

Geocaching aplikace pro mobilní zařízení

Geocaching application for mobile devices

Tomáš Šimíček

Bakalářská práce
2012



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Tomáš ŠIMÍČEK**
Osobní číslo: **A09156**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**

Téma práce: **Geocaching aplikace pro mobilní zařízení**

Zásady pro vypracování:

1. Vytvořte Windows Phone 7 aplikaci pro Geocaching s možností správy uživatelských seznamů jednotlivých cache.
2. Implementujte navigaci k souřadnicím cache pomocí kompasu.
3. Umožněte přehledný výpis seznamu cache dle uživatelského výběru s filtrací a řazení položek (velikost, obtížnost, terén, vzdálenosti, geotagy, atd.).
4. Zařídte podrobný náhled každé cache (listing, uživatelské logy, souřadnice, parametry, hint).
5. Naprogramujte komunikaci s GPS modulem se zobrazením údajů o kvalitě signálu.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. SHARP, John. Microsoft Visual C 2010: krok za krokem. Vyd. 1. Brno: Computer Press, 2010, 696 s. ISBN 978-802-5131-473.
2. PETZOLD, Charles. Programming Windows Phone 7. Redmond, WA: Microsoft Press, 2010. ISBN 978-073-5643-352.
3. SHARP, John. Microsoft silverlight edition: Programming Windows phone 7. Microsoft silverlight ed. Redmond, Washington: Microsoft Press, 2010, 792 s. ISBN 07-356-5667-3.
4. Windows Phone Development. Msdn.com [online]. December 15, 2011 [cit. 2012-02-01]. Dostupné z: [http://msdn.microsoft.com/en-us/library/ff402535\(v=vs.92\).aspx](http://msdn.microsoft.com/en-us/library/ff402535(v=vs.92).aspx)
5. Seriál C: C - Úvod. Programujte.com [online]. ? 2005-2009 [cit. 2012-02-01]. Dostupné z: <http://programujte.com/clanek/2005071001-c-uvod/>

Vedoucí bakalářské práce:

Ing. Dalibor Slovák

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

24. února 2012

Termín odevzdání bakalářské práce:

8. června 2012

Ve Zlíně dne 24. února 2012

prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Cílem této bakalářské práce je vytvořit aplikaci pro mobilní zařízení s operačním systémem Windows Phone 7, která bude pomáhat hráčům turistické hry Geocaching. Aplikace umožní uživatelům přehledně spravovat, třídit a filtrovat jejich seznamy cache, bude je navigovat k finálním GPS souřadnicím pomocí kompasu i mapy a zobrazí jim podrobný náhled jednotlivých cache se všemi potřebnými informacemi k jejímu úspěšnému nalezení.

Teoretická část práce popisuje pravidla hry Geocaching, typy cache, historii hry, dále pak navigační systém GPS s popisem jak lze vypočítat vzdálenost s azimutem a nakonec také popisuje práci s Windows Phone 7 a jeho SDK 7.1.

V praktické části je popsána samotná aplikace, použité třídy, její nejdůležitější vybrané metody a samotná práce s aplikací spolu s popisem jednotlivých aplikačních obrazovek.

Klíčová slova: Geocaching, GPS, WP7, gpx, turistika

ABSTRACT

The aim of this Bachelor thesis is to create an application for mobile devices with operating system Windows Phone 7 which will help players of hiking game Geocaching. Application will allow users to clearly manage, sort and filter their cache lists, will navigate them to final GPS coordinates by both compass and map and displays thorough view of particular caches with all necessary information leading to successful discovery.

Theoretical part of thesis describes game rules of Geocaching, its types of caches, history of a game, also describes navigation system GPS with description how to calculate distance with azimuth and finally describes work with Windows Phone 7 and its SDK 7.1.

In practical part is described application itself, used classes, most important selected methods and work with application together with description of particular application screens.

Keywords: Geocaching, GPS, WP7, gpx, hiking

Rád bych poděkoval vedoucímu mé bakalářské práce, panu Ing. Daliboru Slovákovi, za poskytnutou odbornou pomoc, rady a připomínky při tvorbě mé bakalářské práce a také mé rodině i přátelům za psychickou a morální pomoc během mé práce i samotného bakalářského studia.

„Never trust a computer you can't throw out a window.“ - Steve Wozniak

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD.....	10
I TEORETICKÁ ČÁST	11
1 ÚVOD DO GEOCACHINGU	12
1.1 CACHE.....	12
1.1.1 Obsah.....	13
1.1.2 Speciální předměty	13
1.1.3 Velikosti	13
1.1.4 Typy.....	14
1.2 HISTORIE GEOCACHINGU.....	15
2 NAVIGAČNÍ SYSTÉM GPS	16
2.1 SEGMENTY GPS	16
2.1.1 Kosmický segment	16
2.1.2 Řídící segment.....	16
2.1.3 Uživatelský segment.....	17
2.2 URČENÍ POLOHY	17
2.3 FORMÁT GPS SOUŘADNIC	18
2.3.1 Výpočet vzdálenosti	19
2.3.2 Výpočet azimutu	19
3 WINDOWS PHONE 7.....	20
3.1 WINDOWS SDK 7.1	20
3.2 NĚKTERÉ OVLÁDACÍ PRVKY	21
3.2.1 Stack Panel	21
3.2.2 Grid	21
3.2.3 Polygon.....	21
3.2.4 Pivot	22
3.2.5 Panorama.....	23
II PRAKTICKÁ ČÁST	24
4 POPIS APLIKACE	25
4.1 TŘÍDY APLIKACE	25
4.1.1 SeznamKeseK_Class	25
4.1.2 Log_Class.....	26
4.1.3 Waypoint_Class	26
4.1.4 Keska_Class	27
4.2 METODY APLIKACE	28
4.2.1 Import uživatelských seznamů cache z GPX souboru	28
4.2.2 Převod souřadnic ze stupňů na stupně a minuty.....	30
4.2.3 Aplikační nastavení	31
5 PRÁCE S APLIKACÍ.....	32

5.1	BLOKOVÉ SCHÉMA STRÁNEK APLIKACE.....	32
5.2	HLAVNÍ STRÁNKA.....	33
5.3	VYBRANÝ SEZNAM CACHE	35
5.4	MAPA	39
5.5	PŘIDÁNÍ/UPRAVENÍ WAYPOINTU	41
5.6	NASTAVENÍ PROGRAMU.....	42
5.7	NÁHLED VYBRANÉ CACHE.....	43
	ZÁVĚR	47
	ZÁVĚR V ANGLIČTINĚ.....	48
	SEZNAM POUŽITÉ LITERATURY.....	49
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	51
	SEZNAM OBRÁZKŮ	52
	SEZNAM TABULEK.....	53
	SEZNAM PŘÍLOH.....	54

ÚVOD

Cílem této bakalářské práce je vytvořit aplikaci pro mobilní zařízení se systémem Windows Phone 7, která bude pomáhat hráčům turistické hry Geocaching, jenž využívá navigačního systému GPS.

Turistická hra Geocaching se v posledních letech těší velké popularitě, propadá ji stále více a více hráčů všech věkových skupin a je hrána po celém světě. Geocaching lze přirovnat k hledání pokladu, tedy přináší hráčům pocit nadšení z nálezů něčeho, co ostatní nevidí, pocit výzvy a dodává jim patřičnou dávku nadšení ze samotného hledání pokladu či lidově řečeno lovu. Hráči s nadšením nejen luští chytře vymyšlené šifry, vedoucí k získání finálních souřadnic schránek či se pouští do jiných náročných výzev souvisejících s hledáním cache, ale hrají Geocaching kupříkladu jen pro radost. Aby se podívali na nějaké zajímavé místo, prohlédli si historické památky, dostali se na malebná místa či jen tak vyrazili s rodinou na výlet kdesi do přírody. [7] [8] [9]

Pro mobilní zařízení se systémem Windows Phone 7 existuje jen malý počet aplikací pro hráče Geocachingu, dá se říct, že skutečně použitelná je pouze ta oficiální, od Groundspeaku [9], avšak ta je určena jen pro platící hráče. Poté existuje asi tucet aplikací umožňujících navigaci k ručně zadaným souřadnicím, což je nepohodlné. Čili je tato aplikace první svého druhu, jelikož umožňuje hráči importovat libovolné seznamy, spravovat je, obsahuje všechny potřebné informace o jednotlivých cache a hráč je přehledně navigován k úspěšnému nalezení cache pomocí kompasu a mapy. Aplikace byla navržena po zevrubném otestování většiny těchto existujících aplikací, po prozkoumání jejich kladů a záporů tak, aby byla co možná nejvíce uživatelsky přívětivá a měla jednoduché a hlavně přehledné ovládání.

Teoretická část práce se zabývá popisem hry Geocaching, jejími pravidly a historií. Také je zde popsán navigační systém GPS, užívaný k jejímu hraní a mobilní operační systém Windows Phone 7, pro jehož zařízení je určena tato aplikace.

Praktická část popisuje samotnou aplikaci jak z pohledu uživatele – kdy jsou popsány jednotlivé stránky aplikace, jednotlivé funkce, způsoby práce s aplikací a vůbec celé grafické rozhraní aplikace -, tak z pohledu programátora – kde je aplikace popsána blokovým schématem a jsou popsány některé důležité metody (např. import dat z GPX souborů).

I. TEORETICKÁ ČÁST

1 ÚVOD DO GEOCACHINGU

Jak bylo již zmíněno v úvodu této práce, Geocaching je turistická hra, v níž jde o nalezení skryté schránky (cache), pomocí navigačního systému GPS. O každé cache jsou na webu známy informace o místě uložení cache (přímo souřadnice, textový popis, popřípadě nějaká šifra k vyluštění – závisí na typu cache); zajímavé údaje o místě (např. historie místa); parametry cache (schůdnost terénu, obtížnost jejího nalezení, atd.); nápověda k dohledání cache na místě finálních souřadnic, neboli *hint* a zápisy, neboli *logy*, o nalezení/nenalezení či jiné poznámky ke cache ostatních hráčů. Hráč Geocachingu se nazývá *geocacher* – česky potom lidově *geokačer*. Po nalezení cache, se hráč запиše do přiloženého *logbooku* – sešitu, listu či knihy s kolonkami pro datum, přezdívku hráče a vzkazem -, popřípadě provede výměnu předmětů v cache uložených a cache opět pečlivě ukryje a zamaskuje. Následně si запиše na webových stránkách nález - neboli *log* - tedy cache *zaloguje*.

Cache jsou obvykle umístěny na místech, která jsou nějakým způsobem zajímavá, něčím výjimečná a snaží se na tyto místa upozornit v listingu cache (uvádí v něm zajímavosti, historii a další údaje místa). Umísťují se ale také na místa velice frekventovaná - např. náměstí velkých měst, parky, školy, kostely, atd. V mnoha případech je zajímavý samotný úkol/úkoly spojené s nalezením cache (např. tzv. *multi-cache* nebo *unknown cache*).

