

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra informatiky a výpočetní techniky

Diplomová práce

Multimédia pro informační majáky

Plzeň, 2013

Bc. Jaroslav Ráb

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 14. 8. 2013

Bc. Jaroslav Ráb

Poděkování

Na tomto místě bych rád poděkoval vedoucímu práce panu Ing. Ladislavovi Pešíčkovi za rady, připomínky a pomoc při řešení práce.

Abstrakt

Multimedia for Information Beacons

This diploma thesis deals with multimedia on modern mobile devices and their use in system of information beacons. Information beacon is either visible or invisible element, which can be detected by smartphone or tablet, and information content, can be read, heard or seen from it.

In this thesis is also exploring the possibilities offered by modern operating systems on mobile devices to work with multimedia. The result is demonstrating the functionality on the mobile client application for the Windows Phone platform.

Obsah

1. Úvod.....	9
1.1 Informační majáky	9
1.2 Cíl práce.....	9
2. Existující systémy informačních majáků.....	10
2.1 Druhy využití systému informačních majáků	10
2.1.1 Využití v zoologických zahradách.....	10
2.1.2 Využití v muzeích, galeriích a na výstavách	10
2.1.3 Využití v logistických skladech.....	11
2.1.4 Využití na letištích	11
2.1.5 Využití v elektronických průvodcích měst nebo národních parků	11
2.1.6 Využití v obchodních centrech	11
2.1.7 Využití v hromadné dopravě.....	11
2.1.8 Využití ve školách a na univerzitách	12
2.2 TripAdvisor.....	12
2.3 Trips in Paris	13
2.4 Zoo Guide	13
2.5 iZeeum, iOS	14
2.6 Central Park (Listen to the Light), iOS	15
3. Multimédia na mobilních zařízeních	16
3.1 Obrázky.....	16
3.1.1 Repräsentace dat obrazu	16
3.1.2 Komprese obrazu	16
3.1.3 Formáty	17
3.1.4 Podpora na mobilních zařízeních.....	20
3.2 Audio	20
3.2.1 Kodeky a formáty	21
3.2.2 Podpora na mobilních zařízeních.....	23
3.3 Video.....	24
3.3.1 Formáty (Standardy)	24
3.3.2 Kodeky.....	25
3.3.3 Podpora na mobilních zařízeních.....	26
3.4 Distribuce multimédií	27
3.5 Streaming	28
3.5.1 Síťové protokoly pro podporu streamování.....	29

3.5.2	Příklady některých streamovacích řešení	31
3.5.3	Zhodnocení	33
4.	Windows Phone.....	34
4.1	Historie.....	34
4.1.1	Windows Mobile.....	35
4.1.2	Windows Phone 7	35
4.1.3	Windows Phone 8	36
4.2	Vývoj aplikací.....	38
4.2.1	Přehled	38
4.2.2	Základní prvky	39
4.2.3	Model-View-ViewModel.....	41
4.2.4	Síťová komunikace	41
4.2.5	Isolated storage a vnitřní databáze.....	42
4.2.6	Lokalizace	43
4.2.7	Využití funkčnosti interních aplikací.....	44
4.3	Publikace aplikace	44
5.	Mobilní klient pro Windows Phone	46
5.1	Systém Tagiee.....	46
5.2	Návrh aplikace	47
5.2.1	Komunikace s Tagiee API	47
5.2.2	Zobrazení dostupných institucí	47
5.2.3	Zobrazení detailu instituce.....	48
5.2.4	Hledání informačních majáků.....	48
5.2.5	Zobrazení obsahu majáku	49
5.2.6	Odesílání statistik.....	49
5.3	Implementace	49
5.3.1	Komunikace s API Tagiee	49
5.3.2	Databáze a Isolated storage.....	50
5.3.3	Jedinečná identifikace	52
5.3.4	Návrh uživatelského prostředí	52
5.3.5	Hledání majáků	55
5.3.6	Stažení obsahu	56
5.3.7	Streamování videa.....	56
5.3.8	Nahrání fotografií	58
5.3.9	Oprávnění aplikace	59

5.3.10	Upgrade aplikace pro WP8	60
5.3.11	Schválení a publikování aplikace	60
6.	Ověření funkčnosti	61
6.1	Instituce Západočeská univerzita v Plzni	61
6.2	Klientská aplikace Tagiee pro platformu Windows Phone	62
7.	Závěr	64
	Seznam zdrojů a literatury	65
	Seznam zkratk	68
	Seznam obrázků	70
	Seznam tabulek	71
	Přílohy	72
A.1	Uživatelská příručka mobilního klienta	72
A.2	Návod na nastavení VLC	74
A.3	Seznam JSON požadavků a odpovědí	78

1. Úvod

1.1 Informační majáky

Informační majáky není doposud ustálený pojem v běžném životě ani v oboru informačních technologií, a proto na úvod práce uvedu přiblížení tohoto pojmu. Jedná se o bod, ať už viditelný pro běžného člověka nebo jen dostupný přes určité technologie, který je svázán s fyzickým místem. Toto místo rozšiřuje o další funkčnost a obsah, se kterým může uživatel dále pracovat.

V dnešní moderní době jsme zavaleni spoustou informací, ať už v papírové nebo elektronické podobě. Obvykle jsou tyto informace netříděné, špatně zacílené a spíše obtěžují, než aby nám ulehčily každodenní život. Informační majáky jsou nástroj jak dostat informace pro ty, kteří je potřebují a v okamžik, kdy je potřebují. Některé takovéto majáky jsou již v našem okolí dostupné a přes technologie v našich tzv. chytrých mobilních telefonech, tabletech nebo počítačích se k nim můžeme připojit.

Například informační majáky na autobusových zastávkách, pomocí nichž zjistíme nejbližší odjezdy námi požadovaných spojů. Nebo inteligentního průvodce měst, památek, přírodních rezervací atd., kdy dostaneme zajímavou informaci o památce přesně ve chvíli, kdy se u této památky nacházíme.

1.2 Cíl práce

Cílem diplomové práce bylo prozkoumání trhu a nalezení podobných systémů, které využívají informačních majáků s ohledem na poskytování multimediálního obsahu. Prozkoumání možností, které poskytují moderní operační systémy na mobilních zařízeních pro práci s multimédií. Současně demonstrovat funkčnost na klientské mobilní aplikaci pro platformu Windows Phone společně se streamovacím systémem.

2. Existující systémy informačních majáků

Situací, kdy firma nebo instituce potřebuje doručit informace k návštěvníkovi, dokážeme vymyslet nespočet. Existuje mnoho možností, jak informační majáky využít. V této kapitole jsou uvedeny některé příklady použití, obory. Dále jsou pak popsány některé konkrétní systémy, které existují a mají klíčovou uživatelskou aplikaci pro platformu Windows Phone.

2.1 Druhy využití systému informačních majáků

Systémy můžeme nejjednodušeji dělit podle toho, v jaké organizaci se používají. Obvykle budou s uživatelem integrovat aplikace v mobilním zařízení, který má uživatel vlastní nebo mu může být v rámci třeba prohlídky zapůjčeno. Mobilními zařízeními myslíme tzv. chytré mobilní telefony nebo tablety. Zařízení mohou informační majáky nacházet pomocí integrovaných technologií: WiFi, Bluetooth, NFC, Fotoaparát, Infračervený port, GPS. Informace mohou získávat z vnitřní paměti nebo komunikací se serverem prostřednictvím internetu. Systém se může také použít pro zpětnou vazbu od zákazníka nebo návštěvníka, který jí odešle pomocí svého zařízení.

2.1.1 Využití v zoologických zahradách

Zoologické zahrady mohou být velice rozsáhlé a i výběhy jednotlivých zvířat mohou být veliké. Využití majáků v zoologických zahradách se tedy samo nabízí. Poskytne návštěvníkům komfort, pomocí něhož mohou dost informace o daném zvířeti přímo do rukou. Informace mohou obsahovat i dodatkové fotografie nebo videa či zvukový doprovod k jednotlivým zvířatům. Díky tomu, že zoo jsou obvykle venkovní prostory, se může uživateli poskytovat informace na základě jeho GPS polohy.

2.1.2 Využití v muzeích, galeriích a na výstavách

Podobně jako u zoologických zahrad by se využilo majáku k poskytování informací o exponátech. Aplikace by mohla také zvýšit interaktivitu návštěvníka například soutěží nebo kvízem, který by návštěvníci vyplňovali v průběhu prohlídky na svém mobilním zařízení a na konci odeslala na vyhodnocení nebo slosování. Návštěvníci by mohli také pořizovat snímky či krátká videa a ta následně nahrávat na server instituce, za což by mohli být nějakým způsobem odměněni, například bonusovým obsahem a podobně.

Jelikož muzea a galerie jsou povětšinou v budovách, využilo by se u exponátů štítků s QR kódy nebo NFC čipy.

2.1.3 Využití v logistických skladech

Zaměstnanci mají speciální čtečky čárkových kódů, pomocí nichž získávají informace o různých předmětech ve skladu. Pomocí WiFi triangulace mohou snadno uložit k předmětu i jeho polohu v rámci skladu. Zpětně jde tuto vnitřní navigaci použít k hledání předmětu ve skladu.

2.1.4 Využití na letištích

Letiště bývají obvykle rozsáhlé budovy, ve kterých se člověk snadno ztratí. Informační majáky by byly v tomto případě použity pro navigaci cestujících, popřípadě by mohli poskytovat tipy na to, jak si zkrátit čekací čas na odlet na daném letišti. Pomocí mobilního zařízení by se mohl obsah poskytovat v cestujícího rodném jazyce.

2.1.5 Využití v elektronických průvodcích měst nebo národních parků

Na základě polohy uživatele by systém poskytoval informace k dané lokalitě. Mohl by také poskytnout informace o nejbližších bodech zájmu, jako jsou restaurace, obchody, nádraží, bankomaty atd. V národních parcích by mohl podle měnící se polohy uživatele měnit hudební podkres, který by umocňoval zážitek z výletu.

