

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Výuková aplikace Zeměpis Evropy pro OS Android



2016

Vedoucí práce: Mgr. Jiří Zacpal,
Ph.D.

Jan Kašík

Studijní obor: Aplikovaná informatika,
prezenční forma

Bibliografické údaje

Autor: Jan Kašík
Název práce: Výuková aplikace Zeměpis Evropy pro OS Android
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2016
Studijní obor: Aplikovaná informatika, prezenční forma
Vedoucí práce: Mgr. Jiří Zaccpal, Ph.D.
Počet stran: 38
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Jan Kašík
Title: Educational application European Geography for OS Android
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2016
Study field: Applied Computer Science, full-time form
Supervisor: Mgr. Jiří Zaccpal, Ph.D.
Page count: 38
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Cílem práce je vytvořit výukovou aplikaci pro získání a procvičení základních zeměpisných znalostí u osoby s mentálním postižením. Aplikace je určena pro OS Android. Aplikace je vytvářena ve spolupráci s Klíčem, centrem sociálních služeb.

Synopsis

Goal of this work is creating educational application for acquisition and practicing basic geography knowledge for persons with mental disabilities. Application is intended for OS Android. Application is created in cooperation with Klíč, social services center.

Klíčová slova: android; aplikace pro android; výuka zeměpisu; vzdělávání

Keywords: android; android application; geography learning; education

Děkuji vedoucímu bakalářské práce, panu Mgr. Jiřímu Zecpalovi, Ph.D za odborné vedení, konzultace a za věcné připomínky k práci. Děkuji také mé rodině a přítelkyni za jejich velkou podporu při studiu.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

1	Úvod	8
2	Podobné aplikace	8
2.1	Zeměpis - Země a hlavní města	8
2.2	GeoFlight Europe: Geography	10
2.3	All European Countries Quiz	11
2.4	Shrnutí	12
3	Použité technologie	12
3.1	Android	12
3.1.1	Historie	12
3.1.2	Verze systému	12
3.1.3	Rozhraní	12
3.1.4	Architektura	13
3.2	OrmLite	16
3.3	Mapsforge	16
3.4	ExoMedia	16
4	Popis řešení	16
4.1	Výuková část	16
4.2	Herní část	16
5	Programátorská příručka	17
5.1	Zdrojová data	17
5.1.1	Mapové podklady	17
5.1.2	Data pro geolokaci	18
5.1.3	Další data zemí	18
5.2	Implementace geolokace a reprezentace zemí	19
5.2.1	Třída Country	19
5.2.2	Třída Territory	19
5.2.3	Reverzní geolokace	19
5.3	Extrakce dat zemí	20
5.4	Zobrazení map	20
5.4.1	Publikování souborů do externí paměti	20
5.4.2	Barevné rozlišení jednotlivých zemí	20
5.5	Perzistentní uložení dat	21
5.6	Nastavení	22
5.7	SVG obrázky	23
5.8	Audio přehrávač	23
6	Uživatelská příručka	24
6.1	Menu	24
6.2	Výuka	24

6.3	Kvíz	27
6.4	Najdi stát na mapě	29
6.5	Nastavení	29
6.5.1	Správa uživatelů	31
6.5.2	Změna jazyka	31
	Závěr	32
	Conclusions	33
	A Obsah přiloženého CD/DVD	34
	Seznam zkratk	35
	Literatura	36

Seznam obrázků

1	Zeměpis - Země a hlavní města [1]	9
2	GeoFlight Europe: Geography [2]	10
3	All European Countries Quiz [3]	11
4	Podíl jednotlivých verzí systému na používaných zařízeních [6] . .	13
5	Architektura systému Android[10]	15
6	Hlavní menu aplikace	24
7	Aktivita pro výuku	25
8	Galerie v informacích o dané zemi	26
9	Kvíz s vlajkami	27
10	Aktivita s výsledky	28
11	Najdi stát na mapě podle názvu hlavního města	29
12	Hlavní obrazovka nastavení	30

Seznam tabulek

1	Podíl jednotlivých verzí systému na používaných zařízeních. Data byla sesbírána během sedmidenního období končícího dnem 1. 7. 2016. Verze s podílem nižším než 0,1% nejsou uvedeny.[6]	14
---	---	----

1 Úvod

Cílem práce je vytvořit výukovou aplikaci pro získání a procvičení základních zeměpisných znalostí u osoby s mentálním postižením. Aplikace je určena pro operační systém Android. V aplikaci bude možné přepínat mezi různými jazykovými mutacemi.

Aplikace se bude sestávat ze dvou částí. První část bude obsahovat samotnou učební látku, kterou může uživatel libovolně procházet. Druhou částí pak bude sekce sloužící k procvičení znalostí pomocí několika jednoduchých her. Hráč bude motivován odměnou v podobě bodů a možnostmi porovnávat své skóre s jinými hráči.

2 Podobné aplikace

Aplikací, které slouží pro výuku zeměpisu Evropy existuje několik. Autoři je ve většině případů pojali pouze jako hru, která měla dodatečný efekt – a to procvičení zeměpisných znalostí. Do výběru jsem tedy zařadil pouze aplikace, které obsahovaly „výukovou“ část.

2.1 Zeměpis - Země a hlavní města

Tato aplikace na první pohled zaujme profesionálním a čistým designem. Kromě možnosti získávání nových znalostí nabízí několik her pro jejich procvičení v několika úrovních obtížnosti. Součástí „herní“ části je také malá sociální síť. Uživatelé zde mohou porovnávat své skóre s výsledky ostatních hráčů, kteří se připojí pomocí Google účtu.

Jako podklad používá aplikace Google Maps. Ačkoliv tedy její rozhraní zůstává v angličtině, tak názvy států a měst jsou přeloženy. Výběr jazyka lze změnit v nastavení.

Nepříjemností je zobrazování reklam, ty však odstraňuje placená verze za 40,93 CZK.



Obrázek 1: Zeměpis - Země a hlavní města [1]

2.2 GeoFlight Europe: Geography

Tato aplikace je spíše hra, kde i samotné učení má podobu létání s letadlem nad Evropou. Hra nabízí tři základní sekce. První je samotné učení, druhé je létání s letadlem v daném časovém limitu do dané země a třetí je kvíz ve kterém hráč vybírá správnou odpověď. Aplikaci jsem bohužel nemohl vyzkoušet – je placená a stojí 2,99 EUR.



Obrázek 2: GeoFlight Europe: Geography [2]

2.3 All European Countries Quiz

Pěkně zpracovaná aplikace, která nabízí hry, kde je nutné poskládat název státu z dostupných písmenek nebo hry, kde hráč musí vybrat správnou odpověď na otázku. Návykové je především ozvučení jednotlivých úkonů.

Pro srovnání výkonu aplikace nabízí „European Challenge“, ve kterém je cílem získat co nejvíce bodů za správné odpovědi. V aplikaci je pak uložen vždy nejlepší výsledek, jakému se uživateli podařilo dosáhnout.

Aplikace je zdarma a pro svou podporu vydatně zobrazuje reklamy, které leckdy nepříjemně zasahují do hry.