Velkou výhodou geocachingu je jeho variabilita, že si každý hráč přizpůsobí hru sobě – někdo raději hledá cache v přírodě a spojí hledání třeba s rodinným výletem či jízdou na kole, zkrátka rekreačně; jiný zase hraje na body a snaží se nasbírat co nejvíce cache, vysbírat všechny cache v okolí jeho bydliště; další se vydává lovit extrémní cache do jeskyní, podzemí, kanalizací, leze na stožáry, vysoké stromy; jiní nejraději luští šifry, zkrátka každý dle svého gusta. Je vhodný pro všechny věkové skupiny – děti jsou rády, když po náročném výletu můžou nalézt „poklad“, starší pak mají nějaký zajímavý cíl výletu. [7] [8] [9]

1.1 Cache

Cache, neboli česky schránka, bývá většinou plastová, voděodolná krabička/kanýstr o různých velikostech. Měla by být velká tak, aby se do ní vešel alespoň logbook – kde se zapisují hráči - a pokud možno i tužka. Cache by měla být vždy označena názvem, popřípadě gc kódem (kód cache přidělený serverem geocaching.com) a jménem

zakladatele. Také by zde měly být přítomny informace pro náhodné nálezce z řad nehráčů – lidově nazývaných „mudlové“ (podle označení pro nekouzelníky ze série knih o Harry Potterovi, která získala popularitu ve stejném období jako Geocaching).

V České Republice se cache říká mezi hráči „kešky“ a hráčům „geokačeři“ či jen „kačeři“ podle podobnosti anglického slova „geocacher“ s českým „geokačer“.

1.1.1 Obsah

U cache větších velikostí bývá zvykem umisťovat nějaké předměty na výměnu pro ostatní hráče. Mění se kus za kus. Směna však není povinná. Většinou předměty mění geokačeři s malými dětmi, jelikož jsou v cache nejčastěji umístěny např. hračky z kinder vajíček, malé postavičky, různé karty s obrázky či jiné maličkosti.

1.1.2 Speciální předměty

Existují i speciální předměty, které se nemění, ale přenáší mezi cache, tzv. „travelbug“, což jsou předměty opatřené kovovou „psí“ známkou s názvem a identifikačním kódem nebo tzv. „geocoin“, neboli speciálními mincemi také s identifikačním kódem. Každý travelbug či geocoin má specifický cíl, jenž je uveden na webu nebo přímo u předmětu. Jedná se zpravidla o nějaký cestovní cíl (např. docestovat do Itálie, apod.) nebo nějaké přání vlastníka (např. cestovat jen po hradech). Travelbug/geocoin hráči nemusejí přenášet, mohou si jej pouze prohlédnout a zalogovat, že jej objevili.

1.1.3 Velikosti

Cache je možno dělit podle velikosti krabičky, v níž je uložena:

- *Micro* – velice malá krabička, která je vhodná pro velice frekventovaná místa. Bývá doplněna např. magnetem pro jednodušší uchopení na místo skrýše a bývá častokrát velice chytře maskována, aby splynula s okolím. V micro cache nebývají umístěny předměty na výměnu, ale pouze logbook. Nejčastěji se používá např. obalu od kinofilmu, od bonbónů tic-tac, léků, speciálně vyrobených malých schránek (falešných šroubů, kamenů, apod.) či jiných malých krabiček. Někdy se jim také říká „nano“, pokud se jedná o schránky velice malé, např. velikosti nehtu (tyto malé schránky, ale i jiné schránky, se dají zakoupit ve speciálních Geocaching obchodech).

- *Small* – malá krabice na potraviny. Obsahuje logbook, tužku a menší předměty na výměnu, popřípadě i travelbug či geocoin.
- *Regular* – krabice ideální velikosti, aby se do ní vešly větší předměty na výměnu, malí plyšáci, travelbugy, geocoiny, větší logbook a další předměty. Opět se využívá krabic na potraviny, nyní větších.
- *Large* – speciální cache o nadměrných velikostech, do nichž se vejdu větší předměty na výměnu, např. knihy. Používá se plastových/plechových barelů, apod., ale může se jednat i o objekty, např. garáž, opuštěný železniční vagón.

Cache bývají často uloženy v nějakých originálních podobách, např. falešný kámen, falešný šroub (uvnitř dutý, aby se do něj vešel logbook), falešná šiška či jiných zajímavých a originálních podobách, zkrátka aby cache co nejvíce splýnula s okolím místa uložení.

1.1.4 Typy

Cache jsou děleny podle jejich charakteru:

- *Tradiční cache* (česky tradička) – je umístěna přímo na souřadnicích uvedených v logbooku
- *Multi-cache* (česky multina) – hráč musí nejprve najít místo (anglicky stage) určené uvedenými souřadnicemi, kde je buď schránka s dalšími instrukcemi, nebo ho čeká nějaký úkol – spočíst schody, sloupy, zjistit letopočet z památníku, apod. -, který ho pak zavede na další místo. Míst (anglicky stages) může být více, až se hráč propracuje k finálnímu místu, kde je uložena cache. Multi-cache mají mnoho podob, ale základ postupování po jednotlivých stageích zůstává stejný.
- *Unknown cache* (česky mysterka) – v listingu jsou uvedeny souřadnice blízkého místa s tím, že finální souřadnice cache musí hráč zjistit z nějaké šifry, vyluštěním hádanky, určením ze zadaných informací, apod.
- *Earth cache* (česky earthka) – na cílových souřadnicích není uložena žádná krabička, hráč zde musí zjistit nějaké informace, vyfotografovat určité místo, apod. a odeslat zakladateli cache, aby mu byl uznán log. Cache bývají uloženy na místě s geologickou či jinou zajímavostí týkající se naší planety Země (např. na vrcholech vysokých hor).

- *Event cache* (česky event) – společenské setkání hráčů v určitý čas na určitém místě s nějakým doprovodným programem, kde dojde k výměně travelbugů či geocoinů.

Existují ještě další typy cache, např. *wherigo* (podobné multi-cache, využívá se speciálního programu s cartridge pro danou cache), *letterbox* (hráč hledá cache pomocí slovních indicií, ne souřadnic), atd., avšak ty nejsou tak časté a vycházejí či nějak rozšiřují uvedené typy. [7] [8] [9]

1.2 Historie Geocachingu

Geocaching vznikl prakticky ihned poté, co byla z navigačního systému GPS odstraněna umělá odchylka v roce 2000, což zlepšilo přesnost zařízení pro běžné civilisty z tehdejších desítek až stovek metrů na jednotky metrů. Po pár dnech dostal Američan Dave Ulmer nápad a ukryl do lesa kdesi v Oregonu schránku, uvedl její souřadnice na diskuzním fóru na webu, kde vyzval čtenáře, aby se ji pokusili najít. Přidal do ní pár drobností – dárků – a také sešit, kde se měli nálezci zapsat – tedy předchůdce logbooku. Schránka byla nalezena ještě v den ukrytí a v dalších dnech začaly vznikat po USA další schránky. Tím se zrodil základ Geocachingu, ovšem tehdy se tomuto hledání schránek říkalo „stash hunt“ a schránkám „stash“ (což lidově znamená tajná skrýš).

O pár dní později založil Mike Teague webovou stránku, která evidovala založené schránky. A jelikož je slovo *stash* v angličtině hovorové a spíše negativního významu, rozhodl se jistý Matt Stum nazvat toto hledání *Geocachingem*. V prosinci 2000 založil Jeremy Irish webovou stránku geocaching.com, kde zahrnul existující databázi cache a převzal správu listingů. Později z webu vznikla společnost Groundspeak, což je největší spravovatel cache na světě – mimo něj existují ještě menší weby, které nezaplatňují některé cache jako Groundspeak, např. opencaching.eu či terracaching.com.

Dnes je celosvětově registrováno zhruba 1 800 000 cache po celém světě a je aktivních 5 miliónů hráčů. ČR patří mezi státy, kde se Geocaching těší velké oblíbenosti – začal se u nás rozšiřovat v roce 2001, kdy byla založena první cache na českém území (*Tex-Czech*), a to ve Štramberku prvního června. Letos v květnu oslavil Geocaching 12 let. [7] [8] [9]

2 NAVIGAČNÍ SYSTÉM GPS

Je základním a klíčovým prvkem hry Geocaching. GPS (dříve nazývaný také jako *Navstar GPS*) je zkratkou *Global Positioning System* (česky globální navigační systém) vyvinutý v 70. letech 20. století armádou USA za účelem zjištění polohy kdekoliv na světě. V civilním sektoru našel větší uplatnění až po roce 2000, kdy byla ze signálu odstraněna umělá odchylka, díky níž se zpřesnilo určení pozice z desítek až stovek metrů na jednotky metrů.

Vývoj začal v roce 1973 sloučením projektů *System 621B* (určování polohy) a *Timation* (přesné určování času). V letech 1974 až 1979 byly provedeny testy na pozemních stanicích a vyroben experimentální přijímač a v roce 1979 byl rozšířen původní plán 18 družic na 24. Plná operační dostupnost byla vyhlášena v roce 1994, kdy byla na orbitě umístěna kompletní soustava 24 družic. [9] [10]

2.1 Segmenty GPS

Systém můžeme rozdělit na tři části, tzv. segmenty – *kosmický, řídicí a uživatelský*.

2.1.1 Kosmický segment

Jedná se o družice obíhající okolo Země. Původně mělo jít o 24 družic, ale dnes je využíván až mezní počet 32. Družice obíhají ve výšce 20 200 km v 6 kruhových drahách se sklonem 55° vzhledem k rovníku Země. Družice se pohybuje rychlostí 3,8 km/s (11 300km/hod), jeden oběh jí trvá 11 hodin 58 minut – zvládá tedy dva oběhy denně. Na každé oběžné dráze je umístěno 5 družic s tím, že jsou dráhy od sebe posunuty o 60° .

Každá družice je vybavena velice přesnými (10^{-13} s) *atomovými hodinami* s rubidiovým oscilátorem (dříve i cesiovým), přijímačem (pro řízení z pozemního centra, např. korekce dráhy), vysílačem (antény pro vysílání kódu v pásmu L – 2000-1000MHz; antény pro komunikaci mezi družicemi v pásmu UHF), solárními panely pro zdroj energie a dalšími přístroji, které slouží k navigaci nebo jiným speciálním úlohám.

2.1.2 Řídicí segment

Jeho úkolem je monitorovat kosmický segment (sledovat dráhy družic), zasílat povely družicím (např. korekce dráhy), provádět jejich údržbu (např. synchronizace atomových

hodin), starat se o požadovanou míru přesnosti družic a vypouštět nové či odstraňovat vysloužilé z oběžných drah.

Skládá se z *velitelství, řídicího střediska, povelových stanic a monitorovacích stanic* (umístěné rovnoměrně po obvodu Země, většinou blízko rovníku).

2.1.3 Uživatelský segment

Samotné zařízení schopné přijímat GPS signál a tedy schopné získat informaci o přesném čase a přesné poloze na Zemi. Přijímače pracují v pasivním režimu, tedy pouze přijímají signál z družic, které jsou nad obzorem a nic nevysílají. To proto, aby nebylo možné zaměřit GPS přijímače nepřitelem a byl systém schopen obsloužit neomezený počet uživatelů.