2.1.6 Využití v obchodních centrech

Systém informačních majáků by poskytoval informace o obchodech, u kterých se zákazník nachází. Mohl by poskytovat přehled nabízeného zboží a také aktuální slevy. U konkrétních výrobků by pak mohl poskytovat podrobné informace jako například přehledně zobrazené nutriční hodnoty. U oblečení by mohl existovat systém virtuálního zkoušení zboží. A také na základě sběru dat od uživatele by mohla obchodní centra zjistit, o co mají zákazníci největší zájem a u čeho se nejdéle zdrželi.

2.1.7 Využití v hromadné dopravě

Na zastávkách hromadné dopravy jsou již v dnešní době majáky, které poskytnou informace o nejbližších odjezdech spojů. Možné rozšíření by mohla být informace o nejbližších prodejnách městských dopravních podniků, možnost zakoupení jízdenky nebo plánování cest hromadnou dopravou.

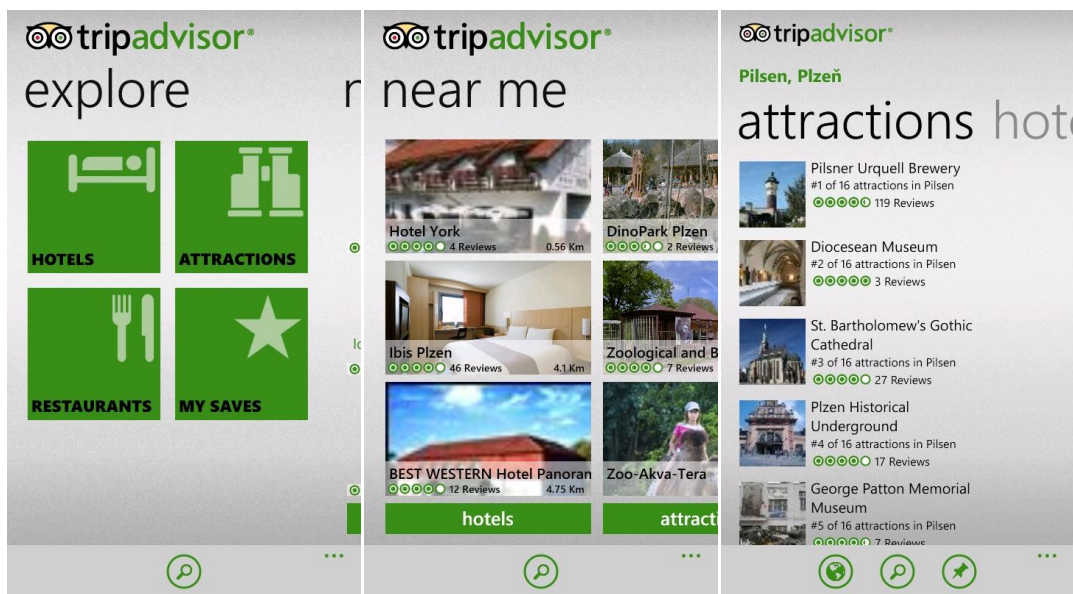
2.1.8 Využití ve školách a na univerzitách

Pro usnadnění orientace nových studentů by systém poskytoval základní informace o budovách. Velice přínosné by také byla navigace ve velkých budovách nebo databáze učeben a přednáškových místností s plánkem, kde se nachází. U knihoven by poskytoval otevírací dobu, u menz zase aktuální jídelní lístky.

2.2 TripAdvisor

TripAdvisor [1] je multiplatformní aplikace a webový portál, který slouží k vyhledávání bodů zájmů ve vašem okolí na základě vaší polohy, kterou může získat z GPS signálu nebo od BTS bran GSM signálu. Je možné také zadat místo ručně pro naplánování výletu předem. Z pohledu multimédií nabízí aplikace pouze zobrazení fotek a obrázků u jednotlivých bodů zájmu (viz Obrázek 2.1).

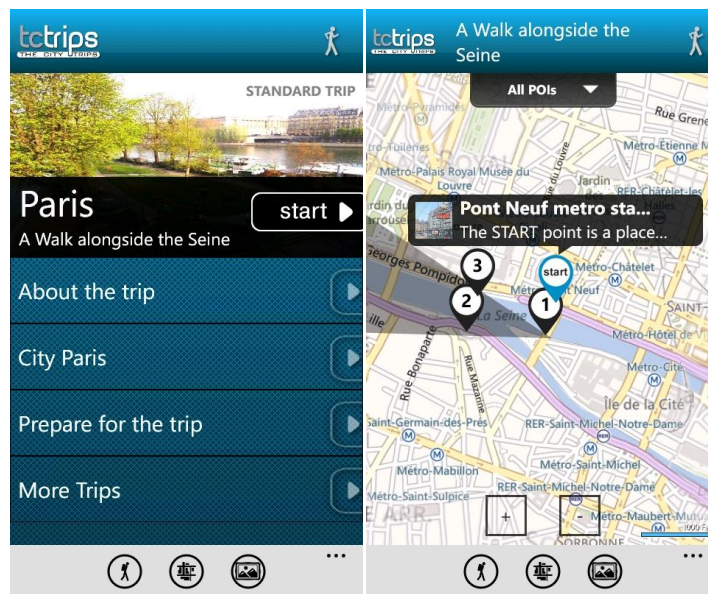
Na platformě Windows Phone existuje pouze jedna celosvětová aplikace, která veškerá data stahuje z internetu vždy až v době hledání na místě. Na platformách Android a iOS existují specializované aplikace pro velká světová města. V takovýchto aplikacích je možné obsah stáhnout a uložit do zařízení.



Obrázek 2.1 Aplikace TripAdvisor. Zobrazení bodů zájmu v okolí. Zajímavosti v okolí Plzně.

2.3 Trips in Paris

Trips in Paris [2] a další aplikace z kolekce TCtrip, jsou průvodcem světových měst. Aplikace naviguje uživatele městem a na určitých místech mu poskytne zajímavosti (viz Obrázek 2.2). Uživatel se může u památky také vyfotit a fotografii pak nahrát mezi ostatní.

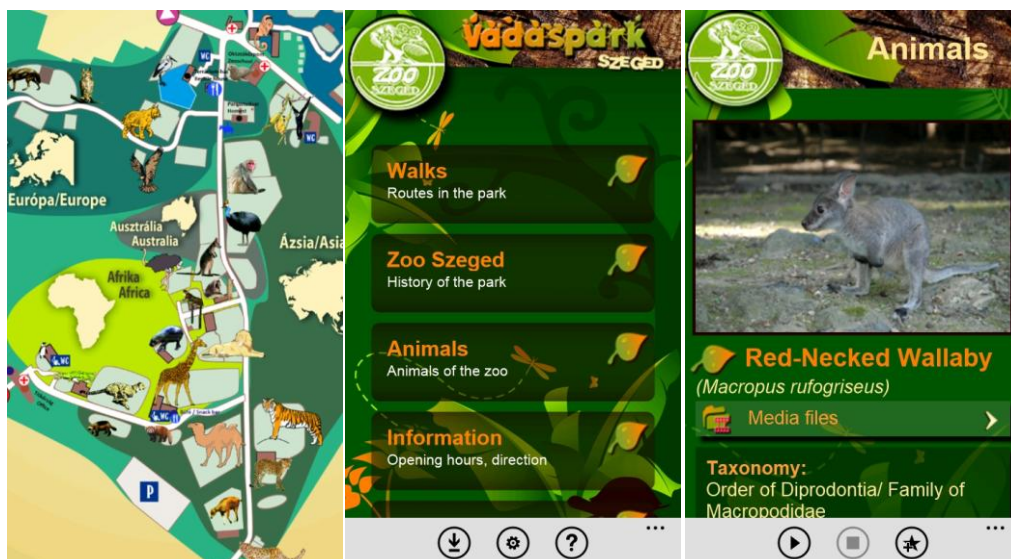


Obrázek 2.2 Aplikace Trips in Paris.

2.4 Zoo Guide

Zoologická zahrada v maďarském městě Szeged má na platformě Windows Phone svojí multimediální aplikaci. Aplikace zobrazuje obsah v několika jazycích. [3]

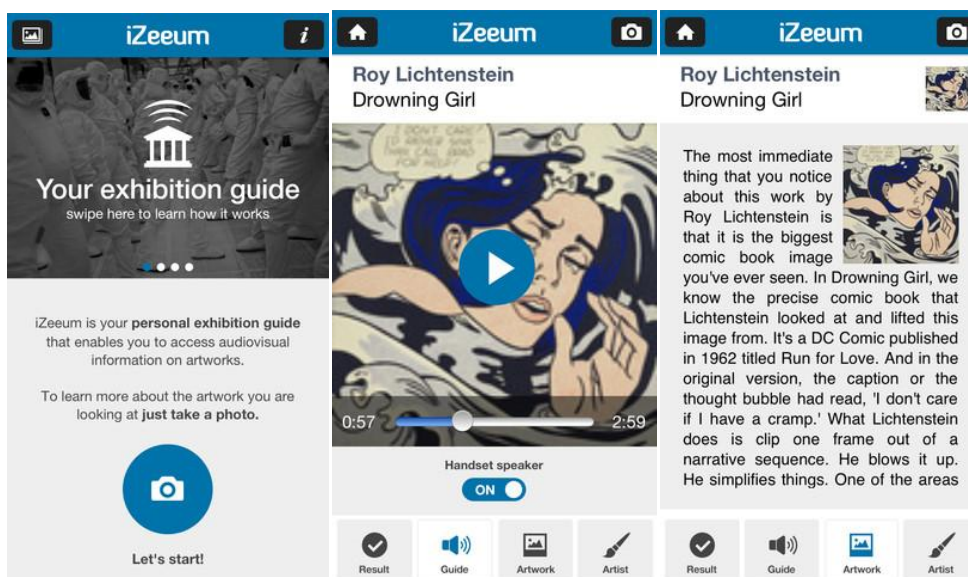
Mimo základní informace o otevírací době, vstupném a adrese, obsahuje i plánec zoo s vyobrazenými zvířaty, ke kterým je připraven textový popis, fotografie, ale také audio záznam. Audio záznamy je možné stáhnout předem do zařízení, což pro zahraniční turisty bude jistě výhodou. Navíc jsou dostupné v 5 jazycích (viz Obrázek 3.3).



Obrázek 2.3 Zoo Guide. Plánek Zoo. Menu. Detail zvířete s popisem a možností přehrát audio záznam.