Obrázek 3: All European Countries Quiz [3]

2.4 Shrnutí

Ačkoliv her a kvízů, které se věnují zeměpisu je opravdu velké množství, vybral jsem tři aplikace, které mě zaujaly nejvíce. Většinu aplikací na Google Play totiž tvořili pouze jednoduché kvízy se špatně přeloženým popisem a nicneříkajícími obrázky.

Nejvíce mě pak zaujala aplikace „Zeměpis - Země a hlavní města“, které bych se chtěl alespoň přiblížit.

3 Použité technologie

3.1 Android

Android je operační systém postavený nad Linuxovým jádrem používaný převážně na mobilních zařízeních jako jsou mobilní telefony a tablety. Android je primárně uzpůsoben pro ovládání dotykem a v současné době ho vyvíjí společnost Google.

3.1.1 Historie

První stabilní verze, již vydaná společností Google, která koupila společnost Android, Inc. v roce 2005 – Android 1.0 byla vydána v září roku 2008. Prvním zařízením na trhu používajícím Android bylo T-mobile G1 (před vydáním označované jako HTC Dream)[4]. Toto vydání již zahrnovalo všechny typické prvky pro tuto platformu.

3.1.2 Verze systému

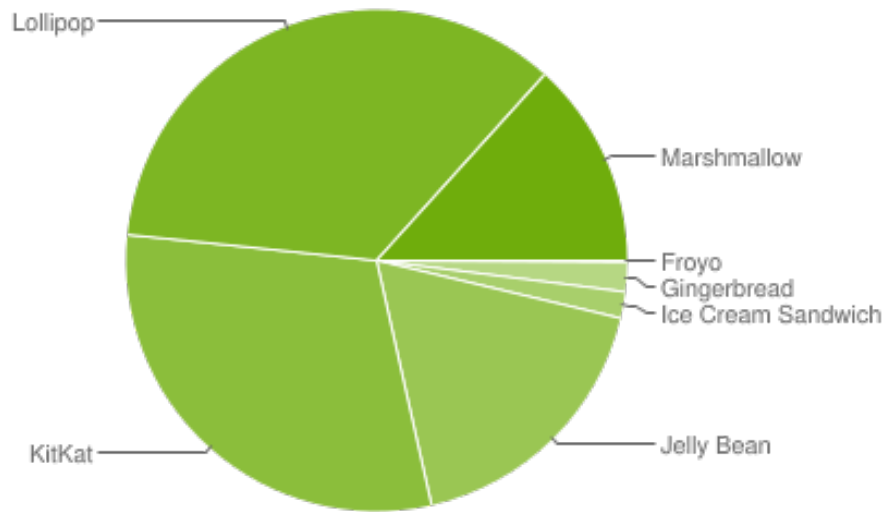
Vývoj systému Android prošel do dnešního dne při svém vývoji mnoha verzemi. Revolučních verzí bylo několik. Nejvýznamnější z pohledu poslední doby a pro účely této práce je verze 4.0 Ice Cream Sandwich, která sjednotila verze systému pro tablet i pro telefony. Do té doby používané číslování 3.x pro tablety a 2.x pro telefony tedy zaniklo.

S každou novou verzí systému je také zveřejněno nový [Software Development Kit \(SDK\)](#) a [Application Programming Interface \(API\)](#). Verze API je označována jako [API level](#) a současný Android 6.0 má [API level 23](#)[5].

Z tabulky 1 a obrázku 4 je vidět, že podíl verzí Androidu 4.0 a vyšší tvoří více než 95% všech verzí. Je tedy zřejmé, že vyvíjet aplikaci má smysl právě pro tuto skupinu, aby byla zaručena kompatibilita na co nejvíce používaných zařízeních a aby měl programátor zároveň k dispozici co nejvíce funkcí, které Android API nabízí. Pro tuto aplikaci tedy bude použit [API level 15](#).

3.1.3 Rozhraní

Uživatelské rozhraní Androidu je založené na přímé manipulaci s prvky rozhraní pomocí dotyků a gest jako tažení, scrollování a podobně. Reakce na uživatelův



Obrázek 4: Podíl jednotlivých verzí systému na používaných zařízeních [6]

vstup je vždy téměř okamžitá. Text je zadáván ve většině případů pomocí virtuální klávesnice, která je integrována v systému. U většiny zařízení je možné připojit i externí klávesnici i myš pomocí USB portu. Díky tomu, že je Android optimalizovaný pro ovládání dotykem nemá tato možnost na dotykově ovládaných zařízeních příliš mnoho využití.

3.1.4 Architektura

Architektura systému android má celkem pět vrstev přičemž každá vrstva má své specifické funkce.

1. Na nejnižší úrovni vrstvou je **linuxové jádro**. Toto jádro zajišťuje abstrakci nad hardware a tedy základní komunikaci mezi hardware a software. Upravený kernel však neobsahuje všechny nástroje, které jsou pro něj obvyklé ani podporu glibc. Vývojáři byl vybrán především proto, že nabízí výbornou správu paměti, plánování procesů, dobře zpracovaný bezpečnostní model a jeho zdrojový kód je dostupný jako open source.
2. Nad jádrem je **sada nativních knihoven** obsahujících funkcionalitu, která je dostupná všem aplikacím. Jedná se například o implementaci relačních databází SQLite, webovým prohlížečem využívaný WebKit, šifrování a autentizace pomocí SSL, optimalizovanou verze libc (není kompatibilní s glibc), OpenGL sloužící pro 3D vykreslování, Media Framework poskytující funkce pro přehrávání a zobrazení nejrozšířenějších formátů multimédií a další. Tyto knihovny jsou napsány v jazyce C/C++.
3. **Běhové prostředí** systému Android má velký vliv na rychlost aplikací a proto byl navržen s ohledem na to, že Android je určený zejména pro mo-

Verze	Označení	API level	Podíl
2.2	Froyo	8	0,1%
2.3.3 – 2.3.7	Gingerbread	10	1,9%
4.0.3 – 4.0.4	Ice Cream Sandwich	15	1,7%
4.1.x		16	6,4%
4.2.x	Jelly Bean	17	8,8%
4.3		18	2,6%
4.4	KitKat	19	30,1%
5.0		21	14,3%
5.1	Lollipop	22	20,8%
6.0	Marshmallow	23	13,3%

Tabulka 1: Podíl jednotlivých verzí systému na používaných zařízeních. Data byla sesbírána během sedmidenního období končícího dnem 1. 7. 2016. Verze s podílem nižším než 0,1% nejsou uvedeny.[6]

bilní zařízení. Do verze 4.3 byl využíván virtuální zásobníkový stroj Dalvik. Programy pro systém Android bylo potřeba nejdříve zkompileovat do Java bajtkódu, který byl následně převeden do optimalizovaného bajtkódu Dalviku (*.dex*) nástrojem dx[7]. Ve verzi 2.2 byla do Dalviku přidána **just-in-time (JIT)** kompilace. V tomto řešení je aplikace profilována při každém spuštění a nejčastěji používané úseky jsou kompilovány do nativního kódu, čímž je dosaženo ještě vyšší rychlosti[8].