Uživatelé se pak dělí do dvou skupin:

- *Autorizovaní* – vojenský sektor USA a vybrané spojenecké armády mají zaručenou vyšší přesnost systému a využívají GPS převážně pro podporu vojáků v poli, navádění zbraní, dopravu, vojenskou geodézii a přesný čas.
- *Ostatní* – civilní sektor využívající GPS pro dopravu, geologii, geofyziku, geodézii, zemědělství, turistiku, atd.

2.2 Určení polohy

Každá z družic vysílá informace o přesném čase z jejich atomových hodin, o své poloze a o přibližné poloze ostatních družic. Zařízení vybavené GPS přijímačem využívá pro výpočet polohy časového rozdílu mezi vysláním dat družicí a jejich přijutím přijímačem. Aby mohlo výpočet provést, potřebuje zařízení znát přesný čas – nepotřebuje tak přesné hodiny jako jsou atomové -, který se při načítání informací z družic aktualizuje. Chyba určení polohy narůstá s nepřesností času, takže pokud by byl čas rozdílný jen o jednu tisícinu sekundy, poloha by se lišila o řádově téměř stovky kilometrů.

Aby zařízení určilo plnou *3D polohu* – tedy šířku, délku a nadmořskou výšku -, musí mít signál alespoň ze čtyř družic s tím, že čím více družic, tím přesněji může zařízení určit polohu. Pokud získá signály jen ze tří družic, může určit pouze tzv. *2D polohu*, neboli pouze zeměpisnou šířku a délku.

Po zapnutí zařízení vybaveného GPS přijímačem až do zobrazení aktuální polohy uplyne několik sekund až minut. To proto, že nejprve je nutné načíst informace o jednotlivých družicích (údaje o jejich údajných polohách, tzv. *efemeridy*) a další data (tzv. *almanach*) – tento proces se nazývá *inicializace*.

Almanach je soubor parametrů drah družic a informací o jejich zdravotním stavu. Nese informace o všech družicích a je součástí signálu vysílaného každou družicí. Je použit pro vyhledání všech viditelných družic v daném místě umístění GPS přijímače. Je velký 37 500 bitů, takže by jej při rychlosti stahování 50 b/s trvalo stáhnout do zařízení přibližně 12 a půl minuty. Není ovšem nutné jej stahovat celý, zařízení postačí jistá část. Samotná rychlost stahování - a tedy příjmu signálu od družice – je ovlivněna prostředím, kde se nachází GPS přijímač. Například ve městě v úzkých ulicích krytých vysokými budovami či hustém lese je příjem signálu značně zpomalen, oproti louce či vrcholu hory/kopce.

GPS přijímač komunikuje se zařízením (telefon, počítač, turistická/auto GPS navigace, atd.) pomocí protokolu *NMEA*, který předává zařízení informace o čase, poloze GPS, poloze družic, azimutu, rychlosti, počtu aktivních družic, atd. Protokol pracuje v textovém formátu. [9] [10]

2.3 Formát GPS souřadnic

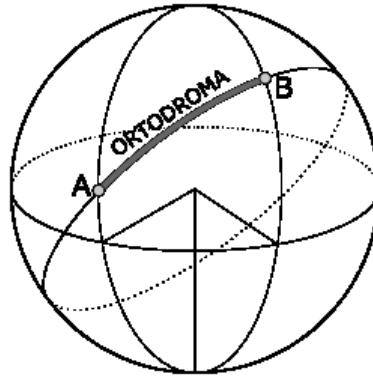
Souřadnice se skládá ze dvou údajů – zeměpisné šířky (anglicky *latitude*) pro severní (N) či jižní (S) polokouli a zeměpisné délky (anglicky *longitude*) pro západní (W) či východní (E) polokouli. V Geocachingu – a v jiných GPS navigacích - se používá systému zápisu souřadnic *WGS-84*, což je systém založený na elipsoidu nahrazujícím Zemi.

Souřadnice je možno udávat pouze ve *stupních*[°] (např. 64,12[°]), ve *stupních*[°] a *minutách*[′] (např. 32[°] 16[′]) či ve *stupních*[°], *minutách*[′] a *vteřinách*[″] (např. 32[°] 16[′] 8[″]). V Geocachingu se nejčastěji využívá zápisu druhého, tedy *stupně a minuty*[′].

Mezi těmito formáty souřadnic lze pak lehce převádět. Např. ze souřadnic udaných ve stupních 64,10[°] lze získat souřadnice ve stupních a minutách 64[°]6[′] – minuty se získaly vynásobením čísla za desetinnou čárkou u stupňů šedesáti. Opačně by se pak minuty vydělily šedesáti a přičetly ke stupňům. [8]

2.3.1 Výpočet vzdálenosti

Vzdálenost mezi aktuální pozicí (označme jako A) a nějakou jinou pozicí na Zemi (B) - např. souřadnicemi cache -, se vypočítá pomocí *ortodomy* (Obr.1), což je nejkratší spojnice dvou bodů na kulové ploše (tedy povrchu Země).



Obr. 1 – ortodoma

Ortodoma se vypočítá podle vzorce (1), kde zeměpisná šířka je označena jako φ a délka jako λ . [12]

$$d = \arccos(\sin(\varphi_1) * \sin(\varphi_2) + \cos(\varphi_1) * \cos(\varphi_2) * \cos(\lambda_2 - \lambda_1)) * 6372,795 \quad (1)$$

2.3.2 Výpočet azimutu

Azimut se vypočte dle vzorce (2), kde je opět zeměpisná šířka označena jako φ a délka jako λ .

$$\alpha = \arctan\left(\frac{\sin(\lambda_2 - \lambda_1) * \cos(\varphi_2)}{\cos(\varphi_1) * \sin(\varphi_2) - \sin(\varphi_1) * \cos(\varphi_2) * \cos(\lambda_2 - \lambda_1)}\right) \quad (2)$$

Pokud je zeměpisná šířka v bodě A větší nebo rovna šířce v bodě B, tedy pokud $\varphi_1 \geq \varphi_2$, přičteme k azimutu 360° . [12]

3 WINDOWS PHONE 7

Jedná se o nový operační systém pro mobilní zařízení od Microsoftu. Nahrazuje svého předchůdce Windows Mobile, jenž byl na trhu už od dob prvních PDA, a tedy nebyl příliš vhodný pro dnešní éru dotykových telefonů – byl primárně navržen pro ovládání stylusem, ne prstem a jeho ovládací prvky byly velice podobné těm ze stolního počítače. Nový OS WP7 byl však od samého začátku navržen pro dotykové mobilní zařízení, což znamená, že zavádí určité nové ovládací prvky (viz. dále)

Zařízení s OS WP7 mají přesně danou minimální hardwarovou konfiguraci, kterou musí výrobci dodržet (např. požadavkem je jednotné rozlišení displaye 800x480), což vede k maximální kompatibilitě, bezproblémovosti a svižnosti běhu aplikací. Aplikace se sdružují v centrálním obchodě, tzv. Marketplace, kde je možno je hodnotit, psát k nim recenze a uživatel hned vidí, co ta která aplikace dělá, vidí fotky z jejího běhu, takže si může ihned udělat jasný obrázek v hlavě o aplikaci - což ještě podporuje možnost aplikaci vyzkoušet před koupí - a nekupuje tak zajíce v pytli. Aplikace jsou v obchodě seřazeny dle kategorií a hodnocení, čímž je jejich přehlednost ještě zvýšena.

Aplikace na OS WP7 běží v tzv. *sandboxu*, tedy jakémisi virtuálním „pískovišti“, neboli každá aplikace má svůj vlastní prostor, do něhož nemá jiná aplikace přístup a tato aplikace zase nemá přístup kdekoli jinde. To zvyšuje stabilitu a rychlost zařízení, což v předchozích zařízeních s Windows Mobile nebylo pravidlem a zařízení s tímto systémem se musela velice často restartovat, trpěla záseky či jinými problémy. Ač má toto řešení sandboxu nesporné výhody, přináší to i nevýhody. Například již není možné prohlížet souborovou strukturu zařízení (pouze po odemčení zařízení a instalaci neoficiálního programu) a je nutno multimediální obsah nahrávat do zařízení přes oficiální program *Zune*. Na druhou stranu je tak zvýšena bezpečnost a stabilita zařízení. [2] [3]

3.1 Windows SDK 7.1

Aplikace se vyvíjejí v prostředí *Microsoft Visual Studio 2010* určeném pro WP7, a to v jazycích patřících do *.NET Compact Framework* (tedy C# a Visual Basic) a technologie Silverlight, která využívá značkovacího jazyka *XAML*.

XAML je značkový jazyk – něco jako např. HTML -, který je založen na XML a je využíván k popisu grafického rozhraní v aplikacích. Vše napsané v XAMLu, je možné také

napsat v C# - či jiných standardních .NET jazyků – avšak výhodou XAMLu je jeho přehlednost a jednoduchost. XAML soubor je zkompileován do binárního XAML souboru (*.baml*) a použit jako resource, takže si aplikace za běhu z něj vezme informace a vykreslí grafické rozhraní jím popsané. XAML může popisovat kromě uživatelského vzhledu aplikace, také různé animace či vazby na data - tzv. *binding*, kdy určitému prvku přiřadí programátor jako hodnotu např. celou třídu a ta je použita jako zdroj dat pro daný prvek; tedy kupříkladu pro *listbox*, kdy budou jednotlivými položkami prvky třídy; může se jednat také o dvoucestné bindování, kdy se po změně nabindované hodnoty v prvku změní i hodnota uložená v nabindované třídě. [2] [3] [12]

3.2 Některé ovládací prvky

Kromě již standardních ovládacích prvků – např. *button*, *listbox*, *radiobutton*, *checkbox*, *image*, *atd.* – obsahuje WP7 nějaké nové prvky [4] [12].