2.5 iZeeum, iOS

Systém iZeeum [4] obsahuje databázi exponátů a uměleckých děl, které se vyskytují v muzeích. Nejzajímavější je použití, při kterém není potřeba načítat QR kód nebo udávat svou polohu, ale jednoduše dílo vyfotíte a vyfocenou fotku odešlete k analýze. Při analýze se zjistí, o jaké dílo se jedná a je poskytnut audio nebo video záznam s informacemi o exponátu (viz Obrázek 2.4).



Obrázek 2.4 iZeeum. Princip použití: Vyfocení, odeslání, analýza a za závěr poskytnutí multimediálního obsahu.

2.6 Central Park (Listen to the Light), iOS

Tato aplikace [5] v podstatě přehrává jen hudební skladby, které se stáhnou společně s aplikací. Skladeb je dostupných přes 400 od bratrů Hayse a Ryana Holladay. Vlastnost, kterou dělá tuto aplikaci zajímavou je to, že hudbu přehrává pouze v Central Parku v New Yorku. Navíc na základě měnící se polohy chodce v parku se mění také přehrávaná hudba. Je potřeba mít zapnuté přijímání signálu GPS.

3. Multimédia na mobilních zařízeních

Tato kapitola nejdříve teoreticky přibližuje vlastnosti a možnosti ukládání jednotlivých typů multimediálních dat (obrázků, zvuku a videí) a dále pak konkrétní podporu práce s multimédií na nejpoužívanějších mobilních platformách. Také jsou zde uvedeny možné způsoby distribuce videí.

3.1 Obrázky

Za nejjednodušší typ multimediálních dat můžeme považovat obrázky. Obrazový materiál je jen potřeba upravit na příslušný formát a vhodnou velikost (komprese dat) pro mobilní zařízení, podobně jako prezentace obrázků na webových stránkách. Je zbytečné na malém displeji mobilního zařízení zobrazovat několika megapixelové fotografie, pokud se jedná pouze o doplňující informační prvek prezentace. Popřípadě je možné poskytnout plnou kvalitu až dodatečně na vyžádání. Distribuce obrázků probíhá stáhnutím samotného souboru, a to buď do cache paměti pouze na omezenou dobu, nebo stáhnutím do vnitřní paměti zařízení.

3.1.1 Re prezentace dat obrazu

Re prezentovat data obrazu lze dvěma způsoby:

- Vektorová grafika - popis pomocí geometrických objektů (poloha, velikost, barva). Nezáleží na výsledném zobrazovaném rozlišení. Kvalita je vždy zachována. Nelze využít na fotografie a podobně složité obrazy. Vektorový formát je například SVG.
- Rastrová, bitmapová grafika - popis obrazu pomocí matice pixelů. Lze takto popsat nekonečně velké plátno, ale je zde vysoká paměťová náročnost. Právě u takovýchto obrazů se využívá komprese.

3.1.2 Komprese obrazu

Komprese se snaží o úspornější zápis matice pixelů. Lze toho docílit několika způsoby. Například neopakovat často se opakující úsek bitů nebo aproximovat data a naleznout vhodnou funkci pro jejich jednodušší zápis. Podrobný popis komprimačních technik přesahuje rozsah této práce. Nejdůležitějším parametrem je kompresní poměr. Jedná se o podíl velikosti původních dat a velikosti dat komprimovaných. Komprese provádí hlavně ke snížení datového toku při práci s daty.

Kompresce se dělí na [6]:

- bezztrátovou - dosahuje malé kompresní poměry. Malá úspora dat, ale lze rekonstruovat původní zprávu.
 - Run-length encoding (RLE)
 - Huffmanovo kódování
 - LZ77, LZW
 - Aritmetické kódování
- ztrátovou - lze určit kompresní poměr předem. Lze také předem definovat velikost výsledného obrázku. Ztráty vznikají u méně důležitých dat (přechody barev) z důsledku využití transformací dat.
 - Fourierovské transformace
 - Waveletová transformace
 - Fraktálová komprese

3.1.3 Formáty

V této kapitole jsou uvedeny nejpoužívanější formáty pro ukládání a kompresi obrazového materiálu.

BMP (Windows Bitmap)

- jednoduchý formát Windows, OS/2.
- nekomprimovaný formát
- paměťová náročnost
- různé barevné hloubky (1, 4, 8, 24 bitů na pixel)

Velice jednoduchý formát pro ukládání rastrové grafiky uveden v roce 1988 jako součást systému OS/2 [6][7]. Jedná se o nekomprimovaný formát. Tělo souboru obsahuje hlavičku, paletu použitých barev a sekvenčně po řádcích uložené hodnoty barev pixelů.

GIF (Graphics Interchange Format)

- omezení na 256 barev
- ideální pro obrázky s velkými jednobarevnými plochami
- používá LZW kompresi

- podpora animací
- využití na webu
- průhlednost

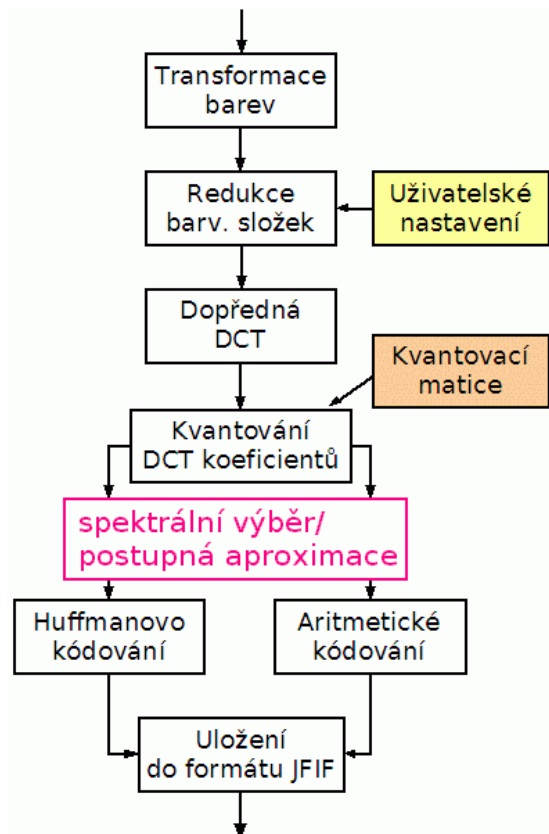
Největší omezení tohoto bezztrátového kompresního formátu je omezení na 256 barev [6][7]. Původní varianta byla představena v roce 1989 a vedla k několika sporům z důvodu použité komprimační metody LZW, jelikož došlo k porušení patentu společnosti Unisys. Výhodou je možnost ukládat více obrázků do jednoho souboru, tím lze vytvořit jednoduché animace. V současné době je nahrazován následovníkem, formátem PNG.

JPEG / JFIF (Joint Photographic Experts Group / JPEG File Interchange Format)

- standardizován (ISO/IEC 10918–1)
- ztrátová komprese s nastavitelnou úrovní (kvalitou)
- komprimační poměr 1:50 až 1:100
- podpora metadat (EXIF)

Zkratka JPEG (Joint Photographic Experts Group)[6][7][8] vlastně není název formátu ale skupiny, která vytvořila v roce 1992 ztrátový kompresní formát JFIF. Formát je vhodný pro ukládání fotografií a jiné složité obrázky, kde se prolíná mnoho barev, nejsou zde ostré přechody z jedné barvy do druhé, jelikož formát rozmazává ostré přechody.

Komprese [9] v případě formátu JFIF má několik fází. Nejprve je provedena transformace barev z barvových prostorů RGB, CMYK do barvového prostoru $YCbCr$. Tato transformace je bezztrátová. Následně může uživatel zvolit redukci barevných složek C_b a C_r . Další krok je aplikování diskrétní kosinové transformace (DCT) na bloky 8×8 hodnot. Výsledkem DCT jsou bloky 8×8 hodnot ležící ve frekvenční rovině. Bloky DCT jsou kvantovány pomocí vypočtených kvantizačních tabulek. Výsledkem je stav, kdy je mnoho hodnot v tomto bloku nulových, čehož se využívá v následujícím kroku zpracování. Kvantované DCT koeficienty jsou následně kódovány pomocí aritmetického či Huffmanova kódování. Posledním krokem zpracování je uložení vytvořených dat do souboru typu JFIF/JPEG Proces je znázorněn na Obrázku 3.1.



Obrázek 3.1 Přehled operací při kompresi pomocí JPEG. [9]

PNG (Portable Network Graphics)

- standardizován (ISO/IEC 15948)
- bezztrátová komprese za bázi LZ77
- vytvořen pro web

Formát [10] byl vytvořen jako náhrada formátu GIF, který měl problémy s licencovanou kompresní metodou. Dokáže navíc ukládat větší barevnou paletu a dosahuje lepších kompresních poměrů. Uveden byl v roce 1996.

TIFF (Tagged Image File Format)

- rozšiřitelný formát
- široká specifikace, lze uložit prakticky cokoliv => problém se čtením
- lze použít různé komprese

TIFF [6][7] je rozšířený formát dříve hlavně pro ukládání dokumentů pro tisk nebo dokumentů ze scannerů a faxů. Nyní se jedná o flexibilní kontejner, který je

schopen uložit více obrázků v jednom souboru s různými druhy kompresí. Vhodný pro archivaci obrázků. Jeho vysoká flexibilita a široká specifikace vede ke špatné kompatibilitě.

3.1.4 Podpora na mobilních zařízeních

Podpora jednotlivých formátů na nejrozšířenějších mobilních platformách je možné vidět v Tabulce 3.1.

Formát / Platforma	JPEG (.jpg, .jpeg)	GIF (.gif)	PNG (.png)	BMP (.bmp)	WebP (.webp)	TIFF (.tiff, .tif)	ICO (.ico)
Android	Ano	Ano	Ano	Ano	Android 4.0+	Ne	Ne
iOS	Ano	Ano	Ano	Ano	Ne	Ano	Ano
Windows Phone	Ano	Ano	Ano	Ano	Ne	Ne	Ne

Tabulka 3.1 Podpora jednotlivých formátů obrázků na mobilních platformách. Kompletní podpora formátů a kodeků pro Android je uvedena na [11], pro iOS [12] a pro Windows phone [13].