Od verze 4.4 byl Dalvik nahrazen prostředím **Android Runtime (ART)**. **ART** využívá **ahead-of-time (AOT)** kompilace. To znamená, že celá aplikace je zkompileována do nativního kódu během instalace do zařízení. Díky tomu, že je proces překladu oproti **JIT** proveden pouze jednou, se snížila celková spotřeba energie zařízení. Pro zachování zpětné kompatibility používá **ART** stejný vstupní formát jako Dalvik (*.dex*).[9]

4. Pro programátora nejdůležitější vrstvou je **Application Framework**. **API** obsahuje řadu služeb v podobě knihoven, které jsou napsané v jazyce Java a přístupné pomocí class souborů při psaní aplikací.
5. Nejvyšší vrstvou jsou **aplikace**. Skrze ně komunikuje uživatel s nižšími vrstvami. Aplikace jsou do systému instalované uživatelem nebo jsou obsaženy jako výchozí v systému (Nastavení, Fotoaparát, Telefon...).



Obrázek 5: Architektura systému Android[10]

3.2 OrmLite

Object Relation Mapping Lite poskytuje funkcionalitu, díky které je možné uložit data vytvořená v objektově orientovaném systému do systému relačního. V praxi pak vypadá konverze následovně: Třída, zjednodušeně předpis objektu, je v relačním modelu reprezentována jako relační schéma. Množinou atributů jsou názvy slotů. Instance dané třídy (objekt) je převedena jako n-tice nad tímto relačním schématem. Jednotlivá zobrazení v n-tici jsou pak hodnoty slotů daného objektu.

3.3 Mapsforge

Knihovna Mapsforge poskytuje stejné [API](#) jako třída `com.google.android.maps.MapView`, která umožňuje zobrazení map. Původní implementace umožňuje používat data pouze z Google Maps. Protože mým cílem bylo vytvořit aplikaci nezávislou na datovém připojení, použil jsem Mapsforge. Tato knihovna používá jako zdroj dat vlastní binární formát, do kterého lze převést jakýkoliv validní [OpenStreetMap \(OSM\)](#) soubor. Tedy mapu extrahovanou z OpenStreetMaps.

3.4 ExoMedia

Knihovna pro přehrávání multimediálních souborů. Je použitelná na zařízeních s Android [API](#) 16 a vyšší.

4 Popis řešení

4.1 Výuková část

V této části si uživatel bude moci procházet samotná data. Na politické mapě se po kliknutí na území daného státu zobrazí základní informace o něm. Mezi základními informacemi tak nalezneme rozlohu státu, počet obyvatel, měny, kterými lze v zemi platit a jazyky, kterými se zde mluví.

Součástí těchto informací je také fotogalerie s fotkami z jednotlivých zemí a také základní národní symboly mezi něž patří vlajka a hymna daného státu.

4.2 Herní část

Hry jsou zábavnou formou učení. Uživatel má mnohem větší šanci si informace zapamatovat, pokud jsou procvičeny hrou. V aplikaci se budou nacházet tyto minihry:

- Uhádnutí státu na základě názvu jeho hlavního města.
- Uhádnutí státu na základě názvu jeho vlajky
- Nalezení státu na mapě na základě jeho vlajky.

- Nalezení státu na mapě na základě jeho názvu.
- Nalezení státu na mapě na základě názvu jeho hlavního města.

5 Programátorská příručka

V této sekci popisují nejdůležitější algoritmy, struktury a postupy používané v aplikaci.

5.1 Zdrojová data

5.1.1 Mapové podklady

Získání mapových podkladů bylo jednou z náročných částí přípravy aplikace. Zdrojová data pochází z projektu Natural Earth Data. Použil jsem administrativní mapu „Admin 0 – Countries“^[11], která nabízí hraniční body všech světových států. Po mnoha pokusech se mi osvědčil následující postup:

1. Pomocí skriptu ogr2osm^[12] jsem převedl shapefile na mapová data ve formátu Open Street Map – zdrojový kód 1.

```
1 ./ogr2osm.py ne_50m_admin_0_countries.shp
2   -o ogr2osm_ocean_translated.osm
3   -t translations/admin_countries.py
```

Zdrojový kód 1: Použití nástroje ogr2osm

Zajímavý je přepínač `-t`, který dovoluje uvést cestu k souboru, který obsahuje „háčky“, jež jsou zavolány v určitých fázích převodu. Takto lze jednoduše definovat překlad jednotlivých atributů a zajisti, aby finální překlad proběhl hladce. Já jsem použil vlastní sadu konverzních pravidel. Najdete ji v `/osm-parser/translations/admin_countries.py`.

2. V prvním kroku jsem získal [OSM](#) soubor, ale atributy zůstaly zachovány ve stejné podobě, jako jsou uvedeny v původním shapefile¹. Získaný soubor také není možné použít v následujícím kroku, protože zásuvný modul pro Osmosis^[14] nemá úplně ideální chování. Pokud nejsou uzly seřazené podle hodnot atributů nebo pokud dojde k další nepřijímané kombinaci atributů, konverze do binárního formátu nelze provést.

Řešením je nastavit těmto atributům „neutrální“ hodnotu. Například pomocí skriptu `xapi_attr.py`^[15].

```
./xapi_attr.py ogr2osm_admin_translated.osm.
```

¹Pro ověření je možné použít nástroj JOSM^[13].

3. Protože zdrojovým souborem byla data pro celý svět, která nepotřebujeme, ořízneme soubor tak, aby obsahoval pouze data Evropy. Toho dosáhneme nástrojem Osmconvert[16]. Stačí zde pomocí přepínače `-b` definovat ohraničující čtverec, který bude vyříznut jako ve zdrojovém kódu 2.

```
1 osmconvert --complete-ways
2   --drop-broken-refs
3   -b=-31.266,27.636,39.869,81.009
4   ogr2osm_admin_translated_attr.osm
5   -o=europe.osm
```

Zdrojový kód 2: Použití nástroje osmconvert

Pro další přepínače doporučuji nastudovat manuál.

4. Nyní stačí finální osm soubor převést do binárního souboru, který je zpracovatelný knihovnou Mapsforge pomocí pluginu[17] pro Osmosis: `~/apps/osmosis-latest/bin/osmosis --rx file=europe.osm --mw file=europe.map zoom-interval-conf=5,0,7`

Pokud bysme vynechali konverzi atributů pomocí „háčků“ z prvního kroku, výsledný map soubor by byl prázdný, protože program by atributy ze shapefile nerozeznal.

5.1.2 Data pro geolokaci

Dalšími důležitými zdrojovými daty jsou data pro reverzní geolokaci. Je to metoda, která slouží k tomu, aby bylo možné na základě souřadnic určit, do jakého státu daný bod spadá.

Zde nebylo možné použít stejný binární soubor obsahující mapové podklady, protože knihovna Mapsforge nenabízí [API](#) poskytující tuto funkcionalitu.

Pro snadnost zpracování a přehlednost jsem se rozhodl použít data uložená v JSON souborech. Původně jsem používal vlastní formát, ale posléze jsem začal používat standart GeoJson[18].

Souřadnice polygonů území jednotlivých zemí jsou extrahována z [OSM](#) souboru před převedím do map formátu. Další data, jako ohraničující čtverce jednotlivých poligonů, nebo údaje o měně a jazycích, kterými se mluví v daných státech jsou získávána z různých Python knihoven.

Pro vytváření těchto souborů jsem vytvořil malý skript v jazyku Python `osm-parser`.