Zde jsou popsány především prvky nejčastěji použité v aplikaci:

3.2.1 Stack Panel

Prvek umožňující zobrazit další prvky horizontálně či vertikálně seřazeny. Je možno mu nastavit samozřejmě nějaké pevné rozměry, obtékání, okraje, *atd.* [4]

3.2.2 Grid

Prvek skládající se ze sloupců a řádků, v nichž je možno zobrazit další prvky – tedy v podstatě obdoba tabulky. Je možno nastavit každé buňce různé rozměry – napevno, automaticky, sloučením více buněk, *atd.* [4]

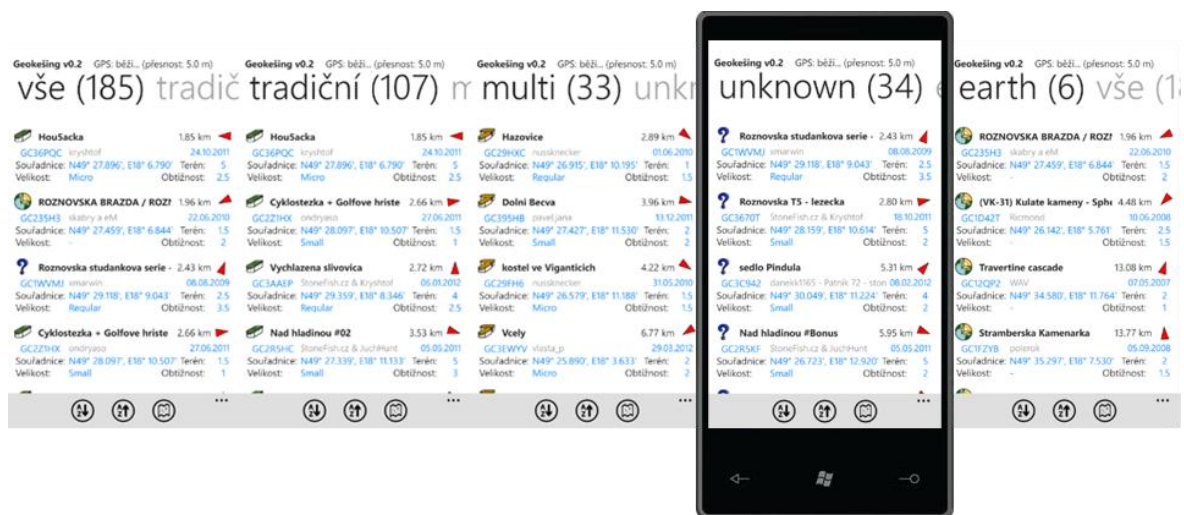
3.2.3 Polygon

Slouží k vykreslení tvaru pomocí série bodů spojených čarou, které vytvářejí uzavřený geometrický útvar – v aplikaci je využíván pro vykreslení šipky kompasu a šipek ukazujících od aktuální pozice k souřadnicím cache. Je možné u něj nastavit libovolnou barvu výplně, barvu a tloušťku obrysu, průhlednost, efekty (*pixel shader*, *atd.*), způsoby vyplnění barvy a mnoho dalších. [4]

3.2.4 Pivot

Horizontální prvek obsahující vertikální „scrollovatelný“ obsah („rolovatelný“, s možností postupně listovat dolů stránkou). Jde v podstatě o prvek, který má v sobě více rámců – něco jako záložky. Nabízí možnost rychle spravovat/prohlížet jednotlivé rámce, filtrovat/zobrazovat velké množství dat nebo měnit různé rámce programu. Příklad lze spatřit na Obr. 2.

Základem prvku je kontejner, který obsahuje prvky PivotItem, v nichž může programátor umístit jakýkoliv jiný prvek, např. tlačítka, tabulky, obrázky, odkazy, atd. Mezi PivotItemy se přesouvá pomocí kliknutí na název prvku/itemu nebo gestem „švenku“, tedy pohybem prstu zprava doleva či naopak. [4]



Obr. 2 – příklad prvku *pivot*

3.2.5 Panorama

Jedná se o podobný horizontální prvek umožňující vertikálně „scrollovat“ obrazovku (podobně jako *pivot*). Jeho hlavní vlastností a předností je simulace jedné velké horizontální plochy – v podstatě spojí více obrazovek vedle sebe do jedné velké s nadpisem přese všechny tyto obrazovky (tedy rámce) – lépe je to vidět na (Obr.3).

Je možné umístit na pozadí panorama prvku velký širokoúhlý obrázek, jenž se bude postupně scrollovat podle aktuálně vybraného rámce. Doporučuje se použít maximálně čtyř rámců v jednom panorama prvku, kvůli přehlednosti. [4]



Obr. 3 – příklad prvku *panorama*

II. PRAKTICKÁ ČÁST

4 POPIS APLIKACE

Cílem této bakalářské práce bylo vytvořit aplikaci pro mobilní zařízení, specificky pro zařízení s operačním systémem Windows Phone 7. Aplikace slouží hráčům Geocachingu k zobrazování různých jejich seznamů cache - jenž jim umožní importovat ze souboru gpx -, přehledné možnosti filtrování jednotlivých cache v seznamech dle různých parametrů (vzdálenost, názvu, velikosti, obtížnosti, atd.), jejich zobrazení na mapě a následnou navigaci až k samotné cache. A to tak, že zobrazí hráči přehledně všechny potřebné informace o cache – tedy plný listing, hint, logy uživatelů, waypointy cache, parametry cache a také umožní hráči přidat vlastní body zájmu (například pro luštění unknown cache či různých zajímavých míst v okolí cache). Navigace k zvolenému waypointu cache - ať už finálním souřadnicím či nějakému bodu zájmu – probíhá pomocí digitálního kompasu v telefonu (nebo při jeho absenci vypočteného směru GPS přijímačem), jenž ukazuje směr k severu a zvolenému waypointu nebo pomocí mapy – možnost leteckých či klasických mapových podkladů od Microsoft Bing -, na níž zobrazí všechny waypointy související s cache (na mapě zobrazí také přesnost GPS či již ušlou trasu).

4.1 Třídy aplikace

4.1.1 SeznamKesek_Class

Jedná se o třídu popisující jednotlivé uživatelské seznamy. Ukládá název seznamu, kdy byl seznam importován do aplikace, kolik obsahuje cache jednotlivých typů (tradičních, multi, unknown či earth) a samotnou kolekci s cache (ty jsou uloženy ve třídě *Keska_Class*).

Tab. 1 – proměnné třídy *SeznamKesek_Class*

<i>Dědičnost</i>	<i>Typ</i>	<i>Název</i>	<i>Popis</i>
private	string	_navez	název seznamu
private	DateTime	_datum	datum vytvoření seznamu
private	uint	_tradicni	počet tradičních cache
private	uint	_multi	počet multi cache
private	uint	_unknown	počet unknown cache
private	uint	_earth	počet earth cache
private	ObservableCollection<Keska_Class>	_kesky	kolekce cache seznamu

Třída využívá rozhraní (anglicky *Interface*) *INotifyPropertyChanged*, které se stará o upozornění, že byla změněna hodnota klienta – toho se využívá převážně u dvoucestného bindování (viz. kapitola 3.1). Takže má třída proměnné s dědičností *private* (Tabulka 1),

přičemž má každá proměnná svou public metodu, která se stará o *set* (přiřazení hodnoty) i *get* (získání hodnoty) soukromé proměnné. Tato metoda při vykonávání *set* zavolá metodu *NotifyPropertyChanged*, což upozorní rozhraní, že se proměnná změnila – např. uživatel edituje hodnotu názvu seznamu v TextBoxu, což vyvolá akci *set* v dané metodě, jelikož má TextBox nastaveno pomocí XAMLu dvoucestné bindování a uloží se nový název do příslušné instance třídy.

4.1.2 Log_Class

Třída popisuje uživatelské logy u jednotlivých cache. Ukládá přezdívku hráče, který log uložil, typ logu (nalezl cache, nenalezl cache, provedl údržbu cache, napsal poznámku, zobrazil cache, skryl cache, archivoval cache, atd.), čas provedení logu a samotný text logu.

Tab. 2 - proměnné třídy *Log_Class*

<i>Dědičnost</i>	<i>Typ</i>	<i>Název</i>	<i>Popis</i>
private	string	_uzivatel	přezdívka hráče
private	string	_typlogu	typ logu
private	DateTime	_caslogu	čas uložení logu
private	string	_textlogu	text logu

Třída obsahuje metodu *typlogu_image*, která slouží k převodu typu logu na obrázek – tedy aby se místo např. „found it“ zobrazil usměvavý „smajlík“.

4.1.3 Waypoint_Class

Tato třída popisuje jednotlivé waypointy cache. Její proměnné ukládají informace o názvu, typu (finální souřadnice, parkoviště, nějaká otázka, stage multi cache, atd.), jedinečném kódu waypointu, textový popis o jaký waypoint se jedná, jeho vzdálenost od aktuální pozice GPS, azimut k němu a jeho GPS souřadnice.

Tab. 3 – proměnné třídy *Waypoint_Class*

<i>Dědičnost</i>	<i>Typ</i>	<i>Název</i>	<i>Popis</i>
private	string	_nazevwaypointu	název waypointu
private	string	_typwaypointu	typ waypointu
private	string	_kodwaypointu	jedinečný kód waypointu
private	string	_popiswaypointu	textový popis waypointu
private	double	_vzdalenost	vzdálenost od akt. pozice
private	double	_azimut	azimut k němu
private	GeoCoordinate	_souradnicewaypointu	jeho souřadnice

Třída opět mimo jiné obsahuje rozhraní *INotifyPropertyChanged* (viz. popis třídy *SeznamKesek_Class*). Dále pak obsahuje metodu *typ_image*, která se stará podobně jako metoda *typlogu_image* u *Log_Class* o převod typu waypointu na obrázek; metodu pro převod souřadnic z formátu ve stupních na stupně a minuty (*souradnicewaypointu_stupneminuty*) a metodu pro výpis vzdálenosti waypointu v kilometrech nebo metrech podle potřeby (*vzdalenost_pekne*).

4.1.4 Keska_Class

Nejdůležitější metoda aplikace ukládá všechny potřebné informace o cache. Její název, GC kód, přezdívku zakladatele, datum založení, typ, hint, velikost, obtížnost, terén, plný listing, vzdálenost od aktuální GPS pozice, azimut, zdali ji má už uživatel nalezenou, jestli je cache k dispozici, její ID na serveru geocaching.com, finální souřadnice (pokud jsou uvedeny, např. u tradičních cache), kolekci s logy (třída *Log_Class*), kolekci s waypointy (třída *Waypoint_Class*) a informaci, který waypoint je aktivní.

Tab. 4 – proměnné třídy *Keska_Class*

Dědičnost	Typ	Název	Popis
private	string	_navez	název cache
private	string	_gckod	GC kód cache
private	string	_zakladatel	přezdívka zakladatele
private	string	_typkesky	typ cache
private	string	_hint	nápověda k nalezení
private	string	_idkesky	ID cache
private	string	_listing	plný listing
private	string	_velikost	velikost schránky
private	float	_teren	obtížnost terénu
private	float	_obtiznost	obtížnost ukrytí
private	double	_vzdalenost	vzdálenost od akt. Pozice
private	double	_azimut	azimut ke cache
private	bool	_kdispozici	cache je k dispozici
private	bool	_nalezena	cache je už nalezena
private	GeoCoordinate	_souradnice	souřadnice cache
private	GeoCoordinate	_aktivniwaypoint	aktivní waypoint
private	DateTime	_zalozena	datum založení
private	ObservableCollection<Log_Class>	_logy	logy hráčů
private	ObservableCollection<Waypoint_Class>	_waypointy	waypointy cache

Opět je využito rozhraní *INotifyPropertyChanged* a metody pro převod typu cache na obrázek. Dále obsahuje třída metody pro převod slovně udané velikosti schránky na číslo –

kvůli možnosti řadit cache dle velikosti – (*velikost_cislo*); metoda pro převod souřadnic ze stupňů na stupně a minuty jak pro výchozí souřadnice (*souradnice_stupneminuty*), tak pro souřadnice aktivního waypointu (*aktivniwaypoint_stupneminuty*) a metodu vypisující vzdálenost v kilometrech či metrech (*vzdalenost_pekne*).