Z tabulky 3.1 je zřejmé, že všechny platformy podporují většinu běžných formátů obrázků, proto nebude problém se zobrazením na mobilních zařízeních. Pro fotografie se obvykle na mobilních zařízeních používá formát JPEG a pro ostatní grafiky PNG jako v případě webových stránek.

3.2 Audio

Další typ multimediálních dat jsou zvukové záznamy. Pro efektivní přenos je potřeba zvolit vhodnou kompresi zvukových souborů. Hlavní důvod je zmenšení datového toku při práci se souborem. Distribuce zvuku může podobně jako u obrázků

probíhat stáhnutím samotného souboru nebo jeho streamováním do zařízení vzdáleně ze serveru.

Komprese [6] může být opět dvojího typu. Bezeztrátová komprese nedosahuje tak dobrých komprimačních poměrů, ale lze se zpětnou operací vrátit zpět k původní podobě. Využívá redundance v datech. Při ztrátové kompresi v případě zvukových souborů dochází k oříznutí zvukového signálu o část frekvenční charakteristiky, kterou není lidské ucho schopno zaznamenat. Získáme tak zmenšený soubor, aniž by to posluchač byl schopen rozeznat.

3.2.1 Kodeky a formáty

Kodek (anglicky codec) je složenina z počátečních slabik slov „kodér a dekodér“, respektive komprese a dekomprese. Převzato z anglického codec analogického původu. Kodek je HW zařízení nebo počítačový program pracující na základě algoritmů, který dokáže transformovat datový tok nebo signál. Kodéry ukládají data do zakódované formy. Naproti tomu dekodéry jsou používány pro obnovování přesné či alespoň originálu co nejpodobnější formy dat, například pro zobrazování. Kodeky jsou implementací nějakého formátu.

Apple Lossless (ALAC)

Formát ALAC [14] vyvíjí společnost Apple pro jejich přehrávače iPod. Jedná se o bezeztrátový kodek, který používá lineární predikci podobně jako ostatní bezeztrátové audio kodeky. Jeho výhodou je nenáročný dekódování vhodné pro mobilní zařízení a přehrávače. Kodek je součástí Přehrávače QuickTime, také od společnosti Apple. K přenosu využívá kodek kontejner MPEG-4 (.mp4, .m4a).

FLAC

Jedná se o otevřený kodek [6][14] pro bezeztrátovou kompresi hudby. I tento kodek využívá lineární predikci. Využívá se pro archivaci hudby nebo jako pracovní formát pro pozdější zpracování nebo úpravu ztrátovou kompresí. Pro ukládání používá kontejner FLAC (.flac).

MPEG-I Audio

MPEG-I Audio [6][14] je ztrátová komprese zvuku, která je součástí standardu MPEG-1, což je dříve používaný standard audiovizuálního obsahu. Jako ztrátový kodek ořezává pro lidské ucho neslyšitelné úseky nahrávky.

MPEG -1 Audio se skládá ze 3 vrstev:

- Layer 1 (mp1) - základní kompresní schéma, nejjednodušší, datový tok ≥ 128 kbit/s
- Layer 2 (mp2) - střední složitost, datový tok ≥ 128 kbit/s, video CD, DVD
- Layer 3 (mp3) - zavedení transformace signálu pomocí MDCT, nejsložitější, datový tok 32 – 224 kbit/s, nejrozšířenější hudební formát, kontejner MP3(.mp3)

Každá vyšší vrstva je výpočetně náročnější a více efektivní v nižším datovém toku než vrstva předchozí. Vrstvy jsou zpětně kompatibilní.

Vorbis

Vorbis [14] je ztrátový kodek, který měl být náhradou za MP3, jelikož dosahuje mnohem lepších kvalit. Bohužel se příliš neprosadil. Jedná se o open source od Xiph. Org. Nejčastěji je uložen v kontejneru OGG (.ogg) nebo Matroska (.mkv).

MPEG-II část 7 a MPEG-4 část 3 ACC

Kodek [6], který je součástí standardů MPEG-II a MPEG-4, je přímý následovník formátu MP3. Jde opět o ztrátový kodek nazývaný též Advanced Audio Coding (AAC). Formát je široce podporován na řadě zařízení. Existuje mnoho variant tohoto kodeku, např.: HE-AACv1, HE-AACv2, AAC+, eAAC+. Přenášeny může být mnoho kontejnery: MPEG-4, 3GP, MKV, MOV, AVI (typy souborů: .aac, .m4a, .m4b, .m4p, .m4v, .m4r, .3gp, .mp4).

WMA (Windows Media Audio)

WMA[14] je komprimovaný zvukový formát vyvíjen společností Microsoft jako součást Windows Media. Dosahuje větších kvalit než nejrozšířenější formát MP3. K formátu neexistuje otevřená specifikace, proto je pro práci s kodekem nutné používat systém Windows. Pro přenos se využívá kontejner ASF (typ souboru .wma).

WAV (Waveform Audio File Format)

Jedná se o jeden z prvních zvukových formátů [6], který vytvořili společnosti IBM a Microsoft. Jde o variantu formátu RIFF, který obvykle formátu WAV slouží jako kontejner. WAV může obsahovat různé komprimované i nekomprimované formáty zvuku, nejčastěji je však použita nekomprimovaná data LPCM. Jelikož jsou data nekomprimovaná, jsou velice paměťově náročná. Typ souborů: .wav, .wave.

3.2.2 Podpora na mobilních zařízeních

Podpora jednotlivých formátů a kodeků pro práci se zvukem na mobilních platformách je možné vidět v Tabulce 3.2.

Formát, kodek / Platforma	AAC (.3gp, .mp4, .m4a, .aac)	FLAC (.flac)	MP3 (.mp3)	Vorbis (.ogg, .mkv)	WAV (.wav)	WMA (.wma)	ALAC (.mp4)
Android	Ano	Ano	Ano	Ano	Ano	Ne	Ne
iOS	Ano	Ne	Ano	Ne	Ne	Ne	Ano
Windows Phone	Ano	Ne	Ano	Ne	Ne	Ano	Ne

Tabulka 3.2 Podpora jednotlivých formátů a kodeků pro práci se zvukem na mobilních platformách. Kompletní podpora formátů a kodeků pro Android je uvedena na [11], pro iOS [15] a pro Windows phone [13].

U práce se zvukem už není situace tak přehledná jako v případě formátů obrázků. Přesto všechny platformy podporují nejrozšířenější formát MP3 a jeho přímého nástupce AAC (viz Tabulka 3.2). Zvolíme-li tyto dva formáty, bude podpora stoprocentní.

3.3 Video

Videa jsou nejnáročnější multimédia, co se týče datového toku. Proto i zde je potřeba zvolit správný formát a kodek videa. V případě, že by kompresí nebyl zmenšen datový tok, nemohlo by se video přehrávat ani z klasického pevného disku, natož pak posíláno bezdrátovými sítěmi Wi-Fi nebo prostřednictvím sítí GSM mobilního operátora.

Samotné kódování videa je velice obsáhlá kapitola, proto jsou zde uvedeny jen základní příklady některých formátů, kodeků a video kontejnerů.

3.3.1 Formáty (Standardy)

MPEG-I (Motion picture experts Group)

- ISO/IEC-11172

Jeden z nejstarších standardů [6][16] pro ztrátovou kompresi zvuku a videa byl vytvořen skupinou MPEG (Motion picture experts Group) v roce 1991. Při tvorbě bylo účelem vytvořit formát, kterým se bude moci ukládat videa na VideoCD, kde je omezený datový tok čtení CD 1,5 Mbit/s.

Samotný standard se skládá z 5 částí:

- systém - synchronizace obrazu a zvuku
- video - kódování a komprese videa
- audio - kódování a komprese zvuku
- testování
- referenční software

Součástí části dvě, nazývanou také MPEG-1 Audio, je kodek pro hudební kompresi MP3.

MPEG-II

- ISO/IEC 13818

Tento formát [6][16] je vylepšením předchozího MPEG-1. Existuje zde zpětná kompatibilita. Formát je využíván u DVD. Není zde již omezení datového toku jako v

předchozí variantě. Standard je rozšířen z původních pěti částí na celkových 11 (např.: Advanced Audio Codec - AAC, Digital Storage Media Command and Control -DSM CC).

MPEG-4

- ISO/IEC 14496

Pokračovatel formátů [6][16] z rodiny MPEG navazující na MPEG-II. Využívá se pro přenos audiovizuálního obsahu u digitální TV nebo streamovaného internetového videa. Může mít velice proměnný datový tok, a to již od 5 kbit/s až do 1 Gbit/s. Oproti předchůdci je opět rozšířen co do počtu obsažených částí. Mezi důležitě novinky patří část 2, ze které vychází kodek pro kompresi videa Advanced Simple Profile (ASP) nebo pokročilejší část 10, též pro kompresi videa Advanced Video Coding (AVC nebo H.264).

VC-1 (WMV)

Jedná se o uzavřený formát [17] od společnosti Microsoft. Jde o alternativy k H.264 (MPEG-4 AVC), která je s ní srovnatelná jak z hlediska rozlišení, tak i přenosové rychlosti. Formát je používán na discích Blu-ray, ve Windows Media Player, Silverlight konzolích Xbox 360 i Playstation 3.

3.3.2 Kodeky

Uvedené kodeky jsou pouze výběrem z mnoha video kodeků vyskytujících se na trhu. Jedná se o implementace výše zmíněných video formátů. Existují bezztrátové a ztrátové kodeky. Bezeztrátové kodeky nemají vysoký komprimační poměr, proto se jimi práce dále nebude zabývat.

H.263

Jedná se o rozšíření implementace kodeků [6][16] z formátů H.261, MPEG-I a MPEG II. Využití našel při videokonferencích nebo při streamování internetových videí (YouTube), kde byl dříve hodně rozšířen. Nyní je postupně nahrazován novější H.264 variantou. Na rozdíl od předchůdce H.261 umožňuje měnit datový tok. Pro přenos využívá kontejnery 3GP (.3gp) a MPEG-4 (.mp4).