5.1.3 Další data zemí

Tato data jsou vkládána do stejných souborů jako data pro geolokaci. Jako zdroj je použita sbírka dat v několika formátech[19]. Tato data obsahují jednotlivé ISO kódy, rozlohu, údaje o sousedech a další užitečné informace. Jsou rovněž zpracována skriptem `osm-parser`.

5.2 Implementace geolokace a reprezentace zemí

5.2.1 Třída Country

Třída Country představuje základní abstrakci nad státem jako administrativní jednotkou. Obsahuje základní informace o dané zemi a také metodu `contains(double longitude, double latitude)`, pomocí které lze zjistit, zda zadané souřadnice patří dané zemi.

5.2.2 Třída Territory

Představuje abstrakci nad územím dané země. Nejdůležitějšími metodami jsou metody zjišťující zda dané souřadnice patří do daného území. Území však může mít více částí – například ostrovy. Proto je tato třída kompozicí instancí třídy `TerritoryPolygon`, z nichž každá reprezentuje jeden souvislý celek území.

5.2.3 Reverzní geolokace

Reverzní geolokace je metoda, při které je zjišťováno, zda daný objekt náleží do nadřazeného objektu. V tomto případě zjišťuji, zda bod náleží do území nějaké země a pokud ano, tak do kterého.

Popis použitého algoritmu pomocí pseudokódu naleznete ve zdrojovém kódu 3. Pro zajištění vyšší efektivity jsou nejdříve hledány země, do jejichž hraničního čtverce daný bod patří. Tím se podstatně zúží počet procházených polygonů a tudíž se celý proces zrychlí.

```
1 vstup = bod{zeměpisná šířka, zeměpisná délka}
2 výstup = země, do které bod patří nebo null, pokud leží například
   v oceánu
3
4 possibleCountries <- []
5 for each country in countries:
6     if bod is in bounding box of country:
7         possibleCountries.append(country)
8     endif
9 endfor
10
11 for each country in possibleCountries:
12     if bod is inside territory of country:
13         return country
14     endif
15 endfor
16
17 return null
```

Zdrojový kód 3: Používaný algoritmus pro reverzní geolokaci

Vnitřně pak funguje hledání v instanci třídy `Territory` tak, že se postupně zjišťuje, zda daný bod patří do některého polygonu daného území. Pro toto je

použit klasický algoritmus pro zjišťování přítomnosti bodu uvnitř nekonvexního polygonu.

Zastřešení nad reverzní geolokací nabízí třída `LocationManager`, která by měla být výhradně používána k tomuto účelu.

5.3 Extrakce dat zemí

Ve výchozím stavu jsou data zemí extrahována ze souborů formátu GeoJSON[18], jak jsem uvedl výše.

Extrakce probíhá ve více vláknech, podle toho, kolik je k dispozici procesorových jader. Na vícejádrových procesorech by tedy měla teoreticky extrakce proběhnout rychleji. Na mém tabletu Dell Venue 7 při dvou jádrech procesoru trvá extrakce přibližně deset vteřin. Na telefonu Vodafone Smart Prime 6 je pak výsledný čas v průměru kolem dvou až tří vteřin.

Způsob extrakce lze libovolně měnit. Stačí vytvořit novou implementaci rozhraní `CountryExtractor` a zajisti nový zdroj dat.

5.4 Zobrazení map

Mapy jsou zobrazovány pomocí knihovny `Mapsforge`[20]. Pro správnou funkci je třeba, aby knihovna měla k binárnímu souboru s mapami náhodný přístup. Soubor tedy nemůže být v nainstalovaném stavu umístěn ve složce `/assets/` nebo v `/res/`, ale musí být umístěny do interní nebo externí paměti zařízení.

5.4.1 Publikování souborů do externí paměti

Třída `Publisher` zajišťuje kopírování souborů z lokace `/assets/publish/` do externí paměti zařízení. Soubory, které chceme takto publikovat tedy stačí umístit do složky `/assets/publish/` a po instalaci aplikace budou přístupné v lokaci získané pomocí `FileHelper.getApplicationExternalStoragePath(context)`.

5.4.2 Barevné rozlišení jednotlivých zemí

Obarvování je implementováno pomocí rozšíření třídy `org.mapsforge.map.layer.overlay.Polygon`. Třída `cz.honzakasik.geography.common.location.map.Polygon` umožňuje nad vrstvou mapy vykreslit další vrstvu – „overlay“. Samotné obarvování a animaci barev pak zajišťuje třída `cz.honzakasik.geography.common.location.map.CountryPolygonOverlayHandler`.

Výhodou je, že polygony pro barevnou vrstvu jsou tvořeny ze stejných dat jako data pro geolokaci. Díky přidané vrstvě také není nutné v mapovém podkladu provádět žádné úpravy.

Nevýhodou pak je, že tato vrstva překryje cokoliv, co je zobrazeno na mapovém podkladu. Je proto využito možnosti vytvořit poloprůhledné barvy, kterými jsou poté plochy odpovídající území zemí vybarveny.

5.5 Perzistentní uložení dat

K perzistentnímu uložení dat je použita knihovna ORMLite[21]. Použití je velmi snadné. Pokud chci instanci třídy `SimpleObject` obsahující dva sloty uložit pomocí ORMLite, stačí použít anotace `@DatabaseField` a `@DatabaseTable` – viz příklad 4.

```
1     @DatabaseTable(tableName = "simple_objects")
2     public class SimpleObject {
3
4         @DatabaseField(generatedId = true, canBeNull = false)
5         private int id;
6
7         @DatabaseField(canBeNull = false)
8         private Object slot;
9     }
```

Zdrojový kód 4: Použití anotace `@DatabaseField` a `@DatabaseTable`

Dotazování pak probíhá pomocí třídy `com.j256.ormlite.android.apptools.OrmLiteSqliteOpenHelper`. Tuto třídu je ve většině případů více než vhodné rozšířit. Pomáhá spravovat připojení k databázi a také obsahuje metody, které jsou zavolány při vytvoření databáze nebo jejím upgrade. Příklad rozšíření třídy uvádí zdrojový kód 5.

```
1     public class MyDatabaseHelper extends OrmLiteSqliteOpenHelper {
2         ...
3
4         @Override
5         public void onCreate(SQLiteDatabase database, ConnectionSource
6             connectionSource) {
7             ...
8         }
9
10        @Override
11        public void onUpgrade(SQLiteDatabase database, ConnectionSource
12            connectionSource, int oldVersion, int newVersion) {
13            ...
14        }
```

Zdrojový kód 5: Rozšíření třídy `OrmLiteSqliteOpenHelper`

V aktivitě, kde chci přistoupit k datům stačí použít singleton vzor pro instanci třídy `MyDatabaseHelper`. Tedy vytvořit metodu `getHelper()`, která mi buď vrátí existující instanci `MyDatabaseHelper` nebo vytvoří novou pomocí `OpenHelperManager.getHelper(CONTEXT, MyDatabaseHelper.class)`. Kvůli nebezpečí souběhu není

možné otevírat více `MyDatabaseHelper` současně. Je také třeba `MyDatabaseHelper` uvolnit při zániku aktivity v metodě `onDestroy()` pomocí `OpenHelperManager.releaseHelper()`.

