno s výběrem slovníků.



Obrázek 2.6: Struktura app baru[3]

### ■ 2.5.4 Vlastní obsah

V této části se nachází vlastní obsah. Obsah je tvořen různými komponenty, např:

**Text view** Základní komponenta pro zobrazování textu.

**Text field** Slouží pro získávání textového vstupu od uživatele.

**Buttons** Tlačítka slouží ke spuštění dané akce, když na ně uživatel klikne.

**Cards** Karty se používají pro zobrazení obsahu, který se skládá z více prvků (text, obrázek apod.). Hodí se také pro prvky, které mají rozdílnou velikost apod. Karty mají konstantní šířku, ale liší se ve výšce. Mohou být také roztahovací, pro zobrazení obsahu bez potřeby scrollování.

**Lists** Seznamy se používají pro zobrazení stejnorodého nebo podobného obsahu. Sestávají se z jednotlivých položek, které mají stejnou šířku i výšku.

### ■ 2.5.5 Spodní lišta

Spodní lišta je určena pro tlačítka navigace. V aplikaci bude sloužit jen pro funkce systému.



Obrázek 2.7: Spodní lišta[3]

### ■ 2.5.6 Postraní navigace

Panely postraní navigace mohou být pořád viditelné po stránkách nebo se mohou skrývat a zobrazovat po stisku tlačítka navigace. Panel na levé straně slouží primárně k navigaci v rámci aplikace a panel na pravé straně je používán pro sekundární obsah k hlavnímu obsahu zobrazeném na stránce. V aplikaci bude použitý pravý postraní panel pro navigaci v rámci aplikace.

### ■ 2.5.7 Android aplikace

Nativní Android aplikace jsou psané v programovacím jazyce JAVA poté jsou pomocí Android SDK s dalšími daty a zdrojovými soubory zkompileovány do takzvaného Android balíčku, neboli APK. Pak jakékoliv zařízení s Android OS může danou aplikaci nainstalovat. Základem Android aplikací jsou čtyři základní komponenty:

1. Activity – stará se o uživatelské rozhraní a o uživatelské interakce s aplikací
2. Service – stará se o procesy na pozadí
3. Broadcast Receiver – stará se o komunikaci mezi OS Android a aplikacemi
4. Content Provider – stará se o data a o komunikaci s databází

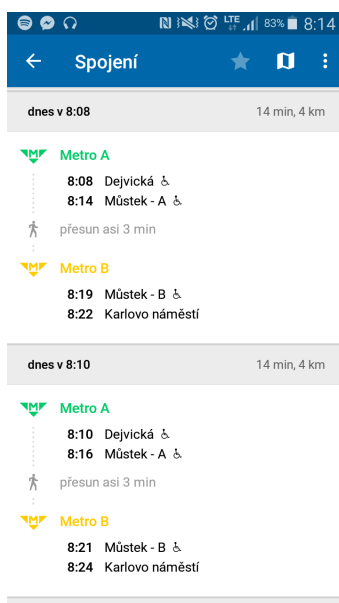


## Kapitola 3

### Podobné aplikace

#### 3.1 IDOS

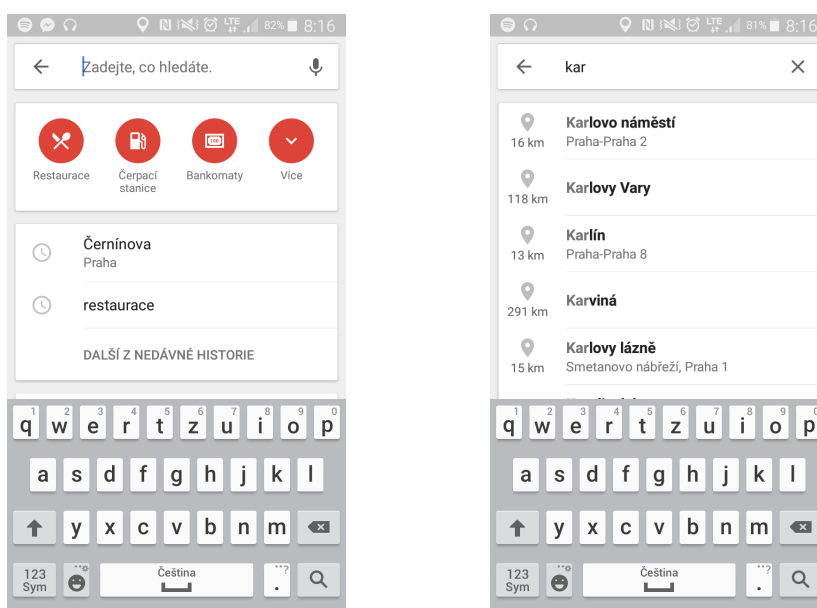
Aplikace pro vyhledávání vlakových, autobusových a mhd spojů.[4] Nalezené spoje se zobrazují v seznamu, který používá metodu nekonečného scrollování pro načítání dalších položek, dále používá metodu „pull to refresh“ k zobrazení předchozích položek. (Obrázek[3.1])