Kodeky implementující standard MPEG-4 ASP

Tyto kodeky [6][16] jsou vylepšením předchozích kodeků vycházejících ze standardů MPEG-II a H.263. Konkrétně se jedná o proprietární kodek DivX od stejnojmenné společnosti a kodek XviD, který je open source. Kodek DivX není pro volné použití, je vázán licenčními poplatky, naproti tomu Xvid je volně šiřitelný a dosahuje stejné nebo lepší kvality. Pro přenos se využívají kontejnery 3GP (.3gp) nebo AVI (.avi)

H.264

Kodek H.264 [6][16] je implementací standardu MPEG 4 části 10 Advanced Video Coding (AVC). Využívá se pro videokonference a streamování internetových videí (YouTube). V současné době jde na internetu o nejrozšířenější kodek, který je velice podporován na mnoha zařízeních. Samotný kodek podporuje jak nízké, tak velice vysoké datové toky, a proto je použitelný na mnoha typech sítí a různých rychlostech internetového připojení. Jedná se o jeden ze 3 kodeků použitých pro kódování videa pro disky Blue-ray. Pro přenos se využívají kontejnery 3GP (.3gp) a MPEG-4 (.mp4).

Windows Media Video (WMV)

Jde o implementace kompresního video formátu VC-1 [17] od společnosti Microsoft (licenční práva, omezení na platformu Windows). Využití našel ve streamování, jelikož podporuje variabilní datový tok. Tento kodek byl jeden ze 3 kodeků použitých pro kódování videa a zvuku na discích Blue-ray. Je integrován ve Windows Media Playeru. Pro přenos se využívá kontejneru ASF (.wmv), lze také využít AVI (.avi) nebo Matrošku (.mkv).

3.3.3 Podpora na mobilních zařízeních

Podpora jednotlivých formátů a kodeků pro práci s videem na mobilních platformách je možné vidět v Tabulce 3.3.

Formát, kodek /	H.263 (.3gp, .mp4, .m4v, .mov)	H.264 (.3gp, .mp4)	MPEG-4 ASP (.3gp, .m4v, .mp4)	VC-1 (.wmv)
Platforma				
Android	Ano	Ano	Ano	Ne
iOS	Ne	Ano	Ano	Ne
Windows Phone	Ano	Ano	Ano	Ano

Tabulka 3.3 Podpora jednotlivých formátů a kodeků pro práci s videem na mobilních platformách. Kompletní podpora formátů a kodeků pro Android je uvedena na [11], pro iOS [15] a pro Windows phone [13].

U práce s videem je situace asi nejsložitější ze všech typů multimédií. Nejlepší volbou, která vychází z Tabulky 3.3 bude formát MPEG-4, a to hned dva kodeky, odvozené od části 2 Advanced Simple Profile (ASP) a části 10 Advanced Video Coding (AVC nebo H.264). U ostatních formátů a kodeků už není stoprocentní podpora u všech mobilních platform.

V případě platformy Windows Phone nastává problém v případě streamování videa, kdy podporuje pouze formát VC1 s kodekem WMV v kontejneru ASF. Řešení může být překonvertování videa na tento formát až přímo v zařízení.

3.4 Distribuce multimédií

Pro možnost využití multimédií na mobilních zařízeních musíme vyřešit problém s jejich distribucí. U obrázků a menších hudebních souborů není problém celý soubor s obsahem stáhnout předem a uložit do mezipaměti nebo do vnitřní paměti zařízení.

V případě ukládání obsahu do vnitřní paměti může nastat problém a to pokud poskytovatel nechce některý obsah volně poskytovat pro ukládání, ale pouze na prohlížení/přehrávání. V mobilních platformách má každá aplikace svůj vlastní přístup

na vnitřní úložiště a jiná aplikace se k datům jiné aplikace nemůže dostat. Platformy Windows Phone a iOS navíc umožňují data také šifrovat. V případě tzv. rootnutí telefonu (Jailbreak) se, ale uživatel může dostat k veškerému obsahu v zařízení, což je u obsahu chráněným autorským právem problém.

Řešením těchto problémů může být streamování audio a video obsahu přímo do zařízení v době, kdy ho uživatel potřebuje. To samozřejmě vyžaduje neustálé připojení minimálně do lokální sítě. V případě sítě mobilního operátora nastává problém v limitu FUP, který mnohdy uživatele odradí od přehrávání videa, které i při nízkém datovém proudu je datově velice náročné. Video, které chceme streamovat musíme tedy vhodně zakódovat a komprimovat vhodným kodekem. Další možnost je mít streamovací server na lokální síti a tudíž pak není nutné připojení do sítě Internet.

3.5 Streaming

Streamování multimédií je proces kontinuálního přenosu audiovizuálního materiálu mezi zdrojem (streamovací server) a uživatelem (v našem případě mobilním zařízením).

Streamování jde rozdělit na několik částí [6]:

- Webcasting - jedná se o neustálé vysílání nehladě na počet přijímacích zařízení. Příkladem jsou internetová rádia a televize. Lze ho přirovnat k televiznímu nebo rádiovému vysílání (broadcasting).
- Video on demand (VOD) - video na vyžádání. Streamovací server naváže s každým klientem jedinečné spojení a vysílá různý nebo stejný obsah v různém čase každému klientovi zvlášť. Příkladem je YouTube.
- Video hovory a video konference (VoIP) - specifický druh streamování videa od uživatele k uživateli (peer-to-peer). Příkladem může být Skype.

Pro naše účely informačních majáků, kdy chce poskytovatel obsahu poskytnout každému majáku a každému uživateli různý multimediální obsah, bude zájmat právě druhá možnost videa na požádání.

3.5.1 Síťové protokoly pro podporu streamování

HTTP (Hypertext Transfer Protocol)

Jedná se o protokol [6][16][18] na aplikační vrstvě. Webový server v případě HTTP streamování neukončí komunikaci po odeslání dokumentu (části videa), ale ponechává odpovědi otevřené, aby mohl v případě dalšího požadavku okamžitě zaslat klientovi další data (další část videa). Jinak by tato data musela zůstat ve frontě, dokud nepřijde další žádost. V poslední době se přechází při streamování právě na tento protokol. HTTP protokol obvykle používá porty 80 nebo 8080. Pro přenos se využívá výhradně protokol TCP.

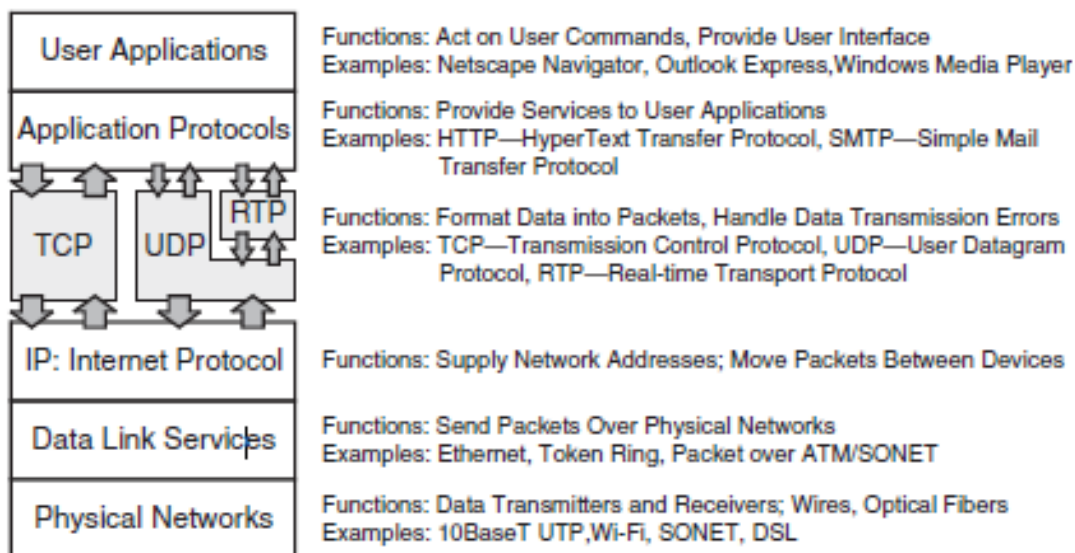
RTSP (Real Time Streaming Protocol)

Jedná se o protokol [6][16][18] na aplikační vrstvě. Byl publikován v roce 1998 jako RFC 2326. Je přímo určen pro streamování médií v systémech, které klientům umožňují dálkově ovládat na serveru multimediální datové proudy (play, pause, skok v čase). Ovládání probíhá pomocí adresní řádky podobně jako v případě HTTP. (například "rtsp://server.cz/mujfilm.rm"). Protokol má schopnost pohybovat se v souboru s videem na základě časových razítek obsažených ve videu a umožňuje uživateli přeskakovat jednoduše v čase videa při zachování audio synchronizace. RTSP může být přepravován přes UDP nebo TCP protokoly nebo také pomocí protokolu RTP (viz níže). Výchozí port pro RTSP je 554.

RTP, RTCP

RTP (Real-time Transport Protocol) [6][16][18] je protokol na transportní vrstvě, který dále spolupracuje s protokolem UDP (viz Obrázek 3.2). RTP přidává packetu speciální hlavičku a ten pak dále přenáší protokol UDP. RTP byl speciálně navržen pro přenos videa, zvuku, hlas nebo videohovorů, kde je důležitý čas doručení. Při těchto přenosech není tak důležité, jestli u některého z paketů dojde k chybě, ale spíše aby nebyl doručen pozdě a v nesprávném pořadí. Může tak dojít někdy k přeskočení obrazu nebo zvuku, ale je to lepší než mít dokonalý přenos, kdyby při každé chybě docházelo k zastavení a spouštění přenosu.

RTCP (RTP Control Protocol) [6][16][18] se používá společně s protokolem RTP. RTCP nese například informace o časových razítkách pro synchronizaci videa a zvuku, statistická data o přijatých a odeslaných (ztracených) paketech, informace o datovém toku, informace o účastnících RTP spojení.



Obrázek 3.2 Přehled protokolů na transportní vrstvě. [16]

MMS (Microsoft Media Server) a Smooth Streaming Transport Protocol

Microsoft Media Server [19] je proprietární streamovací protokol na aplikační vrstvě vyvinutý a používaný společností Microsoft. Používal se dříve ve Windows Media Services, který mimo MMS používal také RTSP. Výchozí port pro MMS je 1755. MMS mohou být přepravovány s pomocí UDP nebo TCP.