Pokud je vše správně naimplementováno, pro získání všech uložených objektů třídy `SimpleObject`, stačí zavolat `getHelper().getDao(SimpleObject.class).queryForAll()` vracející seznam všech obsahující všechny objekty.

Tímto způsobem jsou v aplikaci uloženy záznamy o uživateli a záznamy výsledků v herní části aplikace.

5.6 Nastavení

Veškeré nastavení je řešeno pomocí prostředků, které poskytuje Android [API](#). Jedná se především o `PreferenceFragment` [22] a `SharedPreferences` [23]. V aplikaci by pro veškeré přístupy do nastavení měla být používána výhradně třída `cz.honzakasik.geography.settings.PreferenceHelper`, která slouží jako abstrakce nad klíči nastavení a jejich datovými typy. Tato třída by měla být také vhodně rozšiřována při přidání dalších možností nastavení.

Pro přidání dalšího překladu, je třeba vytvořit přeložený soubor `strings.xml` a umístit jej do složky `/res/values-<locale>/`. Aby se nový překlad projevil v nastavení je třeba přidat odpovídající instanci typu `Locale` do pole `cz.honzakasik.geography.settings.SettingsConstants#AVAILABLE_TRANSLATIONS`. Vše by nyní mělo fungovat.

Při vytvoření `PreferenceFragment` v metodě `onCreate()` stačí přidat předkonfigurované možnosti nastavení pomocí `addPreferencesFromResource(R.xml.preferences)`. Samotným zdrojem nastavitelných „proměnných“ je xml soubor popisující podobu a možnosti nastavení, více v příkladu 6.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <PreferenceScreen xmlns:android="http://schemas.android.com/apk/
  res/android">
3   <PreferenceCategory
4     android:title="User setting">
5     <CheckBoxPreference
6       android:key="onOff"
7       android:title="On/Off switch"
8       android:summary="Switches foo on or off" />
9 </PreferenceScreen>
```

Zdrojový kód 6: Příklad XML zdroje nastavení

Tento příklad definuje pouze jedinou kategorii nastavení a v ní jediný checkbox, kterému je nastaven popis pomocí atributu `title` a vysvětlovací text atributem `summary`. Atribut `key` definuje klíč, pod kterým je dané nastavení uloženo. Typ uložené hodnoty je `boolean`. Z nastavení ji lze získat zavoláním `PreferenceManager.getDefaultSharedPreferences(context).getBoolean("onOff", false)`. V případě, že není hodnota nalezena je vráceno `false`.

Na pozadí je uloženo pomocí XML souboru. Proto je nastavení vhodné pouze pro primitivní typy a není doporučeno do něj ukládat složitější objekty.

5.7 SVG obrázky

Pro vlajky je v aplikaci použit vektorový formát [Scalable Vector Graphics \(SVG\)](#). Ten je vhodný především proto, že kvalita zobrazení není závislá na DPI a rozlišení displeje cílového zařízení. Protože Android zobrazení obrázků v tomto formátu nepodporuje, použil jsem k tomuto účelu knihovnu [AndroidSVG](#)[24].

Knihovna v aplikaci provádí renderování [SVG](#) do instance třídy `android.graphics.Picture`. Tento převod, zvláště u složitějších obrázků s detaily, zabere nějaký čas a proto je nutné jej provádět asynchronně. K tomuto účelu slouží v aplikaci třída `cz.honzakasik.geography.common.tasks.LoadFlagImageTask`.

Samotné přiřazení obrázku do `ImageView` je možné provádět v implementaci rozhraní `PostExecuteTask<Picture>`, kterou je třeba předat jako argument konstruktoru `LoadFlagImageTask`. Metoda `run()` tohoto rozhraní bude spuštěna po tom, co je renderování dokončeno.

Pro obrázky vlajek je také vytvořena globální cache, která je umístěna v aplikační kontextu ve třídě `cz.honzakasik.geography.App`, která je využívána v `LoadFlagImageTask` a pokud nastane cache hit, je vyrenderovaný `Picture` vrácen okamžitě.

5.8 Audio přehrávač

Audio přehrávač je v aplikaci používán pro přehrávání národních hymen jednotlivých zemí. Původně jsem zamýšlel použít vestavěnou třídu `android.media.MediaPlayer`[25], která dokáže přehrávat multimediální formáty. Přestože však podporuje kontejnery Ogg s obsahem ve formátu Vorbis[26], nezvládne některé soubory přehrát. Jedná se pravděpodobně o takové soubory, které mají proměnlivý datový tok[27], přiložené informace o skladbě[28] nebo příliš vysokou vzorkovací frekvenci[29].

Být závislý jen na určitém stavu daného souboru není dobrá volba. Druhá možnost – převod mnoha skladeb do vhodného formátu by pak zabrala příliš času. Třetí možností bylo hledání náhrady třídy `MediaPlayer`.