4.2 Metody aplikace

Následuje popis několika vybraných důležitých metod aplikace s náhledem zdrojových kódů:

4.2.1 Import uživatelských seznamů cache z GPX souboru

Import jednotlivých uživatelských seznamů cache ze souborů gpx je řešen pomocí metody *vyparsujGPX(string cesta)*.

Tato metoda načte stažený soubor gpx z *IsolatedStorage* (dále IS) umístěný na převzaté cestě z parametru metody a vyparsuje (tzn. postupnou analýzou prvků určí jejich strukturu) z něj - pomocí *LINQ* (jeho popis viz. [6]) - potřebné informace o cache, které uloží do třídy instance *Keska_Class*, a to pro všechny cache z gpx souboru - tedy využije kolekce *ObservableCollection<Keska_Class>*. Toto se provádí pouze při stažení nového souboru gpx se seznamem cache, jinak program už pracuje pouze s aktuální instancí třídy *SeznamKesek_Class*, kterou průběžně ukládá (např. při zavření aplikace, při změně nějakého údaje o cachei, atd.) pomocí *serializace* (metoda *serializujAktSeznam()*) a opětovně načítá (např. při otevření aplikace) pomocí *deserializace* (metoda *nactiSeznamyZIS()*).

Ukázka gpx souboru s uživatelským seznamem cache (vytvořeném v programu *GeoGet*), který v sobě ukrývá informace o jedné cachei s jedním logem hráče a žádným dodatečným waypointem (jedná se o tradiční cache, takže jsou finální souřadnice známy):

```
<?xml version="1.0" encoding="utf-8"?>
<gpx xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="1.0"
creator="Groundspeak Pocket Query"
xsi:schemaLocation="http://www.topografix.com/GPX/1/0
http://www.topografix.com/GPX/1/0/gpx.xsd
http://www.groundspeak.com/cache/1/0/1
http://www.groundspeak.com/cache/1/0/1/cache.xsd"
xmlns="http://www.topografix.com/GPX/1/0">
  <desc>Geocache file (HasChildren)</desc>
  <author>GeoGet 2.6.4.671</author>
  <time>2012-04-03T11:22:20.922</time>
```

```

<keywords>cache, geocache, groundspcak</keywords>

<wpt lat="49.441267" lon="18.0948">
  <time>2011-05-26T00:00:00.000</time>
  <name>GC2WXQK</name>
  <desc>Kostel Vidce by Lumikja (1.5/1.5)</desc>
  <url>http://www.geocaching.com/seek/cache_details.aspx?guid=74c68ca9-
ea3f-4374-9e7b-53c8366a327c</url>
  <urlname>Kostel Vidce</urlname>
  <sym>Geocache</sym>
  <type>Geocache|Traditional Cache</type>

  <groundspcak:cache id="2267919" available="True" archived="False"
xmlns:groundspcak="http://www.groundspcak.com/cache/1/0/1">

    <groundspcak:name>Kostel Vidce</groundspcak:name>
    <groundspcak:placed_by>Lumikja</groundspcak:placed_by>
    <groundspcak:owner id="3219721">Lumikja</groundspcak:owner>
    <groundspcak:type>Traditional Cache</groundspcak:type>
    <groundspcak:container>Micro</groundspcak:container>
    <groundspcak:difficulty>1.5</groundspcak:difficulty>
    <groundspcak:terrain>1.5</groundspcak:terrain>
    <groundspcak:country>Czech Republic</groundspcak:country>
    <groundspcak:state>Zlinsky kraj</groundspcak:state>
    <groundspcak:long_description html="True">Obec Vidce a kulturní
památky<br><br>Obec Vidce patří mezi ...</groundspcak:long_description>
    <groundspcak:encoded_hints>Kudy kam, dáti se mám? Tak se na to
podívám.</groundspcak:encoded_hints>
    <groundspcak:logs>
      <groundspcak:log id="201365637">
        <groundspcak:date>2011-11-29T00:00:00.000</groundspcak:date>
        <groundspcak:type>Found it</groundspcak:type>
        <groundspcak:finder id="1294954">skabry a eM</groundspcak:finder>
        <groundspcak:text encoded="False">Keška s pekným výhledem na kostel,
jen kdyby nebyla zima lidi necekali na autobus:-)) chvilku jsem pockal,
odlovil a vyrazil dál na kole. Díky</groundspcak:text>
        </groundspcak:log>
      </groundspcak:logs>

    </groundspcak:cache>

  </wpt>

</gpx>

```

Metoda nejprve načte dříve stažený gpx soubor z IS:

```

XDocument loadedData;
using (IsolatedStorageFile IsolatedStorage =
IsolatedStorageFile.GetUserStoreForApplication())
{
    using (IsolatedStorageFileStream stream = IsolatedStorage.OpenFile(cesta,
 FileMode.Open))
    {
        using (XmlReader xmlReader = XmlReader.Create(stream))
        {
            loadedData = XDocument.Load(xmlReader);
        }
    }
}

```

Poté nastaví správné *namespace* (jmenné prostory gpx souboru):

```

XNamespace gpxNS = XNamespace.Get("http://www.topografix.com/GPX/1/0");

```

XNamespace groundspeakNS = XNamespace.Get("http://www.groundspeak.com/cache/1/0/1");

A následuje samotné načítání parametrů jednotlivých cache pomocí LINQu (zdrojový kód zkrácen):

```
return (from query in loadedData.Descendants(gpxNS + "wpt")
        let groundspeak = query.Descendants(groundspeakNS + "cache")
        where !query.Element(gpxNS + "type").Value.Contains("Waypoint")
        select new Keska_Class
        {
            gckod = query.Element(gpxNS + "name").Value,
            souradnice = new GeoCoordinate( PomFce.PrevedNaDouble(
                query.Attribute("lat").Value),
                PomFce.PrevedNaDouble(query.Attribute("lon").Value)),
            zalozena = (DateTime)query.Element(gpxNS + "time"),
            kdispozici = (bool)groundspeak.Attributes("available").First(),
            idkesky = groundspeak.Attributes("id").First().Value,
            ...
            ...

            logy = (from queryl in groundspeak.Elements(
                groundspeakNS + "logs").Descendants(groundspeakNS + "log")
                select new Log_Class
                {
                    uzivatel = queryl.Elements(groundspeakNS +
                        "finder").First().Value,
                    typlogu = queryl.Elements(groundspeakNS +
                        "type").First().Value,
                    caslogu = (DateTime)queryl.Elements(groundspeakNS +
                        "date").First(),
                    textlogu = queryl.Elements(groundspeakNS +
                        "text").First().Value
                }).ToObservable<Log_Class>(),

            waypointy = (from queryw in loadedData.Descendants(gpxNS + "wpt")
                where queryw.Element(gpxNS + "type").Value.Contains("Waypoint")
                    && queryw.Element(gpxNS + "name").Value.Substring(2) ==
                    query.Element(gpxNS + "name").Value.Substring(2)
                select new Waypoint_Class
                {
                    nazevwaypointu = queryw.Element(gpxNS + "desc").Value,
                    kodwaypointu = queryw.Element(gpxNS + "name").Value,
                    ...
                    ...
                    typwaypointu = queryw.Element(gpxNS + "sym").Value,
                    popiswaypointu = queryw.Element(gpxNS + "cmt").Value
                }).ToObservable<Waypoint_Class>()
        }).ToObservable<Keska_Class>());
```

4.2.2 Převod souřadnic ze stupňů na stupně a minuty

Tato metoda – *StupneNaStupneMinuty(double sirka, double delka)* - se nachází v pomocné statické třídě PomFce, kde jsou umístěny ještě další pomocné metody, např. pro výpočet azimutu či převodu na double.

Jejími parametry jsou zeměpisná šířka a délka a metoda vrací formátovaný string se souřadnicemi ve formátu stupně a minuty. Její zdrojový kód:

```
return (sirka > 0 ? "N" : "S") + Math.Floor(sirka)
    + "° "
    + ((sirka - Math.Floor(sirka)) * 60).ToString("0.000")
    + "' , "
    + (delka > 0 ? "E" : "W")
    + Math.Floor(delka) + "° "
    + ((delka - Math.Floor(delka)) * 60).ToString("0.000") + "'";
```

4.2.3 Aplikační nastavení

K ukládání či změně aplikačního nastavení, je využita metoda *PridejNeboZmen(string klic, Object hodnota)*, která přistupuje do instance *ApplicationSettings* třídy *IsolatedStorageSettings* (dává k dispozici slovník s klíči a hodnotami, tedy *Dictionary<TKey, TValue>*), kde uloží nový klíč se zadanou hodnotou nebo hodnotu stávajícího klíče změní. Zdrojový kód metody:

```
//existuje?
if (IsolatedStorageSettings.ApplicationSettings.Contains(klic))
{
    //nastala změna?
    if (IsolatedStorageSettings.ApplicationSettings[klic] != hodnota)
    {
        //ulož novou hodnotu
        IsolatedStorageSettings.ApplicationSettings[klic] = hodnota;
    }
}
//neexistuje, přidej nový
else
{
    IsolatedStorageSettings.ApplicationSettings.Add(klic, hodnota);
}
//ulož
IsolatedStorageSettings.ApplicationSettings.Save();
```

Metoda je využívána, jak už bylo zmíněno, k nastavení aplikace, tedy aby uživatelem zvolené nastavení např. přesnosti GPS bylo po opětovném spuštění úspěšně načteno - samotné načítání nastavení z *IsolatedStorageSettings* se provádí pomocí kódu např.:

```
IsolatedStorageSettings nastaveni = IsolatedStorageSettings.ApplicationSettings;
if (nastaveni.Contains("presnost"))
    gps = new GeoCoordinateWatcher((GeoPositionAccuracy)nastaveni["presnost"]);
```

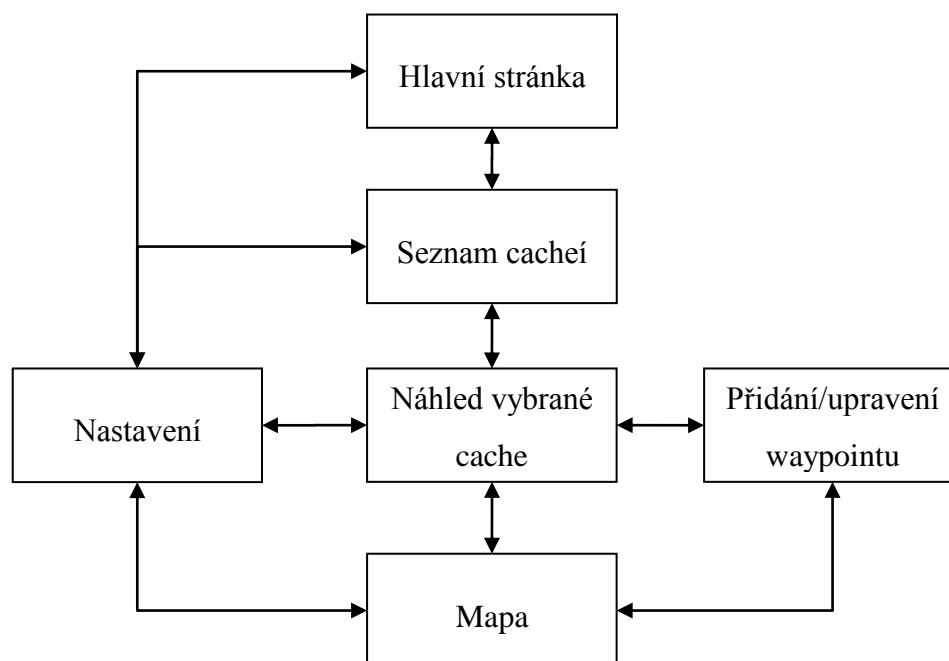
5 PRÁCE S APLIKACÍ

Aplikace je rozdělena do jednotlivých stránek (anglicky *pages*), které mohou obsahovat více rámců (anglicky *frames*), v nichž může být nespočet různých ovládacích prvků (*textblock*, *textbox*, *listbox*, *pivot*, *stackpanel*, *radiobutton*, *toggleswitch*, atd.).