Technologii Windows Media Services [19] později nahradila novější IIS Media Services, který ustupuje od tradičních streamovacích protokolů směrem k univerzálnímu HTTP. Důvodem jsou problémy s RTSP (při přenosu UDP) u některých firewallů. Další důvod je nemožnost stahovat obsah výrazně dopředu. Problém při nedostatečně rychlé lince, kdy není možné video nechat načíst a následně plynule sledovat.

S příchodem IIS (Internet Information Services) Smooth Streaming [19] vytvořil Microsoft nový protokol IIS Smooth Streaming Transport Protocol, který popisuje, jak se distribuují a meziukládají data prostřednictvím protokolu HTTP.

3.5.2 Příklady některých streamovacích řešení

Flash

Platforma Adobe Flash byla široce rozšířená v internetových video aplikacích, včetně hlavních portálů, jako je YouTube. Adobe Flash platforma je plně funkční na všech počítačových systémech s pomocí aplikace Adobe Flash Player. Vytvořit vlastní streamovací server pomocí Flash by bylo jednoduché. Navíc podporuje moderní kodeky a formáty videa. Problém je, že Adobe Flash Player nepodporuje ani jedna z mobilních platform, proto je pro nás nepoužitelný.

HTML5

Další možností je vybudovat streamovací server pomocí jazyka HTML5. Jazyk HTML5 má ve své definici speciální tagy jak pro přehrávání videa <video>, tak audio <audio>. V budoucnu se chce stát nástupcem platformy Flash. HTML5 ve svých prohlížečích podporují všechny mobilní platformy. Navíc všechny umožňují vývojářům vložit do své aplikace tzv. WebView, což je v podstatě okno prohlížeče bez okolního rámečku. Takovému oknu můžeme definovat URL přímo s videem uloženým na serveru. Video se pak začne hned přehrávat nebo se spustí v nativním systémovém přehrávači. Aby video fungovalo, je ho potřeba nejdříve převést na vhodný formát (nejlépe MPEG-4 H.264). Pro konverzi a umístování videa na webové stránky nám může pomoci například program Easy HTML5 Video.

YouTube

Jednou z možností pro poskytovatele obsahu pro informační majáky může být hotové řešení, které nabízí server YouTube. Videa můžeme nahrát na server, kde se mu převedou na vhodný formát. V případě, že nechceme, aby videa byla dostupná široké veřejnosti, lze nastavit videa dostupná pouze přes soukromé URL, které bude znát pouze naše aplikace. V mobilním zařízení se pak spustí rovnou video, protože YouTube má podporu na všech mobilních platformách. Výhoda tohoto řešení je hlavně v tom, že není potřeba vyvíjet a provozovat vlastní streamovací server.








Microsoft IIS Smooth Streaming

IIS [19][20] je webový server vyvíjený společností Microsoft pro operační systém Windows. Součástí serveru je také technologie Smooth Streaming, které umožňuje jak webcast, tak VOD streamování videa prostřednictvím HTTP. Specialitou Smooth Streaming je schopnost reagovat v reálném čase na vytížení linky a podle toho upravovat datový tok videa. Klientská část přehrávače je tvořena technologií Silverlight též od společnosti Microsoft. Nově dokáže také vysílat v podobě HTML5. Hojně se využívá formátu MPEG-4 a kodeku H.264. Omezení tohoto řešení je fakt, že se jedná o komerční produkt, který je hlavně zaměřen na platformu Windows. S přímou podporou technologie Silverlight, například na mobilní platformě iOS od Applu, nemůžeme ani do budoucna počítat.

VLC

Program VLC [21] se součástí projektu VideoLan, které se zabývá přehráváním, konverzí a distribucí videí. Účelem VideoLAN je poskytnout software pro práci s videem, a to zdarma pod GNU licenci. VLC je vyvíjen na operačním systému Linux, ale existuje řešení na Windows i Mac OS. VLC se dá použít jako běžný přehrávač, který jde ale ovládat několika způsoby: přes webové rozhraní, příkazovou řádku, telnet, grafické prostředí a další. Je vybaven podporou mnoha kodek a formátů videa. Mimo jiné obsahuje také knihovny FFmpeg.

Podporuje samozřejmě i streamování, které nabízí nespočet možností nastavení a předvoleb (viz Obrázek 3.3). K dispozici je jak webcast, tak VOD streamování videa a to buď pomocí protokolu HTTP nebo RTP/UDP. Nesporně výhodnou funkcí je, že VLC dokáže při streamování video konvertovat kodekem do vhodného formátu a kontejneru. Nenastane tak problém, že bychom video streamovali v nevhodném formátu pro mobilní platformy.

								
Inputs	RTP/UDP	✓	✓	✓	✓	✓	✓	✓
	RTSP	✓	✓	✓	✓	✓	✓	✓
Output	RTP/DCCP	✗	✗	✓	✗	✗	✓	Untested
	Raw UDP	✓	✓	✓	✓	✓	✓	✓
	(RTP or raw) Multicast	✓	✓	✓	✗	✓	✓	✓
	File	✓	✓	✓	✓	✓	✓	✓
	HTTP	✓	✓	✓	✓	✓	✓	✓
	MMSH	✓	✓	✓	✓	✓	✓	✓
Misc	Transcoding	✓	✓	✓	✓	✓	✓	✓
	Send DVD subtitles	⊖	⊖	⊖	⊖	⊖	⊖	✗
	Send SAP announces	✓	✓	✓	Untested	✓	✓	Untested
Interfaces and more		See the VLC features page						

Obrázek 3.3 Přehled možností streamování serveru VLC. [21]

Mimo zmíněné řešení existují ještě další jako například:

- Darwin Streaming Server, který se součástí QuickTime od společnosti Apple
- Helix server od společnosti RealNetworks

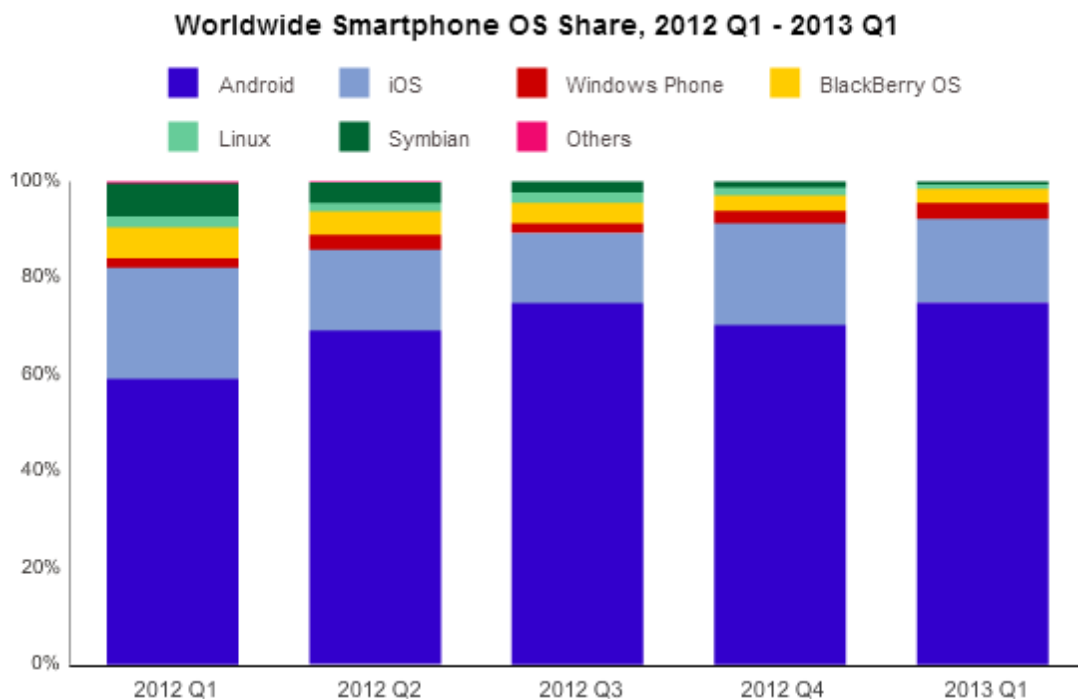
3.5.3 Zhodnocení

Z výše zmíněných streamovacích serverů jsem pro implementaci zvolil dva. Prvním zvoleným je server YouTube jako hotové řešení, které je snadné na nasazení, údržbu a také disponuje širokou podporou na všech mobilních platformách.

Druhým zvoleným řešením je program VLC, který je k dispozici zdarma, a proto v případě jeho nasazení v systému informačních majáků nebude problém. Je dostupný pro operační systém Windows i Linux. Mimo to podporuje opravdu široké spektrum možností streamování videí pomocí různých protokolů a v různých formátech.

4. Windows Phone

V této kapitole je přibližena mobilní platforma Windows Phone od společnosti Microsoft, její historii a specifika a možnosti vývoje na této platformě. Tato platforma v současné době zaujímá 3,2 % na trhu s mobilními operačními systémy.[22] Číslo se může zdát malé, ale jedná se o 3. nejrozšířenější systém na trhu a společně s prvním Androidem má stále stoupající tendenci na úkor druhému iOS a dalším spíše již stárnoucím systémům (viz Obrázek 4.1).



Obrázek 4.1 Rozdělení mobilních operačních systémů od roku 2012 do prvního čtvrtletí roku 2013. [22]

4.1 Historie

Předchůdcem platformy Windows Phone, konkrétně verze 7, byla platforma Windows Mobile (Pocket PC). Ta je Microsoftem vyvíjena od roku 2000. V roce 2010 došlo na přechod na novější platformu, která ze staré vychází. Platformy nejsou vzájemně kompatibilní. O změně na první pohled nejvíce vypovídá nové uživatelské rozhraní Metro, které bylo kompletně předěláno (viz Obrázek 4.2).

4.1.1 Windows Mobile

Tato mobilní platforma byla nejprve využívána v zařízeních typu PDA, až později došlo k jeho nasazení i v mobilních zařízeních. V systému však bylo velmi mnoho možností jak vývojáři, tak uživatelé měli neskutečný počet možností úprav. Toho výrobci, vývojáři i uživatelé využívali a důsledkem toho byla existence mnoha verzí a nástaveb a úprav systému (tak trochu to připomíná dnešní situaci na platformě Android). Což mělo za následek nestabilitu a pomalost systému. Microsoft se rozhodl začít od začátku a přejít na platformu, která bude mnohem omezenější, jak v roce 2010 bylo zvykem u konkurence, konkrétně u platformy iOS.