Obrázek 3.1: Ukázka aplikace “IDOS”

#### 3.2 Google maps

Nově navržená aplikace Mapy Google pro telefony a tablety Android usnadňuje a urychluje navigaci na cestách.[5] Při hledání konkrétních míst nabízí uživateli místa, které v poslední době hledal a místa, která začínají zadávaným výrazem. (Obrázky[3.2])



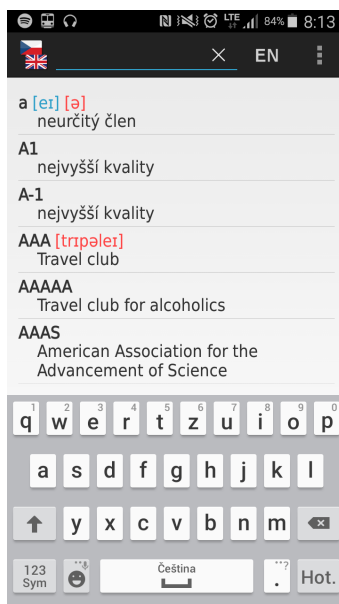
(a) : Ukázka historie

(b) : Ukázka doplnění slova

Obrázek 3.2: Ukázka aplikace “Google Maps”

### 3.3 Anglicko-český offline slovník

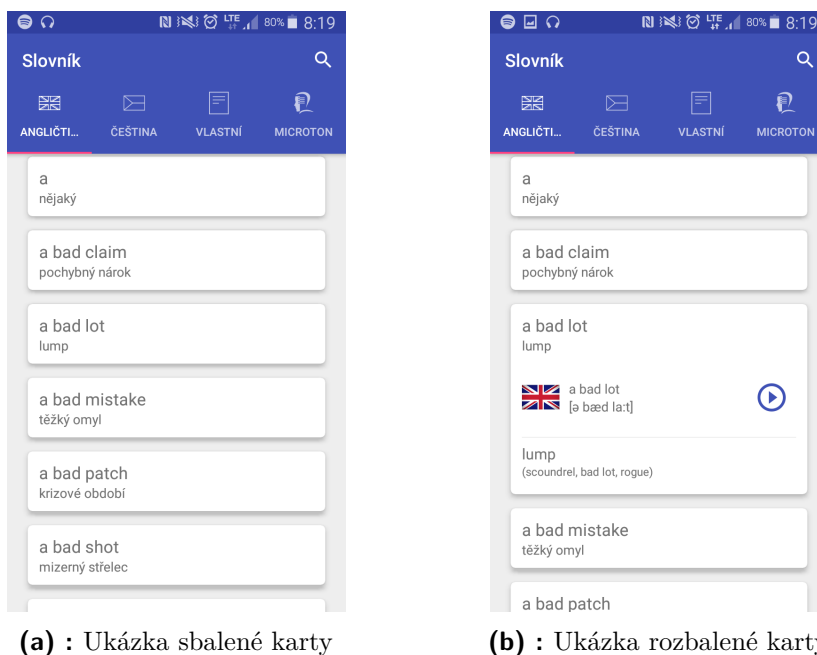
Anglicko - český a česko - anglický offline slovník s výslovností.[6] Používá pro zobrazení hesel seznam, v horní části je umístěno textové pole pro zadávání hledaného výrazu, po zadání výrazu odroluje na hledané heslo. (Obrázek[3.3])