Každá stránka pak může mít menu (anglicky *application menu*), které je umístěno ve spodním rohu obrazovky telefonu. Zde jsou k dispozici akční tlačítka s přehlednou ikonou – pro nejčastější akce, např. zapnutí/vypnutí GPS, přidání waypointu, atd. - a po otevření menu (pomocí tří teček v pravé části menu) také textová tlačítka – ty slouží pro méně časté akce, např. pro přechod do nastavení programu.

Pohyb mezi stránkami aplikace je pak řízen pomocí programového navigačního rozhraní – něco na způsob odkazů na webových stránkách -, kdy se dá tato akce přiřadit libovolnému prvku (tlačítko, položka menu, obrázek, atd.) a k návratu na předchozí stránku většinou slouží hardwarové tlačítko telefonu zpět (tato přednastavená akce se však dá přetížít a upravit dle libosti; např. pro zobrazení informace zdali chce uživatel opravdu ukončit program).

5.1 Blokové schéma stránek aplikace



Obr. 4 – blokové schéma stránek aplikace

5.2 Hlavní stránka

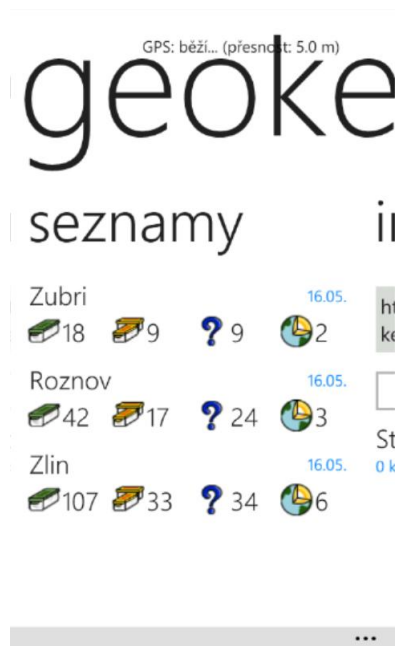
Při spouštění aplikace je zobrazena načítací obrazovka (Obr. 5), kde je vidět logo aplikace složené z oficiálního loga Geocaching.com [9] a Geocaching.cz [8].



©2012 Tomáš Šimíček

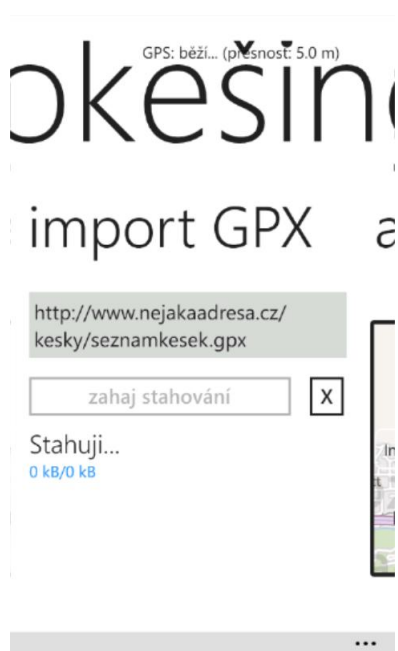
Obr. 5 – načítací obrazovka aplikace

Po úspěšném načtení se zobrazí hlavní stránka aplikace (*MainPage.xaml*), která je tvořena prvkem *panorama* (viz. popis *panorama* prvku v kapitole 3.2.4, kde lze vidět na Obr. 3 celkový pohled na hlavní stránku) jenž je složen ze tří rámců. V horní části panorama prvku vidíme název programu psaný velkým písmem, které je přes všechny rámce a nad ním malým písmem aktuální stav GPS se zobrazením aktuální přesnosti signálu. Dále lze dole na stránce vidět skryté menu aplikace, které uživatel vyvolá stiskem tří teček v pravém spodním okraji a které je zobrazeno ve všech třech rámcích hlavní strany.



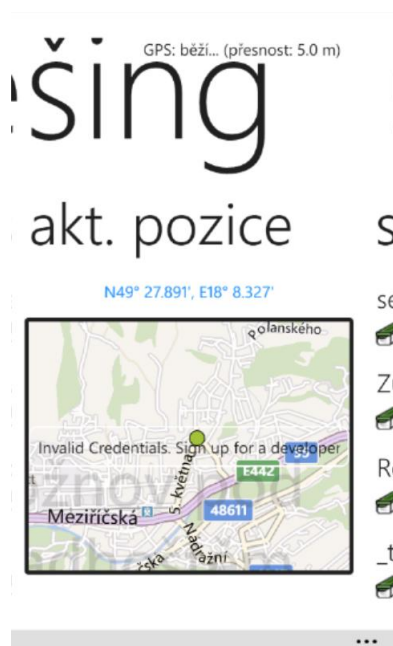
Obr. 6 – hlavní stránka, náhled uživatelských seznamů

V prvním a výchozím rámci (Obr. 6) můžeme vidět náhled jednotlivých uživatelských seznamů cache (prvek *listbox*). Ten sestává z názvu seznamu, data přidání seznamu a počtu jednotlivých typů cache, které seznam obsahuje. Ty jsou znázorněny ikonami typů (viz. přehled typů cache v kapitole 1.1.4). Pokud uživatel podrží dvě vteřiny prst na některém ze seznamů, zobrazí se kontextové menu, jenž mu umožní daný seznam přejmenovat nebo smazat. Klikem na seznam poté uživatel přejde na jeho náhled (stránka *SeznamKesi.xaml*).



Obr. 7 – hlavní stránka, import uživatelských seznamů

Druhý rámeček (Obr. 7) ukrývá možnost importovat uživatelské seznamy cache z internetu (jelikož, jak již bylo dříve řečeno, WP7 nepodporuje USB Mass Storage (protokol pro komunikaci s externími zařízeními pomocí sběrnice USB), čili je toto – nebo přes *SkyDrive* (online úložiště dat) - jediný rozumný způsob jak do telefonu zkopírovat libovolný soubor) pomocí souboru gpx ve formátu *PocketQuery* (což je standardní formát gpx souboru na oficiální stránce Geocachingu [9]). Uživatel zadá webovou adresu požadovaného seznamu nebo jedné cache a po kliknutí na tlačítko *zahaj stahování*, se daný soubor gpx stáhne do *IsolatedStorage* programu, kdy je stahování možno kdykoliv zrušit tlačítkem X. Samotný průběh stahování se zobrazuje pod tlačítky.



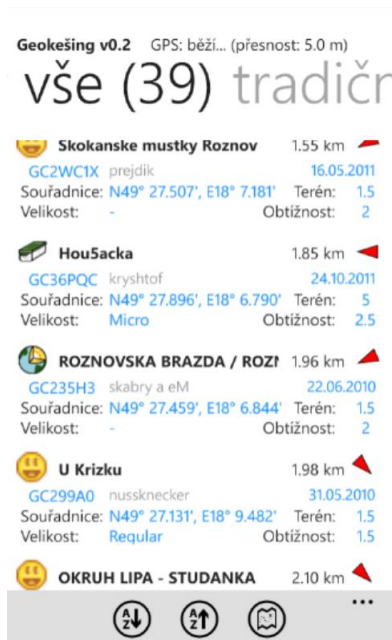
Obr. 8 - hlavní stránka, zobrazení aktuální pozice

Třetí a poslední rámeček (Obr. 8) hlavní stránky aplikace obsahuje zobrazení aktuální pozice uživatele na mapě a textově ve formátu GPS souřadnic stupně° minuty'. Mapa je pasivní, tedy nepřijímá uživatelský vstup a neumožňuje přiblížení/oddálení, protože slouží pouze jako indikátor kde přibližně uživatel nyní je.

5.3 Vybraný seznam cache

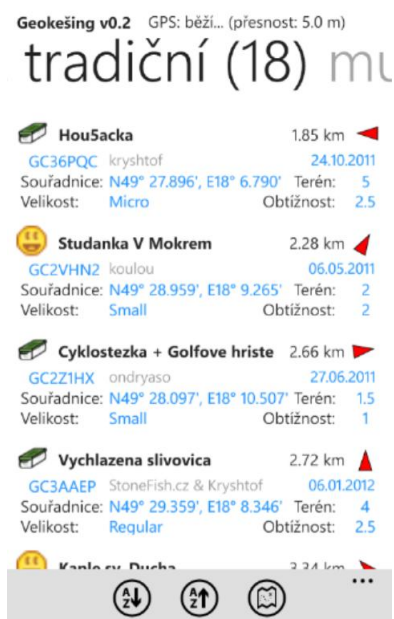
Po výběru nějakého uživatelského seznamu cache na hlavní stránce, se uživatel ocitne na obrazovce daného vybraného seznamu. Ten je tvořen prvkem pivot (viz. kapitola 3.2.3), který má dynamicky plněné rámečky dle typu cache. Tedy pokud jsou v seznamu cache typu např. *unknown*, má pivot rámeček *unknown*, – vždy je zobrazen pivot se všemi cache a dále

podle výskytu cache v seznamu může uživatel vidět: *tradiční* (Obr. 10), *multicache*, *unknown*, *earthcache*. Každý rámeček má za názvem uveden počet cache tohoto typu ve vybraném seznamu a opět nahoře stránky je možno vidět aktuální stav GPS spolu s její přesností.