Obrázek 4.2 Ukázka úvodní obrazovky Windows Mobile, Windows Phone 7 a Windows Phone 8. [23]

4.1.2 Windows Phone 7

Stejně jako předchůdce Windows Mobile stojí i Windows Phone na stejném, ale vylepšeném jádře, kterým je Windows CE pro architekturu ARM. Předělané je hlavně uživatelské prostředí, které se nově nazývá Metro. Od roku 2010 se toto prostředí postupně rozšiřuje i do ostatních produktů společnosti Microsoft a to hlavně do systému pro konzoli Xbox a na desktopový systém Windows 8, který je nově také možné provozovat na architektuře ARM.

Platforma dostala v průběhu prvního roku aktualizaci na verzi 7.1 a poté na 7.5, která přinesla opravu některých chyb a hlavně přinesla některé základní funkčnosti,

kteřé v systému chyběly. Například funkci kopírování a vkládání textu, podporu HTML 5 v novém Internet Exploreru 9, těsnější propojení se sociálními sítěmi Facebook, Twitter a LinkedIn na úrovni systému a mnoho dalších.

Specifikem platformy je, že Microsoft nastavil minimální konfiguraci zařízení, které může systém provozovat. Později se tyto specifikace rozšířily, aby bylo možné vyrábět i levné modely telefonů (viz Tabulka 4.1).

Displej	dotykový, podporovat alespoň 4 dotyky současně, minimální rozlišení WVGA (480x800 px)
Procesor	Snadragon minimálně architektury ARM v7
Grafický čip	standard DirectX9
RAM	minimálně 256 MB
Úložiště	minimálně 4GB
Fyzická tlačítka	standardní 3 tlačítka (zpět, Start, hledání), vypínací, ovládání hlasitosti a spoušť fotoaparátu
Ostatní	Akcelerometr, senzor osvětlení, senzor přiblížení, A-GPS čip, FM rádio

Tabulka 4.1 Minimální požadavky na hardware pro platformu Windows Phone 7.5. [23]

4.1.3 Windows Phone 8

Další verze nese označení Windows Phone 8 a byla vydána na podzim roku 2012. Špatná zpráva pro majitele zařízení s verzí 7.X byla, že nebudou moci na tuto verzi svůj systém aktualizovat, ale dostanou pouze některá z vylepšení ve verzi 7.8.

Mezi největší změny ve verzi 8 patří [23]:

- nové jádro systému, které bylo převzato z desktopových Windows 8
- podpora vícejádrových procesorů

- podpora více rozlišení displeje (viz Tabulka 4.2)
- podpora paměťových karet
- Internet Explorer 10
- podpora NFC
- podpora Bluetooth pro vývojáře v SDK (stále velmi omezená)

Displej	dotykový, podporovat alespoň 4 dotyky současně, rozlišení: WVGA (800 x 480 px, 15:9), WXGA (1280 x 768 px, 15:9) a 720p (1280 x 720 px, 16:9)
Procesor	Qualcomm Snapdragon S4 dual-core procesor
Grafický čip	standard DirectX9
RAM	Minimálně 512MB pro WVGA telefony, minimálně 1GB RAM pro 720p/WXGA
Úložiště	minimálně 4GB
Fyzická tlačítka	standardní 3 tlačítka (zpět, Start, hledání), vypínací, ovládání hlasitosti a spoušť fotoaparátu
Ostatní	Akcelerometr, senzor osvětlení, senzor přiblížení, A-GPS čip, podpora micro-USB 2.0 a 3,5 mm stereo jack, zadní fotoaparát (automatické ostření) s LED nebo xenonovým bleskem

Tabulka 4.2 Minimální požadavky na hardware pro platformu Windows Phone 8. [23]

V době vývoje klientské aplikace bylo již dostupné SDK pro vývoj na platformu Windows Phone 8, které mimo jiné požaduje jako operační systém na počítači Windows 8, ale nebylo k dispozici zařízení s tímto systémem, na kterém by bylo možné funkčnost otestovat, proto vývoj probíhal pro verzi 7.

4.2 Vývoj aplikací

4.2.1 Přehled

Pro vývoj aplikací pro platformu Windows Phone [23] se používají nástroje, které jsou součástí Windows Phone SDK. Součástí balíčku je i Express edice vývojářského programu Visual Studio (verze 2010 v případě verze WP 7 nebo verze 2012 ve verzi WP 8). Pokud je již Visual Studio nainstalováno, rozšíří se o funkčnost vývoje na WP. Součástí je také emulátor, který částečně dokáže simulovat fyzické zařízení mobilního telefonu.

Pro samotné stažení vývojářského SDK není nutné zřizovat si vývojářský účet, ale pro pozdější publikaci aplikací nebo jejich reálného testování je nutné účet zřídit. Registrace probíhá na adrese <http://create.msdn.com>. Pokud uživatel má Windows Live ID z dřívější doby, registrace není nutná a lze použít svůj původní účet. Aktuální cena za vývojářský účet je 330 Kč za rok. Studenti splňující podmínky programu DreamSpark však mají možnost získat účet bezplatně.

Pro běh emulátoru WP 7 je nutné mít systém Windows Vista SP2 nebo novější. Emulátor nelze spustit ve virtualizovaném systému. U verze WP 8 je nutné dokonce mít nejnovější systém Windows 8, virtualizaci musí podporovat procesor a na počítači nesmí být nainstalovány jiné virtualizační nástroje jako například Oracle VM VirtualBox. Podrobné požadované specifikace jsou k nalezení na [24].

Pro samotný vývoj aplikací se využívají programovací jazyky C# nebo Visual Basic. Oba jazyky jsou si rovny a při vývoji lze dosáhnout stejného výsledku pomocí kteréhokoliv z nich, rozšířenější ale mezi vývojáři je C#.

Dále se používají Frameworky Silverlight a XNA [23]. Silverlight se využívá v klasických aplikacích, které mají uživatelské rozhraní definované pomocí jazyka XAML (čti Zaml) pro prvky jako je vykreslení HTML, přehrávání multimédií a základní ovládací prvky systému. Jazyk XAML je založený na XML a definuje se s ním uživatelské rozhraní. XML soubor lze upravovat ručně nebo lze uživatelské rozhraní upravovat pomocí grafického prostředí, kde umísťujete prvky prostředí pohodlně pomocí myši na pracovní plochu. Framework XNA se naopak využívá tam, kde je potřeba vykreslit složitější grafika typicky ve hrách. Podporuje 2D i 3D grafiku.

Další text práce je zaměřen na programování pomocí jazyka C# s využitím jazyka XAML a frameworku Silverlight, pomocí nichž je vytvořena i výsledná aplikace.

4.2.2 Základní prvky

Panorama a Pivot

Základní element každé aplikace, který se uživateli zobrazuje, je tzv. Page (česky stránka). Tyto stránky jsou navzájem propojeny podobně jako webové stránky a dále se na ně vkládají různé ovládací prvky (seznamy, tlačítka, vstupní formuláře, atd.). Rozmístění prvků je definováno v souborech XAML. Pomocí linku si stránky také předávají různé parametry. Definice jedné stránky a navigace na jinou ukazuje následující ukázky.

```
//definice PlayerPage v XAML
<phone:PhoneApplicationPage
  x:Class="Tagiee.PlayerPage"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:Player="clr-namespace:StreamingClient.Player; "
  FontFamily="{StaticResource PhoneFontFamilyNormal}"
  FontSize="{StaticResource PhoneFontSizeNormal}"
  Foreground="{StaticResource PhoneForegroundBrush}"
  SupportedOrientations="Landscape" Orientation="Landscape"
  mc:Ignorable="d" d:DesignHeight="480" d:DesignWidth="728"
  shell:SystemTray.IsVisible="False">

  <Player:PlayerCtrl Name="PlayerControl"/>
</phone:PhoneApplicationPage>

//navigace na Page Browser
NavigationService.Navigate(new Uri("/Browser.xaml?src=http://" +
_src.Host, UriKind.Relative));
```

Stránky nabývají obvykle jeden ze dvou stylů předpřipravených šablon, které jsou typické pro platformu WP. Jsou jimi Pivot a Panorama (viz Obrázek 4.3). Obě mají záložkový systém. U Pivotu můžeme vidět názvy jednotlivých záložek vedle sebe v horní části displeje. Přesun mezi záložkami je možný klikáním na názvy nebo tažením prstu kdekoli na displeji. U Panorama se část další záložky zobrazuje i při zobrazení první záložky.



Obrázek 4.3 Ukázka rozložení v šabloně Pivot a Panorama. [23]

Popup

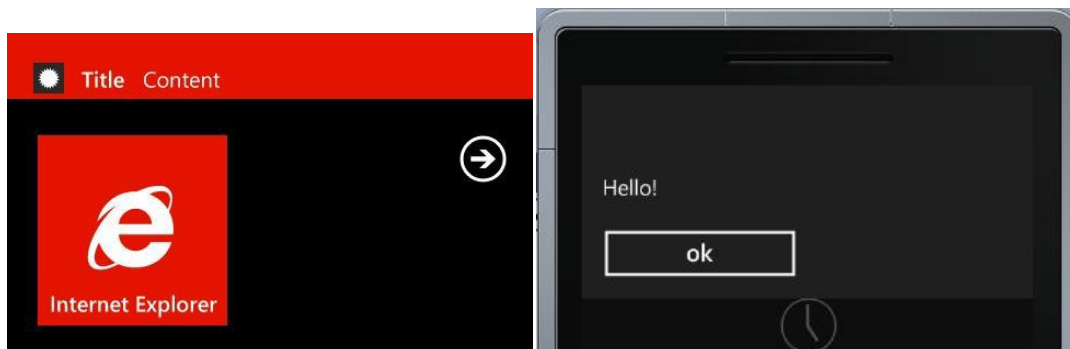
Kromě klasické stránky jako plochy k zobrazování uživatelského rozhraní, existují Popup. Jedná se vyskakovací okna vyvolávána a překrývající klasické stránky. Jejich podoba je opět definována pomocí jazyka XAML. Využívají se jako rychlá možnost k zadávání dat nebo jako potvrzení/odmítnutí některého rozhodnutí.