Po krátkém hledání jsem narazil na `ExoPlayer`[30], který již zmiňovanými problémy netrpí. Abstrakce nad hardware je ale pro potřeby aplikace příliš nízká.

Knihovna `ExoMedia`[31], vystavěna právě nad `ExoPlayer`, tento problém řeší. Její třída `EMAudioPlayer` implementuje stejné [API](#) jako původní `MediaPlayer`[32] v systému Android a umožňuje tak pohodlně přehrávat téměř všechny běžně dostupné formáty audio souborů.

Veškerá funkcionalita přehrávače v aplikaci se nachází ve zdokumentované třídě `AudioPlayerView`. Hymna pro každý stát je pak hledána v adresáři odpovídající vzoru `/assets/anthems/<ISO 3166-1 alpha-2>/`. Je zde očekáván právě jeden soubor, při jiném počtu nalezených souborů dojde k chybě.

6 Uživatelská příručka

6.1 Menu

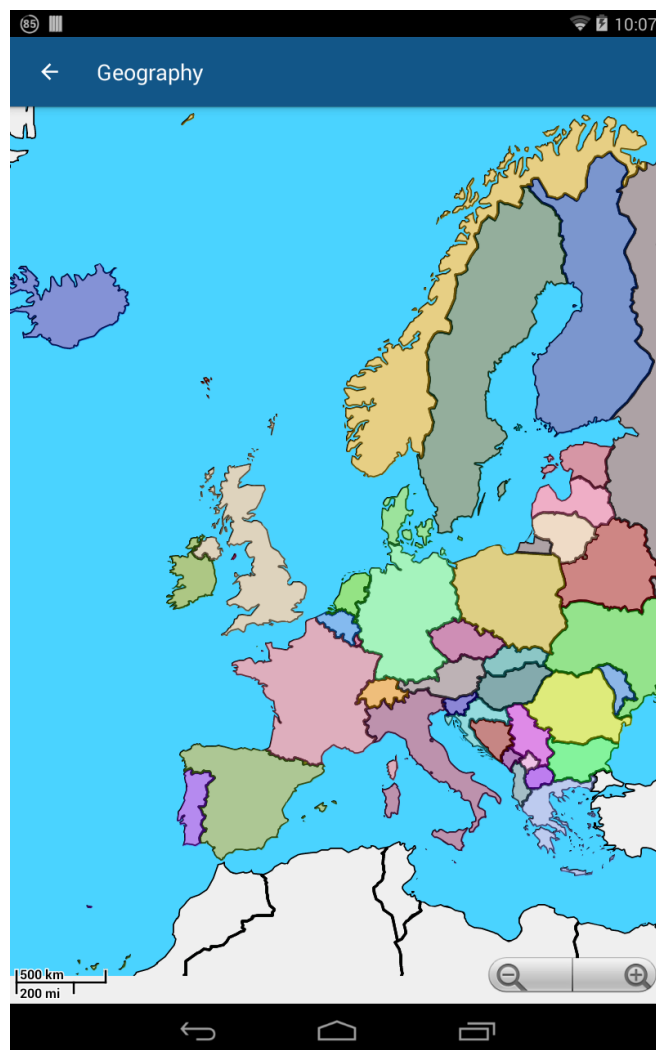


Obrázek 6: Hlavní menu aplikace

6.2 Výuka

Do výukového okna se z hlavního menu dostanete po klepnutí na tlačítko s talárem. Otevře se mapa s politickým rozdělením Evropy stejně jako na obrázku 7. Po kliknutí na jakoukoliv zabarvenou zemi se otevře okno s podrobnými informacemi o dané zemi, jejíž součástí je i galerie fotografií stejně jako na obrázku 8.

Galerie však není jedinou částí tohoto okna. Okno je rozděleno na tři záložky. Na první se nachází podrobné informace o vybrané zemi, na druhé právě galerie



Obrázek 7: Aktivita pro výuku



Obrázek 8: Galerie v informacích o dané zemi

s fotkami dané země a na třetí pak vlajka a přehrávač, který přehraje hymnu země.

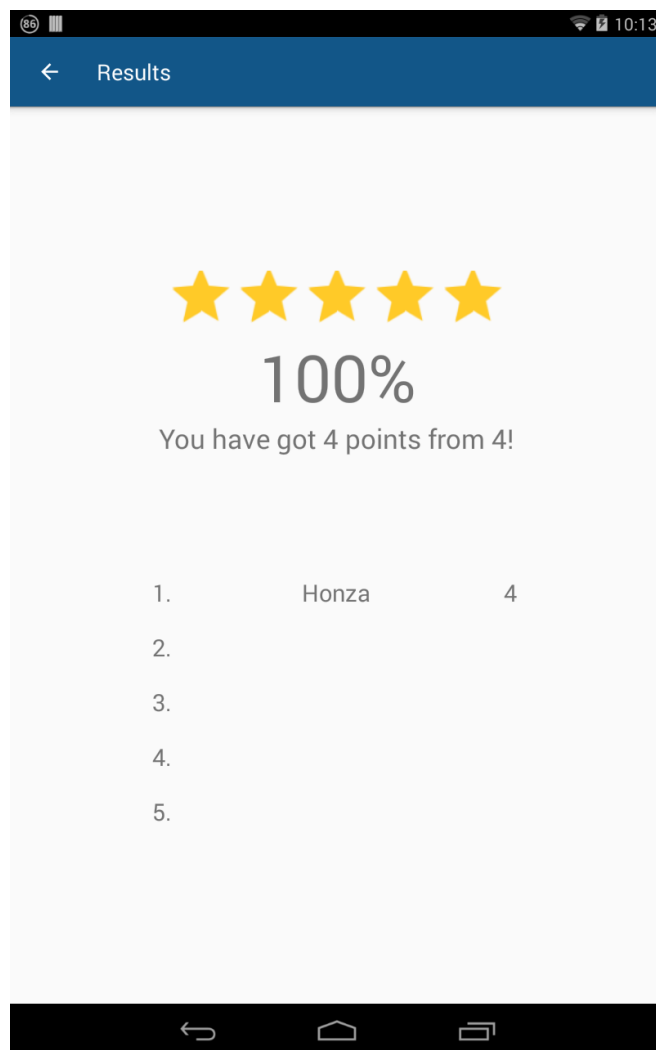
6.3 Kvíz

Po kliknutí na tlačítko s otazníkem v hlavním menu se dostanete do podmenu s kvízy. Liší se v typu otázky. Na obrázku 9 je kvíz, ve kterém hráč musí uhádnout jméno státu podle zobrazené vlajky. Po skončení hry se otevře okno s výsledky.



Obrázek 9: Kvíz s vlajkami

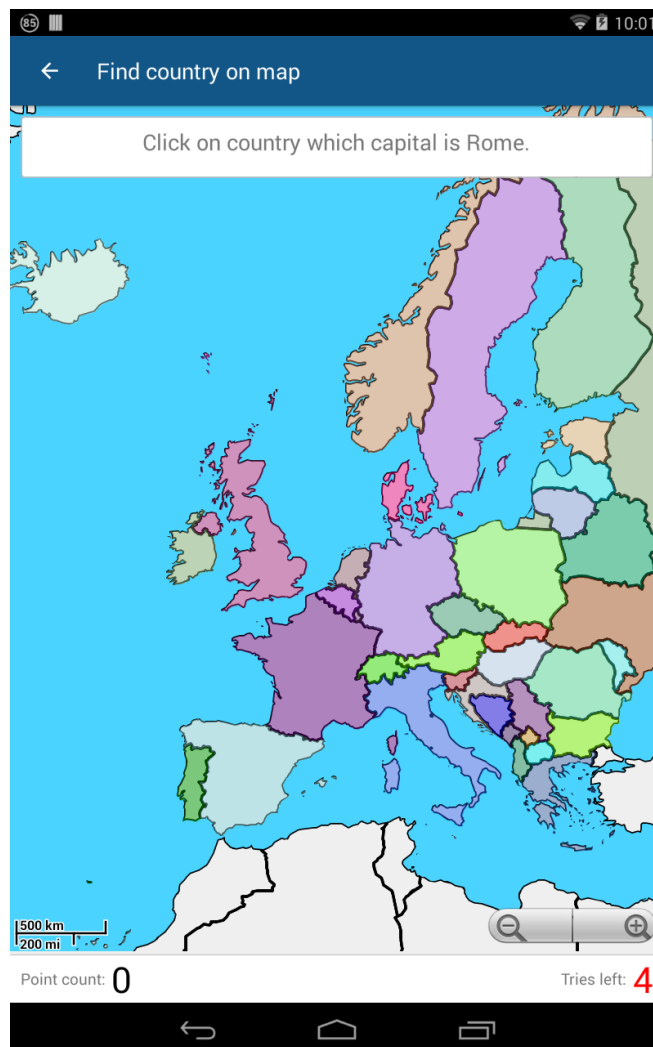
Pokud je v nastavení povolena možnost více uživatelů, zobrazí se rovněž tabulka výsledků. V ní je prvních pět nejlepších výsledků jako na obrázku 10. Tyto výsledky se zobrazí pouze od hráčů, kteří mají nastaven stejný stupeň obtížnosti jako aktuálně hrající hráč.



Obrázek 10: Aktivita s výsledky

6.4 Najdi stát na mapě

Podmenu s mapovými kvízy se zobrazí po kliknutí na tlačítku s obrázkem lupy nad reliéfem Evropy. Jednotlivé kvízy se liší typem otázky. Hráč musí najít zemi podle vlajky, podle jména hlavního města – na obrázku 11, nebo podle jména státu. V pravém dolním rohu je počet pokusu, které uživateli zbývají. Po každém

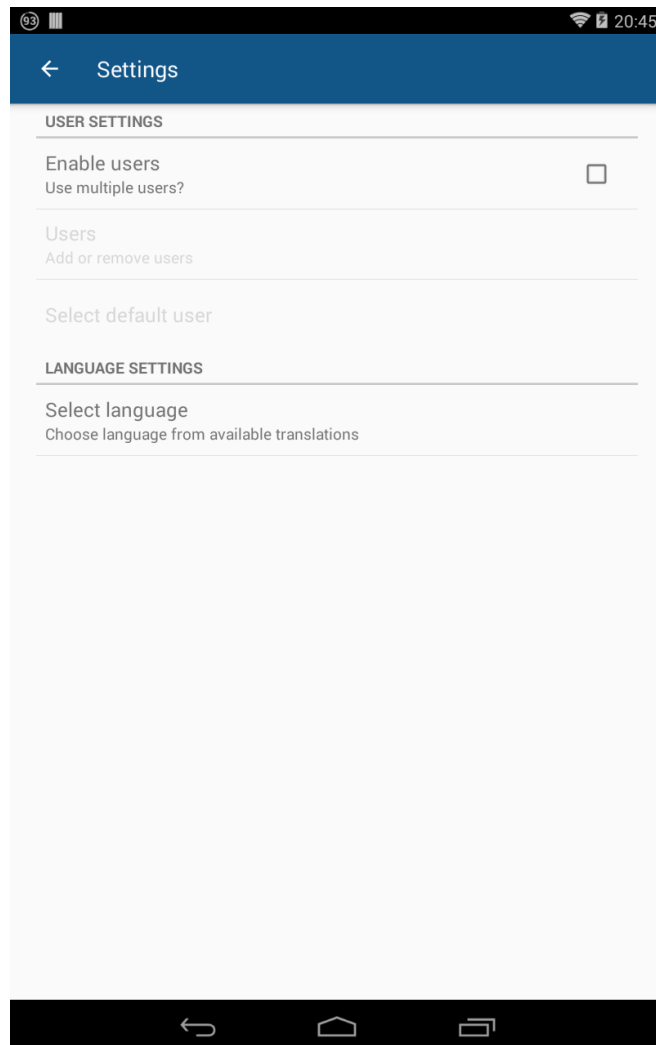


Obrázek 11: Najdi stát na mapě podle názvu hlavního města

neúspěšném pokusu, ze vybraná země zbarví červeně a počet pokusů se sníží. Po úspěšném pokusu se země zbarví zeleně a kvíz se posune na další otázku. Po skončení kvízu se opět zobrazí okno s výsledky.

6.5 Nastavení

V nastavení lze měnit některé herní aspekty aplikace a také změnit primární jazyk aplikace.



Obrázek 12: Hlavní obrazovka nastavení

6.5.1 Správa uživatelů

Nastavení obsahuje správu uživatelů. Po povolení více uživatelů je možné spravovat jednotlivé uživatele a také zvolit výchozího uživatele. Tento uživatel bude „hrát“ veškeré hry a pod jeho jménem se budou ukládat výsledky. Pokud nebude výchozí uživatel vybrán, budete k jeho výběru vyzváni po spuštění první hry.

Pro přidání nového uživatele stačí z hlavní obrazovky nastavení (obrázek 12) zvolit položku „Uživatelé“. Po klepnutí se otevře okno pro správu uživatelů.

Nového uživatele lze přidat klepnutím na tlačítko „+“ v pravém dolním rohu. Při přidání nového uživatele lze nastavit jeho přezdívku a obtížnost, pod kterou bude hrát. Úrovně obtížností se liší v počtu pokusů, které má hráč na každou otázku v kvízu.

Pokud chcete upravit uživatele – změnit jeho přezdívku nebo úroveň obtížnosti – stačí klepnout na přezdívku uživatele a otevře se podobný dialog jako při přidání nového uživatele s již vyplněnými údaji upravovaného uživatele. Po provedení úprav stačí stisknout tlačítko „Uložit“.

Odebrat uživatele lze dlouhým stiskem vybrané přezdívky. V dolním pravém rohu se poté objeví červená ikona po jejímž stisku je uživatel odstraněn.

6.5.2 Změna jazyka

V nastavení je také možné změnit jazyk pokud by výchozí nastavení náhodou uživateli nevyhovovalo. Jazyky jsou vybírány z dostupných překladů. Postup přidání překladu je popsán v programátorské příručce.

Závěr