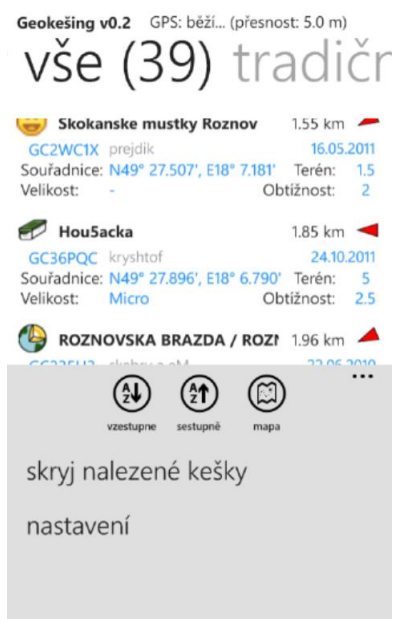
Obr. 9 – vybraný seznam cache, zobrazení všech cache

Po načtení stránky se vždy vybere rámeček se všemi cache (Obr. 9). Zde lze vidět prvek *listbox* s jednotlivými cache, kterému je styl přidělen pomocí prvku *grid* (viz. kapitola 3.2.2). Každá cache má uvedenu ikonu před názvem jejího typu (viz. typy cache v kapitole 1.1.4) nebo „smajlíka“, pokud se jedná o cache už uživatelem nalezenou. Dále je vidět její název, její vzdálenost od aktuální pozice, šipka ukazující k ní, její GC kód, přezdívka zakladatele cache s datem založení a finální gps souřadnice. Také je zde vidět obtížnost terénu, kde je cache uložena; obtížnost jejího nalezení, oboje na stupnici od jedné do pěti a velikost cache (viz. popis cache v kapitole 1.1). Po kliku na nějakou cache v seznamu, přejde program na stránku s jejím náhledem (*AktKeska.xaml*).



Obr. 10 - vybraný seznam cache, zobrazení tradičních cache

Všechny pivot prvky mají dole menu, které má tři tlačítka a dvě položky (Obr. 11). Tlačítka slouží k sestupnému či vzestupnému seřazení cache (ikona AZ se šipkou dle směru řazení) podle určitých parametrů (Obr. 12), a to podle *vzdálenosti*, *názvu*, *velikosti*, *GC kódu*, *obtížnosti*, *terénu* – výběr se provede pomocí prvku *listpicker*, který je vidět přes celou obrazovku - či zobrazení mapy (*Mapa.xaml*). Na mapě jsou vždy zobrazeny cache z aktuálního pivot prvku, tedy jen cache určitého typu (nebo všechny v seznamu, pokud je vybrán pivot prvek *vše*).



Obr. 11 – vybraný seznam cache, otevřené menu

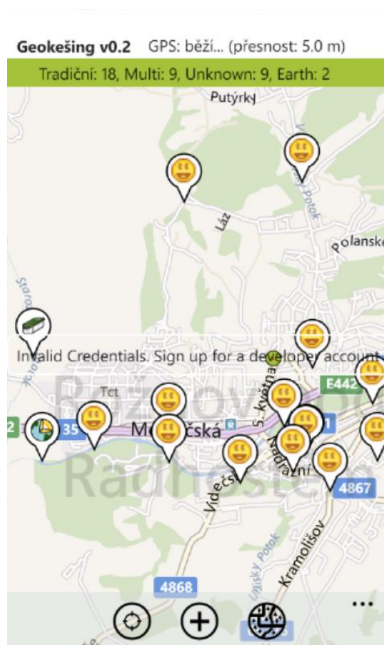
V menu se nacházejí položky pro přechod na nastavení aplikace (*Nastaveni.xaml*) nebo na skrytí již nalezených cache ze seznamu – s tím, že po tomto skrytí se položka menu změní na možnost nalezené cache opět zobrazit.



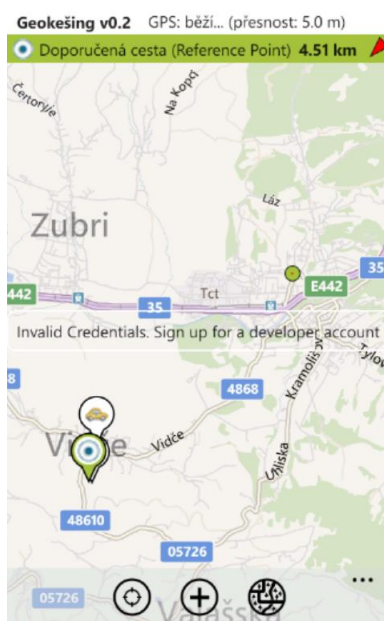
Obr. 12 - vybraný seznam cache, seřadit cache podle

5.4 Mapa

Na stránku s mapou se dá přejít buďto ze seznamu cache (Obr. 9 a Obr. 10) nebo přímo z nějaké konkrétní cache (Obr. 15 a Obr. 18). Tyto dvě možnosti se liší pouze tak, že při přechodu z náhledu konkrétní cache se zobrazí i waypointy patřící k té dané cache (Obr. 14) a při přechodu ze seznamu cache se zobrazí pouze výchozí souřadnice cache (Obr. 13). Nahoře na stránce je klasicky zobrazen stav GPS spolu s její přesností.

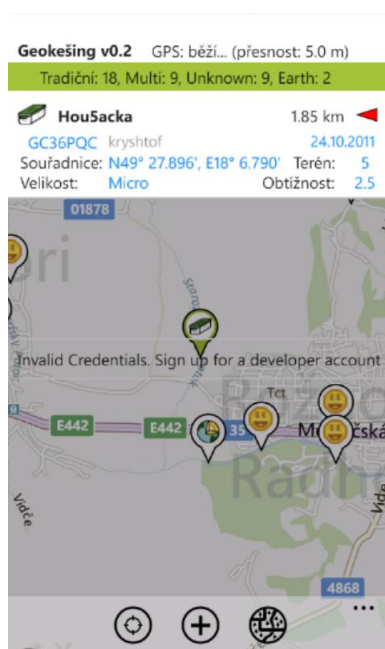


Obr. 13 – mapa, všechny cache seznamu



Obr. 14 – mapa, waypointy vybrané cache

Nahoře na stránce lze vidět informační pruh, který zobrazuje buď počty jednotlivých typů cache v zobrazeném seznamu nebo informace o aktivním waypointu vybrané cache – *typ, název, terén, obtížnost, vzdálenost a směr*. Na mapě jsou pak zobrazeny jednotlivé cache/waypointy, znázorněné ikonkou jejich typu (nebo „smajlíkem“, pokud jsou nalezené).



Obr. 15 – mapa, detail vybrané cache

Po kliknutí na některou cache na mapě, se zobrazí náhled jejich parametrů se stejným stylem jako na stránce se seznamem (Obr. 15). Pokud na něj uživatel klikne, dostane se na další stránku s náhledem té dané cache (*AktKeska.xaml*). Náhled se zavře kliknutím kdekoliv na zašedlou plochu či stiskem hardwarového tlačítka zpět.

Pokud uživatel klikne na waypoint, nastaví se jako aktivní a program bude k němu navigovat – jak je vidět na Obr. 14, kde je uživatel navigován k waypointu *Doporučená cesta*, vzdáleném 4,51km na jihozápad. Také se zobrazí možnost daný waypoint smazat či upravit – což přesměruje uživatele na stránku *PridejWaypoint.xaml*.

Dále lze na mapě vidět zelený kruh, který znázorňuje aktuální pozici uživatele na mapě. Kolem něj je zobrazen červený kruh, znázorňující přesnost GPS.

Ve spodní části na stránce je vidět menu s tlačítky umožňujícími vycentrovat mapu na všechny zobrazené cache/waypointy (ikona zaměřovače), přidat další waypoint (ikona

plusu) nebo změnit mód mapového podkladu na letecký pohled nebo klasické zobrazení ulic. Menu obsahuje položku nastavení aplikace (*Nastaveni.xaml*).

5.5 Přidání/upravení waypointu

Stránka umožňuje buď přidat nový waypoint (Obr. 16) nebo upravit stávající (Obr. 17). V obou případech je nutné, aby byly vyplněny souřadnice a název; popis waypointu je volitelný. Souřadnice je možné zadat pouze ve stupních° nebo ve stupních° a minutách – to uživatel zvolí pomocí *radiobuttonů*.



nový waypoint

Stupně Stupně a minuty

šířka: 49.4648

délka: 18.1388

název: novyy

popis: pobliz cile

 ...

Obr. 16 – nový waypoint

Menu stránky obsahuje pouze tlačítko na uložení waypointu. Při tomto uložení se provede kontrola, zdali je waypoint zadán správně – tedy pokud uživatel nezadal do souřadnic například znaky místo čísel či jsou stupně/minuty ve správných rozsazích.

upravit waypoint

Stupně Stupně a minuty

šířka: 49 ° 27.79500 ' ✓ N

délka: 18 ° 6.74100 ' ✓ E

název: Misto k parkovani

popis: Zde se da zaparkovat, pokud neni misto plne - je to vedle zahradkarske kolonie, takže byva často obsazeno.

Ⓜ ...

Obr. 17 – upravit waypoint

5.6 Nastavení programu

Stránka s nastavením (Obr. 18) umožňuje změnit přesnost funkce telefonu *zjišťování polohy*. Ve výchozím stavu je zvolena výchozí přesnost, což znamená menší přesnost zaměření pozice a větší výdrž baterie – výchozí přesnost zjišťuje polohu uživatele primárně pomocí BTS stanic, nedalekých Wi-Fi sítí a až sekundárně pomocí GPS přijímače. Vysoká přesnost pak primárně využívá GPS přijímače, takže je zjištění polohy přesnější, ale baterie telefonu je dříve vybita. Obě volby se projeví až po restartování programu.

nastavení GPS

Přesnost GPS
výchozí přesnost

Varování: vysoká přesnost sníží výdrž baterie!

Požadovaná přesnost gps v metrech:

5 metrů
15 metrů
30 metrů
50 metrů

pozn.: vyšší hodnota se rovná vyšší výdrži baterie,
jelikož nebude docházet k tak časté aktualizaci
polohy a odruší se nežádoucí šum

Obr. 18 – nastavení programu

Větší výdrži baterie může uživatel dosáhnout nastavením *požadované přesnosti gps v metrech*, což odruší různé skoky/šumy zjišťování polohy a nebude docházet k tak častému vykonání eventu změny aktuální polohy GPS a tedy se bude šetřit baterie telefonu. Na druhou stranu čím uživatel nastaví více metrů, tím nepřesněji bude ukazovat program aktuální pozici, neboť nedojde k aktualizaci polohy při sebemenším pohybu uživatele.

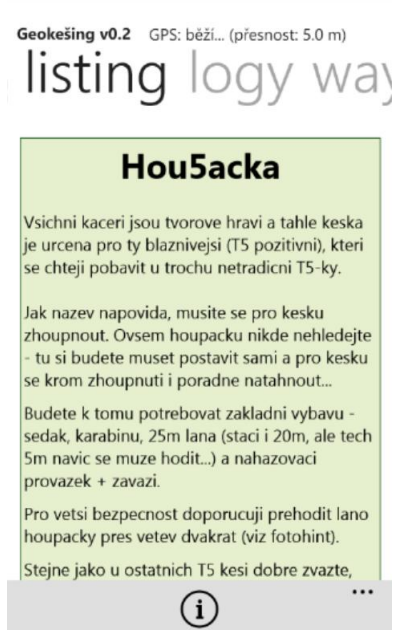
5.7 Náhled vybrané cache

Stránka s náhledem vybrané cache (Obr. 19 až Obr. 22) je vlastně srdcem programu, jelikož zde bude trávit uživatel nejdéle času. Je sestavena z prvku pivot, jehož jednotlivé rámce zobrazují uživateli vše potřebné o cache – všechny parametry; strelku kompasu, která ukazuje jak k severu, tak k aktivnímu waypointu; úplný listing; logy ostatních uživatelů a waypointy. Opět můžeme vidět nahoře na stránce zobrazení aktuálního stavu GPS spolu s její přesností v metrech a pod ním následuje ovládání pivot prvku.