Obrázek 3.3: Ukázka aplikace “Anglicko-český offline slovník”

## 3.4 Slovník

Velmi obsažná slovní databáze, více než 130 tisíc překladových párů v anglicko-českém a téměř stejný počet v česko-anglickém směru. Rozsah srovnatelný s několikasetstránkovým tištěným slovníkem.[7] Hesla jsou zde zobrazeny v rozbalovacích kartách. (Obrázek[3.4])



**Obrázek 3.4:** Ukázka aplikace “Slovník”

## Kapitola 4

### Návrh uživatelského rozhraní

Podle struktury Vokabuláře webového byly navrženy následující obrazovky.

#### 4.1 Informace

Základní obrazovka aplikace, zobrazuje se jako první po spuštění. Předkládá uživateli informace o slovníkovém modulu (Obrázek[4.1]).

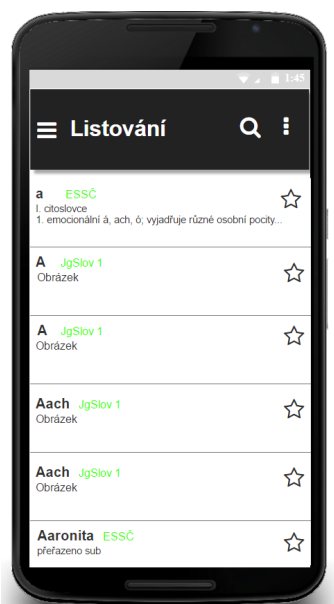


Obrázek 4.1: Návrh obrazovky Informace

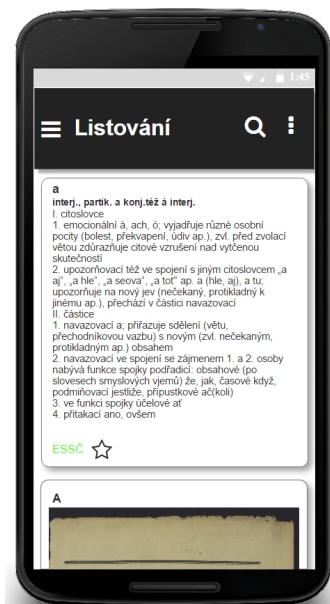
#### 4.2 Listování

Obrazovka listování je hlavní částí aplikace. Používá se pro listování hesly a pro zobrazování statí. To lze provést pomocí dvou variant a to v podobě karet nebo v podobě seznamu(Obrázky [4.2]). Karty mohou být “klasické” jako na obrázku[4.2b] nebo roztahovací (Obrázek [4.3]). Výhody seznamu je jeho v rychlosti scrollování a v přehlednosti, nevýhodou je, že stať se zobrazuje na nové obrazovce. Karty mají tu nevýhodu, že při dlouhém obsahu

je potřebné zdlouhavě scrollovat, výhodou je, že lze mít zobrazeno více statí najednou. Rozbalovací karty kombinují výhody karet a seznamu, ale mohou být nepřehledné.