Cílem práce bylo vytvořit aplikaci pro operační systém Android, zaměřenou na výuku zeměpisu pro osoby s mentálním postižením.

Toto byla moje první větší aplikace, kterou jsem kdy vyvíjel a také moje první aplikace pro mobilní platformy vůbec. Hlavním přínosem pro mě bylo vzdělání v pro mě nové oblasti – získání nových dovedností a osvojení si nových postupů a technologií. Vyzkoušel jsem si práci s mnoha knihovnamy, navrhnout uživatelské rozhraní i vnitřní funkčnost a strukturu aplikace.

V budoucnu lze jistě aplikaci doplnit dalšími překlady, případně vylepšit způsob, jakým jsou parsována data z GeoJSON souborů, což by vylepšilo čas potřebný k načtení aplikace o několik procent. Dalším vylepšením by mohlo být přidání kvízů zaměřujících se na podrobnější informace týkající se zemí jako je rozloha nebo počet obyvatel, měny nebo jazyky dané země.

Conclusions

The main goal of this work was to create an application for Android operating system focusing on teaching geography to persons with mental disabilities.

This was my first bigger application I have ever developed and it was also my first experience with developing for mobile platform ever. The main benefit for me was an education in a completely new area – gaining new skills and adopting new procedures and technologies. I experienced working with many libraries, designing user interface and also internal functionality and structure of the application.

Surely, it is possible to extend the application by new translations or improve the way of parsing GeoJSON files, which could lower the time needed for loading by a few percent. Another improvement would be adding quizzes focusing on more detailed information about countries like an area, population, currency or languages of a given country.

A Obsah přiloženého CD/DVD

Na samotném konci textu práce je uveden stručný popis obsahu přiloženého CD/DVD, tj. jeho závazné adresářové struktury, důležitých souborů apod.

application/

Sestavený apk soubor ze zdrojového kódu z adresáře `src/geography/` pro bezproblémovou instalaci.

doc/

Text práce ve formátu PDF, vytvořený s použitím závazného stylu KI PřF UP v Olomouci pro závěrečné práce, včetně všech příloh, a všechny soubory potřebné pro bezproblémové vygenerování PDF dokumentu textu (v ZIP archivu), tj. zdrojový text textu, vložené obrázky, apod.

src/

Kompletní zdrojové texty aplikace se všemi potřebnými (příp. převzatými) zdrojovými texty, knihovnami a dalšími soubory potřebnými pro bezproblémové vytvoření spustitelných verzí programu.

readme.txt

Instrukce pro instalaci a spuštění aplikace, včetně všech požadavků pro jeho bezproblémový provoz.

U veškerých cizích převzatých materiálů obsažených na CD/DVD jejich zahrnutí dovoluují podmínky pro jejich šíření nebo přiložený souhlas držitele copyrightu. Pro všechny použité (a citované) materiály, u kterých toto není splněno a nejsou tak obsaženy na CD/DVD, je uveden jejich zdroj (např. webová adresa) v bibliografii nebo textu práce nebo v souboru `readme.txt`.

Seznam zkratek

AOT ahead-of-time

API Application Programming Interface

ART Android Runtime

JIT just-in-time

OSM OpenStreetMap

SDK Software Development Kit

SVG Scalable Vector Graphics

Literatura

- [1] Yam Learning. *Zeměpis - Země a hlavní města – Aplikace pro Android ve službě Google Play* [online]. [cit. 2015-06-23]. Dostupný z: (<https://play.google.com/store/apps/details?id=com.yamlearning.geographylearning>).
- [2] TopoMonkey. *GeoFlight Europe: Geography – Aplikace pro Android ve službě Google Play* [online]. [cit. 2015-06-23]. Dostupný z: (<https://play.google.com/store/apps/details?id=air.nl.topomonkey.topoeuropahd>).
- [3] Solovyev, Andrey. *All European Countries Quiz – Aplikace pro Android ve službě Google Play*. *Google Play* [online]. [cit. 2015-06-23]. Dostupný z: (<https://play.google.com/store/apps/details?id=com.asmolgam.europe>).
- [4] gizmodo.com. *T-Mobile G1: Full Details of the HTC Dream Android Phone* [online]. [cit. 2015-06-25]. Dostupný z: (<http://gizmodo.com/5053264/t-mobile-g1-full-details-of-the-htc-dream-android-phone>).
- [5] Android Developers. *<uses-sdk> / Android Developers* [online]. [cit. 2016-07-25]. Dostupný z: (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html>).
- [6] Android Developers. *Dashboards / Android Developers* [online]. [cit. 2016-07-25]. Dostupný z: (<https://developer.android.com/about/dashboards/index.html>).