Obr. 19 – náhled vybrané cache, její parametry a kompas

V rámci s názvem *kompas*, se nachází podobně stylovaný náhled parametrů cache jako na stránce se seznamem cache, avšak zde může uživatel ještě vidět údaje o aktuální nadmořské výšce, aktuální rychlosti a aktuální pozici GPS. Také je zde samozřejmě vidět štrelka kompasu – červená ukazuje k severu a modrá k zvolenému waypointu – a aktuální směr i azimut ve stupních. Aktivní waypoint je znázorněn dole a je vždy zobrazen jeho název, souřadnice ve tvaru stupně° minuty' a jak je daleko od aktuální pozice. Rámec *kompas* má také samozřejmě menu s třemi tlačítky, jenž umožňují vypnout/zapnout GPS kvůli úspoře baterie telefonu (ikona pauzy/přehrávání); nastavit výchozí souřadnice, pokud je zvolena navigace k nějakému waypointu (ikona domečku) a zobrazit cache s jejími waypointy na mapě (*Mapa.xaml*). Menu má dvě položky: obligátní nastavení telefonu a možnost označit cache za nalezenou.



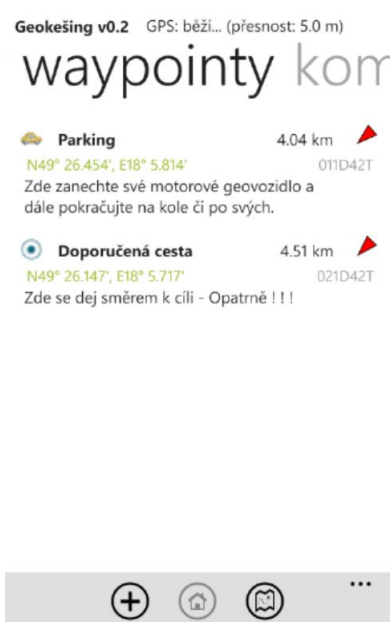
Obr. 20 - náhled vybrané cache, úplný listing

Dalším rámcem je *listing* (Obr. 20), kde je zobrazen úplný listing dané cache i s obrázký (ty nejsou uloženy v telefonu). Jedná se o prvek *webbrowser*, takže je možné listing různě scrollovat, přibližovat a manipulovat s ním stejně jako ve webovém prohlížeči. Menu ukrývá jedno tlačítko (ikona informace) pro zobrazení nápovědy neboli *hintu*, dle slangu geocachingu.



Obr. 21 - náhled vybrané cache, uživatelské logy

Rámec *logy* (Obr. 21) obsahuje několik uživatelských logů k dané cache. Zobrazen je vždy typ logu pomocí malé ikonky, přezdívka autora logu, datum uložení logu a samotný text logu. Uživateli je umožněno přidat vlastní log pomocí tlačítka menu, jenž ho přesměruje na formulář dané cache na oficiálních stránkách Geocachingu.



Obr. 22 - náhled vybrané cache, waypointy cache

Poslední rámec obsahuje *waypointy* cache (Obr. 22). U každého waypointu je zobrazen jeho typ pomocí ikonky, jeho název, jeho vzdálenost, směr k němu, souřadnice, jeho kód a popis. Waypointy mají kontextové menu (zobrazí se po podržení prstu na waypointu po dobu cca 2s, což je stejné pro všechna kontextová menu na WP7 programech), jenž umožňuje waypoint smazat, upravit (přejde na stránku *PridejWaypoint.xaml*) nebo nastavit jako aktivní. Menu dále umožňuje přidat nový waypoint (ikona plusu), nastavit výchozí souřadnice jako aktivní (ikona domečku) a zobrazit cache na mapě.

ZÁVĚR

V teoretické části bakalářské práce byla popsána samotná turistická hra Geocaching, její historie, pravidla a samotný způsob hraní a stručně také funkce navigačního systému GPS i způsob vývoje aplikací pro mobilní zařízení se systémem Windows Phone 7.

Praktická část popisuje samotnou aplikaci, která byla navrhnutá tak, aby byla co možná nejjednodušší na ovládání; měla svižný běh; aby všechny její prvky byly dostatečně viditelné a informace z ní byly přehledně čitelné; aby mohl uživatel jednoduše importovat do aplikace velké množství cache v uživatelských seznamech (ve formátu gpx), přehledně je spravovat, mohl si zobrazit cache na mapě (i jen pouze některé typy), mohl cache v seznamech seřazovat podle určitých parametrů a konečně aby si uživatel mohl zobrazit detail pouze jedné cache se všemi potřebnými informacemi (listing, logy, waypointy, souřadnice, parametry cache) a pomocí kompasu či mapy byl navigován k úspěšnému nalezení cache. Popis aplikace postupně popisuje jednotlivé aplikační stránky a celkové možnosti aplikace. Dále popisuje její vývoj ve Windows Phone SDK 7.1 s popisem jejich nejdůležitějších metod. Samotný návrh aplikace proběhl po zevrubném prozkoumání existujících aplikací pro mobilní zařízení s OS WP7, jejichž klady a zápory výrazně ovlivnily praktický návrh a vývoj aplikace.

ZÁVĚR V ANGLIČTINĚ

In theoretical part of Bachelor thesis was described a turistic game Geocaching, its history, rules and how to play it and briefly too function of navigation system GPS with practise of developing applications for mobile devices with operation system Windows Phone 7.

Practical part describes this particular application which was designed to be the simplest to control; to had swift run; to have its components visible enough with clearly readable content; to allow user easy import large amount of caches from user lists (in gpx format), manage them, allow user to display them on the map (even only specific types), to arrange cache in list specifically by cache details and finally allow user to view only specific cache with all necessary information (listing, logs, waypoints, coordinates, cache details) and navigate user to successfully finding the cache with use of compass or map. Description of application describes individual pages and its overall capabilities. Afterwards describes its development in Windows Phone SDK 7.1 with description its most significant methods. Application design itself preceded thorough research through existing applications for mobile devices with OS WP7 whose pros and cons significantly affected practical application design and development.

SEZNAM POUŽITÉ LITERATURY

- [1] SHARP, John. Microsoft Visual C# 2010: krok za krokem. Vyd. 1. Brno: Computer Press, 2010, 696 s. ISBN 978-802-5131-473.
- [2] PETZOLD, Charles. Programming Windows Phone 7. Redmond, WA: Microsoft Press, 2010. ISBN 978-073-5643-352.
- [3] SHARP, John. Microsoft silverlight edition: Programming Windows phone 7. Microsoft silverlight ed. Redmond, Washington: Microsoft Press, 2010, 792 s. ISBN 07-356-5667-3.
- [4] Windows Phone Development. Msdn.com [online]. December 15, 2011 [cit. 2012-02-01]. Dostupné z: [http://msdn.microsoft.com/en-us/library/ff402535\(v=vs.92\).aspx](http://msdn.microsoft.com/en-us/library/ff402535(v=vs.92).aspx)
- [5] Seriál C#: C# - Úvod. Programujte.com [online]. ? 2005-2009 [cit. 2012-02-01]. Dostupné z: <http://programujte.com/clanek/2005071001-c-uvod/>
- [6] Language Integrated Query. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001-2012 [cit. 2012-05-04]. Dostupné z: http://en.wikipedia.org/wiki/Language_Integrated_Query
- [7] Geocaching. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001-2012 [cit. 2012-05-04]. Dostupné z: <http://en.wikipedia.org/wiki/Geocaching>
- [8] GEOCACHING.CZ. GeoWiki [online]. 2012 [cit. 2012-05-04]. Dostupné z: http://wiki.geocaching.cz/wiki/Hlavní_strana
- [9] GROUNDSPEAK, Inc. Geocaching [online]. 2000-2012 [cit. 2012-05-04]. Dostupné z: <http://www.geocaching.com/>
- [10] Global Positioning System. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001-2012 [cit. 2012-05-04]. Dostupné z: <http://en.wikipedia.org/wiki/Gps>
- [11] Princip a fungování GPS. NaWEBka [online]. 2005 [cit. 2012-05-04]. Dostupné z: <http://www.rydval.cz/phprs/view.php?cisloclanku=2005110301>

- [12] Calculate distance, bearing and more between Latitude/Longitude points. Movable Type Ltd. [online]. 2010 [cit. 2012-05-04]. Dostupné z: <http://www.movable-type.co.uk/scripts/latlong.html>
- [13] Windows Phone Development QuickStarts. Microsoft Corporation [online]. 2012 [cit. 2012-05-04]. Dostupné z: <http://create.msdn.com/en-US/education/quickstarts>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

BTS	Base Transceiver Station
GPS	Global Positioning System
gpx	GPS eXchange Format
IS	IsolatedStorage
ISS	IsolatedStorageSettings
LINQ	Language Integrated Query
NMEA	National Marine Electronics Association
OS	Operační systém
WP7	Windows Phone 7
XAML	Extensible Application Markup Language

SEZNAM OBRÁZKŮ

Obr. 1 – ortodoma.....	19
Obr. 2 – příklad prvku <i>pivot</i>	22
Obr. 3 – příklad prvku <i>panorama</i>	23
Obr. 4 – blokové schéma stránek aplikace.....	32
Obr. 5 – načítací obrazovka aplikace	33
Obr. 6 – hlavní stránka, náhled uživatelských seznamů	34
Obr. 7 – hlavní stránka, import uživatelských seznamů	34
Obr. 8 - hlavní stránka, zobrazení aktuální pozice.....	35
Obr. 9 – vybraný seznam cache, zobrazení všech cache.....	36
Obr. 10 - vybraný seznam cache, zobrazení tradičních cache	37
Obr. 11 – vybraný seznam cache, otevřené menu.....	38
Obr. 12 - vybraný seznam cache, seřadit cache podle	38
Obr. 13 – mapa, všechny cache seznamu.....	39
Obr. 14 – mapa, waypointy vybrané cache	39
Obr. 15 – mapa, detail vybrané cache	40
Obr. 16 – nový waypoint.....	41
Obr. 17 – upravit waypoint	42
Obr. 18 – nastavení programu.....	43
Obr. 19 – náhled vybrané cache, její parametry a kompas	44
Obr. 20 - náhled vybrané cache, úplný listing.....	45
Obr. 21 - náhled vybrané cache, uživatelské logy.....	45
Obr. 22 - náhled vybrané cache, waypointy cache.....	46

SEZNAM TABULEK

Tab. 1 – proměnné třídy <i>SeznamKeseK_Class</i>	25
Tab. 2 - proměnné třídy <i>Log_Class</i>	26
Tab. 3 – proměnné třídy <i>Waypoint_Class</i>	26
Tab. 4 – proměnné třídy <i>Keska_Class</i>	27

SEZNAM PŘÍLOH

1x CD – obsah:

- SOURCE – zdrojové kódy aplikace (celý solution MS VS 2010)
- GPX – příklady gpx souborů se seznamy cache
- DOC – bakalářská práce