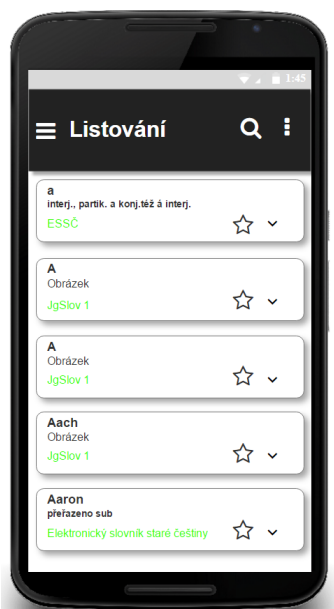


(a) : Seznam

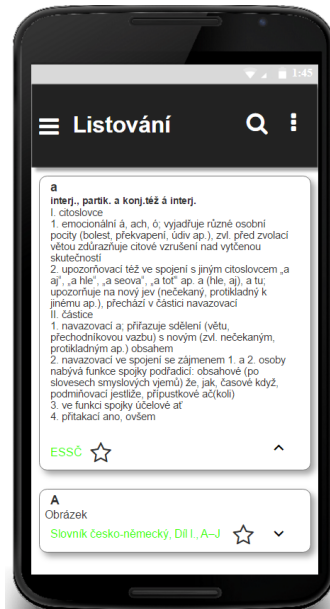


(b) : Karty

Obrázek 4.2: Návrh listování v podobě seznamu a karet



(a) : Zabalené karty

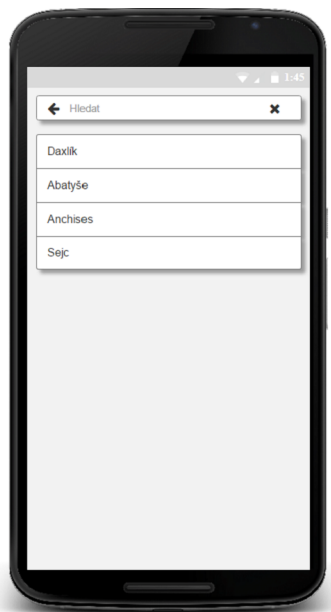


(b) : Rozbalená karta

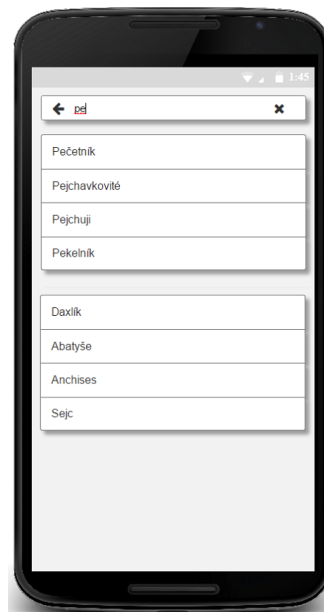
Obrázek 4.3: Návrh listování v podobě rozbalovacích karet

## 4.3 Vyhledávání

Vyhledávání je rozděleno do dvou obrazovek. První obrazovka zobrazuje textové pole, kam uživatel napíše hledaný výraz a aplikace mu nabídne seznam slov, kterým výraz odpovídá. Slovo lze také vybrat ze seznamu výrazů, které uživatel v poslední době vyhledával (Obrázky[4.4]). Po napsání, vybrání, příslušného výrazu se objeví obrazovka s výpisem výsledků. Tato obrazovka bude obsahovat dva seznamy, jeden pro hledání v heslech a jeden pro fulltextové hledání, které budou mít stejnou funkcionalitu jako Listování (4.2). Mezi seznamy se přepíná pomocí záložek (Obrázek[4.5]).



(a) : Seznam hesel z historie



(b) : Seznam hesel, která začínají zadávaným výrazem

**Obrázek 4.4:** Návrh listování v podobě rozbalovacích karet

## 4.4 Slovníky

Slovníky budou zobrazeny v podobě seznamu, který bude obdobný jako v případě Listování (4.2).



Obrázek 4.5: Návrh obrazovky s výsledky fulltextového hledání slova “pes”

## Kapitola 5

### Testování prototypů

#### 5.1 Zadání

Pro účely testování byly vytvořeny tři prototypy aplikace, které odrážejí návrhy. Prototypy se liší stylem listování - v jednom je listování vytvořeno pomocí seznamu, ve druhém pomocí karet a v posledním pomocí rozbalovacích karet. Všichni účastníci testování měli za úkoly splnit dvě úlohy:

1. Z úvodní obrazovky přejděte k listování, listujte hesly a zobrazte si heslové stati pro heslo “a” ze slovníku “ESSČ” a obrázek s heslem “A” ze slovníku “JgSlov 1”.
2. Z úvodní obrazovky přejděte k vyhledávání, vyhledejte heslo “pes” a zobrazte si heslové stati pro heslo “Alpes” ze slovníku “ESSČ”, obrázek pro heslo “Daxlík” ze slovníku “Slovník česko-německý, díl I., A-J” a stat pro heslo “nádvorní” ze slovníku “Staročeský slovník” ve fulltextovém vyhledávání.

Po projití úkolů měli účastníci seřadit jednotlivé prototypy podle toho, jak se hodí na daný úkol. Všichni účastníci testování používají běžně mobilní zařízení a orientují se v používání elektronických slovníků (studenti humanitních studií).