- [7] Bornstein, Dan. *Presentation of Dalvik VM Internals* [online]. [cit. 2015-06-28]. Dostupný z: (<http://sites.google.com/site/io/dalvik-vm-internals/2008-05-29-Presentation-Of-Dalvik-VM-Internals.pdf>).
- [8] Nickinson, Phil. *Google Android developer explains more about Dalvik and the JIT in Froyo* [online]. [cit. 2015-06-28]. Dostupný z: (<http://www.androidcentral.com/google-android-developer-explains-more-about-dalvik-and-jit-froyo>).
- [9] Frumusanu, Andrei. *A Closer Look at Android RunTime (ART) in Android L* [online]. [cit. 2015-06-28]. Dostupný z: (<http://anandtech.com/show/8231/a-closer-look-at-android-runtime-art-in-android-l/>).
- [10] Brady, Patrick. *Android Anatomy and Physiology* [online]. [cit. 2015-06-26]. Dostupný z: (<http://androidteam.googlecode.com/files/Anatomy-Physiology-of-an-Android.pdf>).
- [11] Natural Earth. *Admin 0 – Countries / Natural Earth* [online]. [cit. 2016-07-22]. Dostupný z: (<http://www.naturalearthdata.com/downloads/50m-cultural-vectors/50m-admin-0-countries-2/>).
- [12] Norman, Paul. *ogr2osm.py*. 2015. Dostupný také z: (<https://github.com/pnorman/ogr2osm>).

- [13] OpenStreetMap. *JOSM – OpenStreetMap Wiki* [online]. [cit. 2016-07-16]. Dostupný z: <http://wiki.openstreetmap.org/wiki/JOSM>.
- [14] OpenStreetMap. *Osmosis – OpenStreetMap Wiki* [online]. [cit. 2016-07-16]. Dostupný z: <http://wiki.openstreetmap.org/wiki/Osmosis>.
- [15] abxaz92. *xapi_attr.py*. 2015. Dostupný také z: https://github.com/abxaz92/mapscripts/blob/master/xapi_attr.py.
- [16] OpenStreetMap. *Osmconvert – OpenStreetMap Wiki* [online]. [cit. 2016-07-16]. Dostupný z: <http://wiki.openstreetmap.org/wiki/Osmconvert>.
- [17] davemux86; ludwigb; kcwu; lincomatic; borisbu. Mapsforge Map-Writer. Dostupný také z: <https://github.com/mapsforge/mapsforge/blob/master/docs/Getting-Started-Map-Writer.md>.
- [18] Butler, Howard; Daly, Martin; Doyle, Allan aj. *The GeoJSON Format Specification* [online]. [cit. 2016-07-17]. Dostupný z: <http://geojson.org/geojson-spec.html>.
- [19] Doze, Mohammed Le. *World countries in JSON, CSV, XML and YAML*. 2016. Dostupný také z: <https://github.com/mledoze/countries>.
- [20] davemux86; ludwigb; kcwu aj. *Mapsforge*. 2016. Dostupný také z: <https://github.com/mapsforge/mapsforge>.
- [21] Watson, Gray. *OrmLite - Lightweight Object Relational Mapping (ORM) Java Package*. 2016. Dostupný také z: <http://ormlite.com/>.
- [22] Android Developers. *PreferenceFragment | Android Developers* [online]. [cit. 2016-07-26]. Dostupný z: <https://developer.android.com/reference/android/preference/PreferenceFragment.html>.
- [23] Android Developers. *SharedPreferences | Android Developers* [online]. [cit. 2016-07-26]. Dostupný z: <https://developer.android.com/reference/android/content/SharedPreferences.html>.
- [24] LeBeau, Paul. *AndroidSVG*. 2016. Dostupný také z: <https://github.com/BigBadaboom/androidsvg>.
- [25] Android Developers. *Media Player | Android Developers* [online]. [cit. 2016-07-26]. Dostupný z: <https://developer.android.com/reference/android/media/MediaPlayer.html>.
- [26] Android Developers. *Supported Media Formats | Android Developers* [online]. [cit. 2016-07-26]. Dostupný z: <https://developer.android.com/guide/appendix/media-formats.html>.
- [27] anddev.org. *SoundPool crashing with "could not create track"* [online]. [cit. 2016-07-26]. Dostupný z: <http://www.anddev.org/multimedia-problems-f28/soundpool-crashing-with-could-not-create-track-t17028.html>.
- [28] Watts, Carl. *can not play ogg on android – Stack Overflow* [online]. [cit. 2016-07-26]. Dostupný z: <http://stackoverflow.com/a/36846276/4402950>.

- [29] hinnerk.oetting. *can not play ogg on android – Stack Overflow* [online]. [cit. 2016-07-26]. Dostupný z: <http://stackoverflow.com/a/19185669/4402950>.
- [30] Android Developers. *ExoPlayer | Android Developers* [online]. [cit. 2016-07-26]. Dostupný z: <https://developer.android.com/guide/topics/media/exoplayer.html>.
- [31] Wernick, Brian. *ExoMedia*. 2016. Dostupný také z: <https://github.com/brianwernick/ExoMedia>.
- [32] Wernick, Brian. *EMAudioPlayer Javadoc* [online]. [cit. 2016-07-26]. Dostupný z: <http://devbrackets.com/dev/libs/docs/exomedia/2.3.4/com/devbrackets/android/exomedia/EMAudioPlayer.html>.