#### 5.2 Výsledek testování

Uživatelé zhodnotili test následovně:

- První uživatel pro listování i vyhledávání určil vhodné rozbalovací karty nebo seznam (viz. Tabulka [5.1])
- Druhý uživatel označil pro listování vhodné karty, je důležité hned vidět obsah stati a pro vyhledávání jsou vhodnější rozbalovací karty (viz. Tabulka [5.2])
- Třetí uživatel určil, že pro listování je vhodné použít seznam a pro vyhledávání rozbalovací karty (viz. Tabulka [5.3])



	seznam	karty	rozbalovací karty
listování	1	3	2
vyhledávání	2	3	1

**Tabulka 5.1:** Výsledky podle uživatele 1

	seznam	karty	rozbalovací karty
listování	3	1	2
vyhledávání	3	2	1

**Tabulka 5.2:** Výsledky podle uživatele 2

	seznam	karty	rozbalovací karty
listování	1	3	2
vyhledávání	3	2	1

**Tabulka 5.3:** Výsledky podle uživatele 3

Z výsledku testování vyplývá, že se nejlépe na listování i na vyhledávání hodí rozbalovací karty.

# Kapitola 6

## Implementace

Celá aplikace funguje ve dvou aktivitách.

### 6.1 MainActivity

Primární aktivita, ve které se zobrazují jednotlivé fragmenty daných funkcí (listování, vyhledávání atd.). Definuje základní rozhraní aplikace a stará se o veškerou navigaci v rámci aplikace.

### 6.2 ImageFullScreen

Fullscreen aktivita, která zobrazuje stať v případě, že se jedná o obrázek. Je zde použita knihovna PhotoView[8], díky které se obrázek může přibližovat a oddalovat.

### 6.3 Adapters

V tomto balíčku se nacházejí třídy, které spojují View s daty. Adaptery umožňují přístup k jednotlivým data položkám. Adaptery jsou také odpovědné za vytvoření patřičného View pro každou položku v daných datech.

#### 6.3.1 DataBindingAdapter

Tato třída umožňuje navázání zdrojového bitmapového obrázku s patřičným View. V daném view se obrázek zobrazí.

#### 6.3.2 DictionaryAdapter

Třída, dědicí od třídy ArrayAdapter, poskytuje data pro zobrazení slovníků v seznamu.

#### 6.3.3 HintsAdapter

Třída, která poskytuje data pro zobrazení nápovědy, doplnění slov ke konkrétnímu zadávanému výrazu.

### ■ 6.3.4 HistoryAdapter

Tato třída poskytuje historii hledaných výrazu pro zobrazení.

### ■ 6.3.5 ListingAdapter

Třída poskytující hesla a statě pro zobrazení v rozbalovacích kartách. Zároveň i rozbalování a sbalování konkrétních karet sama řídí. Dále se v ní přiděluje každému obrázku posluchač, který po kliknutí daný obrázek zobrazí v nové aktivitě.

### ■ 6.3.6 ViewPagerAdapter

Třída poskytující fragmenty pro tab view.

## ■ 6.4 DataProviders

V tomto balíčku se nacházejí třídy, které zprostředkovávají data. Data mohou být uložena na serveru nebo v cache paměti.

### ■ 6.4.1 DataProvider

Základní třída fungující jako singleton, která poskytuje všechna data - seznamy hesel pro listování a heslové a fulltextové vyhledávání, seznam slovníků. Při potřebě načtení dalších dat definuje metody, které načítají data ze serveru nebo z cache paměti. Obsahuje také metody pro ukládání dat do cache paměti.

### ■ 6.4.2 CacheDataProvider

Třída poskytující data uložena v chache paměti. Pro cache je vyhrazena paměť 1/10 celkové paměti přidělené aplikaci. Využívá třídu LruCache, která funguje na principu fronty. Pokaždé když se přistupuje k určité hodnotě uložené v cache paměti, je tato hodnota posunuta na začátek fronty. Pokud se přidává další položka při plné cache paměti, tak je poslední položka ve frontě smazána.

### ■ 6.4.3 ServerDataProvider

Tato třída je odpovědná za vytvoření správné žádosti o data na server a zároveň přiděluje jednotlivým zprávám posluchače, které přijmou a zpracují odpověď ze serveru.

## ■ 6.5 Fragments

### ■ 6.5.1 DictionariesFragment

Tento fragment zobrazuje seznam slovníků.

### ■ 6.5.2 DictionariesDialogFragment

Fragment, který vytváří dialogové okno pro výběr slovníků. Po zaškrtnutí slovníků uživatelem fragment uloží vybrané slovníky v DataProvideru.

### ■ 6.5.3 InformationsFragment

Fragment zobrazující informace o modulu. Pomocí asynchronní metody stáhne ze serveru html stránku s informacemi a poté ji rozebere, upraví a zobrazí jen data s informacemi. K tomu používá knihovnu Jsoup[9]. V tomto případě je práce s touto knihovnou následující: nejprve se načte dokument z dané url, poté se v dokumentu najde první element se třídou “module-content”, ve kterém se všechny odkazy změní na tučné písmo (Ukázka kódu [6.1]).

#### Zdrojový kód 6.1: Ukázka práce s Jsoup knihovnou

---

```
Document doc = Jsoup.connect(url).get();
Element content = doc.getElementsByClass("module-content").first();
Elements links = content.getElementsByTag("a");
for (Element link : links) {
    Element newElement = new Element("b");
    newElement.text(link.text());
    link.replaceWith(newElement);
}
```

---

### ■ 6.5.4 ListingFragment

Tento fragment se stará o načítání stránek s hesly a zobrazuje karty s těmito hesly.

### ■ 6.5.5 SearchFragment

Fragment, který poskytuje možnost zadávání hledaného výrazu. Zobrazuje poslední hledaná hesla a doplňuje slova při psaní výrazu.

### ■ 6.5.6 SearchResultFragment

Fragment starající se o tab layout - záložky hledání hesla a fulltextu

### ■ 6.5.7 SearchTabFragment

Fragment, který zobrazuje výsledky hledání, stará se o načítání dalších stránek s hledanými výrazy.

## 6.6 Holders

### 6.6.1 HeadworldHolder

Třída používající se při vázání View a ViewModelu.

## 6.7 Listeners

### 6.7.1 DictionarySetChangedListener

Rozhraní definující posluchače, který reaguje v případě změny vybraných slovníků.

### 6.7.2 EndlessRecyclerViewScrollListener

Třída, díky které je realizována funkce nekončeného rolování. Při rolování pokud zbývá méně než pět prvků do konce seznamu zavolá metodu, která načte další položky.

### 6.7.3 OnHeadwordListCompleteListener

Rozhraní definující posluchače, který reaguje v případě příjetí nového seznamu hesel.

## 6.8 Requests

### 6.8.1 Envelope

Třída, která slouží jako dotaz na server. Podle argumentů konstruktoru vytvoří odpovídajících XML dotaz (Příklad dotazu na seznam hesel[6.2]).

---

#### Zdrojový kód 6.2: Ukázka XML dotazu

---

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:tem="http://tempuri.org/">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:GetHeadwordList>
      <tem:start>0</tem:start>
      <tem:count>20</tem:count>
      <tem:bookType>Dictionary</tem:bookType>
    </tem:GetHeadwordList>
  </soapenv:Body>
</soapenv:Envelope>
```

---

## 6.9 Observers

### 6.9.1 DictionariesWithCategoriesResponseObserver

Třída, která čeká na odpověď serveru se seznamem slovníků. Poté tyto slovníky uloží na DataService.

### 6.9.2 DictionaryEntryFromSearchResponseObserver

Třída, která přijímá stať, ve které je pomocí HTML zvýrazněn hledaný výraz při fulltextovém hledání. Tuto stať rozebere a přidá výraz obarví pomocí HTML elementu font na červeno. Poté uloží stať do cache paměti.

### 6.9.3 DictionaryEntryResponseObserver

Třída, která přijímá stať, kterou uloží do cache paměti.

### 6.9.4 DictionaryImageResponseObserver

Třída, která přijímá stať v podobě obrázku. Obrázek je v podobě zakodovaných bytů, proto se zavolá asynchronní metoda, která byty dekoduje a vytvoří z nich bitmapu.

### 6.9.5 HeadwordListResponseObserver

Třída, která přijímá seznam hesel. Pro každé heslo vytvoří novou instanci třídy Headword a uloží ji do DataService.

### 6.9.6 HeadwordRowEntryResponseObserver

Třída, přijímající číslo hledaného hesla. Podle čísla nastaví EndlessRecyclerViewScrollListener.

### 6.9.7 TypeheadHeadwordsObserver

Třída, která přijímá seznam hesel, která začínají výrazu zadaném při hledání.

## 6.10 Models

Tento balíček obsahuje třídy, které definují celý model aplikace. Jsou zde třídy, které slouží pro rozebrání XML odpovědi ze serveru na objekty (Příklad třídy, která slouží pro vytvoření seznamu slovníků a hesel [6.3])

**Zdrojový kód 6.3:** Ukázka třídy podle které se vytvoří seznam hesel z XML dokumentu

```
@XMLObject("//Body")
public class HeadwordList {
```

```
@XMLField("//KeyValueOfstringDictionaryContractgreX8G6")
private List<Dictionary> dictionaries;

@XMLField("//HeadwordContract")
private List<HeadwordEntry> headwordEntries;

public List<Dictionary> getDictionaries() {
    return dictionaries;
}

public List<HeadwordEntry> getHeadwordEntries() {
    return headwordEntries;
}
}
```

---



## Kapitola 7

### Závěr

V rámci této bakalářské práce jsem provedl analýzu Vokabuláře webového a jeho modulu slovníky. V rámci analýzy jsem také nastudoval principy tvorby aplikací pro Android. Poté jsem díky těmto analýzám vytvořil návrhy uživatelského rozhraní slovníkového modulu ve třech variantách a na základě těchto návrhů vyrobil fungující prototypy, které jsem nechal otestovat a pak z nich vybral ten, který byl podle uživatelů nejvhodnější. Podle tohoto prototypu jsem poté naimplementoval velkou část aplikace. Bohužel z důvodu nedostatku času se mi nepodařilo implementovat lokální ukládání hesel a statí. K tomu bych se chtěl v brzké době vrátit, aby byla tato aplikace zcela funkční.



# Příloha A

## Literatura

- [1] Vokabulář webový. *Vokabulář webový* [online]. Praha: oddělení vývoje jazyka Ústavu pro jazyk český AV ČR, v. v. i., 2015 [cit. 2017-05-25]. Dostupné z: <http://vokabular.ujc.cas.cz>
- [2] *SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)* [online]. World Wide Web Consortium (W3C) [cit. 2017-05-25]. Dostupné z: <https://www.w3.org/TR/soap12/>
- [3] *Android Developers* [online]. Google [cit. 2017-05-25]. Dostupné z: <https://developer.android.com/>
- [4] *Jízdní řády IDOS* [online]. MAFRA [cit. 2017-05-25]. Dostupné z: <https://play.google.com/store/apps/details?id=cz.mafra.jizdnirady>
- [5] *Mapy* [online]. Google [cit. 2017-05-25]. Dostupné z: <https://play.google.com/store/apps/details?id=com.google.android.apps.maps>
- [6] *Anglicko-český offline slovník* [online]. DIC-o [cit. 2017-05-25]. Dostupné z: [https://play.google.com/store/apps/details?id=com.dic\\_o.dico\\_cze\\_eng](https://play.google.com/store/apps/details?id=com.dic_o.dico_cze_eng)
- [7] *Slovník* [online]. Microton [cit. 2017-05-25]. Dostupné z: <https://play.google.com/store/apps/details?id=cz.microton.adicoencz>
- [8] *PhotoView* [online]. Chris Banes [cit. 2017-05-25]. Dostupné z: <https://github.com/chrisbanes/PhotoView>
- [9] *jsoup: Java HTML Parser* [online]. Jonathan Hedley [cit. 2017-05-25]. Dostupné z: <https://jsoup.org>



## Příloha B

### Obsah CD

Přiložené CD obsahuje tyto adresáře:

**Implementation** Celý naimplementovaný projekt

**Thesis** Zdrojové kódy a vygenerované PDF této bakalářské práce

**Design** Návrh UI