Toast a MessageBox

Pro rychlé informování například o v pozadí probíhající operace nebo pro různá upozornění slouží notifikace Toast nebo zprávy v MessageBox. Rozdíl mezi těmito dvěma metodami je, že v případě Toast notifikace se zobrazí krátký text, který po 10 sekundách zmizí. V případě MessageBox se zastaví běh aplikace a je vyžadováno potvrzení od uživatele. Tím máme jistotu, že uživatel notifikaci četl (viz následující ukázka a Obrázek 4.4).

```
// Vyvolání Toast a MessageBox.
ShellToast toast = new ShellToast();
toast.Title = "[title]";
toast.Content = "[content]";
toast.Show();

MessageBox.Show("Hello");
```

Obrázek 4.4 Ukázka Toast a MessageBox notifikace.

4.2.3 Model-View-ViewModel

Model-View-ViewModel je návrhový vzor. Tento vzor vychází z principů vzoru MVC a navrhnul jej architekt WPF/Silverlight John Gossman. Nabízí řešení, jak oddělit logiku aplikace od uživatelského rozhraní. ViewModel je prostředníkem mezi View (XAML soubor) a Modelem (datová třída) a zajišťuje obousměrný databinding. Vlastnosti, které chceme, aby se bindovaly při změně, musí být v ModelView definovány jako notifikační, to znamená, že jednoduché typy jsou vystaveny pomocí vlastností typu `DependencyProperty` a kolekce prvků jsou vystaveny pomocí `ObservableCollection<T>`.

4.2.4 Síťová komunikace

Pro komunikaci s okolním světem lze využít hned několik možností. Používá se knihovna `System.Net`.

Ke zjištění dostupnosti připojení k internetu a to jak prostřednictvím WiFi nebo připojením přes GSM brány, slouží `System.Net.NetworkInformation` (viz následující ukázka kódu).

```
//zjištění dostupnosti připojení
NetworkInterface.GetIsNetworkAvailable();
```

Pro komunikaci s webovou službou lze použít `HttpWebRequest`, která reprezentuje HTTP požadavek a následně synchronně nebo asynchronně očekávat `HttpWebResponse`, který představuje HTTP odpověď od serveru. Pro načítání nebo odesílání dat na konkrétní URL lze použít `WebClient`. Lze také vytvářet sockety a vytvořit spojení na bázi klient - server. Sockety mohou využít spojovou službu TCP nebo nespojovou UDP.

4.2.5 Isolated storage a vnitřní databáze

Pro ukládání dat v zařízení lze využít Isolated storage (System.IO.IsolatedStorage), což je vyhrazený prostor pro naši aplikaci, kam nikdo jiný nemá přístup. Samotná aplikace v rámci bezpečnosti nemá přístup ke společnému souborovému systému. Je možné pouze po přidělení oprávnění přistupovat do galerie s obrázky, fotky a videa nebo k hudebním souborům. V našem úložném prostoru můžeme libovolně pracovat s adresáři a soubory pomocí IsolatedStorageFile (viz následující ukázka).

```
//isolované úložiště aplikace
IsolatedStorageFile myIsolatedStorage =
IsolatedStorageFile.GetUserStoreForApplication();
//vytvoření nového souboru
using (StreamWriter writeFile = new StreamWriter(new
IsolatedStorageFileStream(uiid+"_json.txt", FileMode.Create,
FileAccess.Write, myIsolatedStorage)))
    {
        writeFile.WriteLine(json);
        writeFile.Close();
    }
```

Kromě toho můžeme ukládat nastavení formou klíč - hodnota pomocí IsolatedStorageSettings. Nebo lze si v Isolated storage vytvořit lokální databázi pomocí integrované databázové služby SQL CE 3.5 (viz následující ukázka). Technologie LINQ to SQL nám umožňuje snadno k této databázi přistupovat. Překládá dotazy z jazyka LINQ do SQL.

```
//vytvoření tabulky
[Table]
public class InstitutionDb : INotifyPropertyChanged,
INotifyPropertyChanging
{
    [Column(IsPrimaryKey = true, IsDbGenerated = false, CanBeNull =
false, AutoSync = AutoSync.OnInsert)]
    private string _uiid;

    public string Uiid
    {
        get { return _uiid; }
        set
        {
            NotifyPropertyChanging("Uiid");
            _uiid = value;
            NotifyPropertyChanged("Uiid");
        }
    }
}
```

```

    }
}
}
//uložení databáze do lokálního úložiště
private const string DbConnectionString = "Data
Source=isostore:/Institution.sdf";

//práce s daty
public ObservableCollection<InstitutionDb> getAllInstitution()
{
    var temp = from InstitutionDb insDb in Institution
                select insDb;

    Institutios = new ObservableCollection<InstitutionDb>(temp);
    return Institutios;
}

```

4.2.6 Lokalizace

Pro zjištění polohy zařízení se využívá služeb `System.Device.Location`. Poloha se určuje buď orientačně na základě BTS stanic operátora, ke kterým je zařízení připojeno nebo přesně podle signálu GPS. Pro sledování polohy je potřeba vytvořit `GeoCoordinateWatcher`, který sleduje změny v poloze a reagovat na událost změny polohy. Pokud potřebujeme jen zjistit polohu a nesledovat přesnou pohyb zařízení je vhodné po zjištění polohy sledovatele polohy ukončit, aby zbytečně neplýtl energií (viz následující ukázka).

```

//vytvoření sledovatele změny polohy
public void getGpsLocation()
{
    GeoCoordinateWatcher.
    if (this.watcher == null)
    {
        this.watcher = new GeoCoordinateWatcher(GeoPositionAccuracy.High);
        watcher.MovementThreshold = 10;
        this.watcher.StatusChanged += new
        EventHandler<GeoPositionStatusChangedEventArgs>(watcher_StatusChanged);
    }
    //spuštění sledovače
    this.watcher.Start();
}
//událost při změně polohy
void watcher_StatusChanged(object sender,
GeoPositionStatusChangedEventArgs e)
{
    if (e.Status == GeoPositionStatus.Ready)
    {
        GeoCoordinate co = watcher.Position.Location;
    }
}

```

```

lat = co.Latitude;
lon = co.Longitude;
//vypnutí sledovače
watcher.Stop();
}
}

```

4.2.7 Využití funkčnosti interních aplikací

Pokud je potřeba využít funkčnost některé integrované aplikace, jako například vyfotit snímek nebo zavolat na dané číslo, lze použít nástroje Launchers nebo Chooser (Microsoft.Phone.Tasks). Launcher spouští některé předdefinované akce (poslání SMS, volání čísla), oproti tomu Chooser mimo provedení akce také vrátí požadovaný výsledek (vyfotí a vrátí fotografii, vrátí kontakt). Vyvolání některých funkcí je vidět na následující ukázce.

```

//vyvolání napsání emailu
var task = new EmailComposeTask { To = address };
task.Show();

//vyvolání fotoaparátu pro vyfotografování snímku
ctask = new CameraCaptureTask();
ctask.Show();
void ctask_Completed(object sender, PhotoResult e)
{
if (e.TaskResult == TaskResult.OK && e.ChosenPhoto != null)
{
App.CapturedImage = PictureDecoder.DecodeJpeg(e.ChosenPhoto);
foto = e.ChosenPhoto;
}
}
}

```

4.3 Publikace aplikace

V případě, že je aplikace hotová, nastává proces jejího ověření a publikace na Windows Phone Store. Před samotným publikováním je potřeba udělat několik kroků. V projektu aplikace je potřeba vyplnit Assembly Information ve složce Properties. Dalším krokem je volba oprávnění, které potřebujeme pro běh aplikace. Oprávnění se nastavují v souboru WMAppManifest.xml. Aplikaci musíme zkompilovat v režimu Release. Ve Visual Studiu lze před samotnou publikací provést Store Test Kit, zda aplikace projde certifikací a má veškeré náležitosti pro publikování.

Samotná publikace probíhá na adrese <https://dev.windowsphone.com/en-us/AppSubmission/>. V prvním kroku vyplníme jméno naší aplikace a zvolíme

kategorii, popřípadě cenu aplikace. V druhém kroku zvolíme .xap soubor, který vznikl při Release kompilování. Najdeme ho ve složce Bin\Release. Dále je nutné vyplnit číslo verze aplikace, popis aplikace, ikonu aplikace pro obchod a 8 snímků obrazovky s naší aplikací, které se budou zobrazovat v prostředí Windows Phone Store.

Lze zvolit publikace zkušební verze aplikace, která je dostupná hned, ale pouze omezenému množství lidí, které zvolíme (ostatní vývojáři). Druhá možnost veřejného publikování je podmíněná certifikací, kterou uděluje společnost Microsoft. Pokud na naší aplikaci nebude nalezen nějaký problém, tak v průměru po týdnu dostaneme zprávu, že naše aplikace prošla a bude moci být zveřejněna.

Kompletní podmínky certifikace jsou dostupné na adrese [25]

5. Mobilní klient pro Windows Phone

V této kapitole je popsán klient systému informačních majáků, který byl vytvořen pro platformu Windows Phone. Samotný klient je součástí systému nazvaného Tagiee.

5.1 Systém Tagiee

Systém Tagiee se skládá ze serverové části a mobilních klientů. Serverová část vznikla v rámci diplomové práce Systém pro správu a realizaci informačních majáků, kterou vypracoval Bc. Luděk Vlk v roce 2013. Kromě klienta na Windows Phone existuje ještě klient pro platformu Android a iOS. Systém Tagiee tímto tedy pokrývá více jak 95% trhu s mobilními platformami.

Samotný systém funguje jako katalog institucí a informačních majáků umístěných v těchto institucích. Serverová část systému se ovládá přes uživatelsky přívětivé webové rozhraní, kde je možné spravovat instituce i jednotlivé majáky a jejich obsah.

Každá instituce má v katalogu uvedeny základní údaje jako je adresa, popis instituce, otevírací hodiny, vstupné nebo adresu streamovacího serveru. Poté, co se systém začne v instituci používat může správce vidět na webu statistiky přístupů, počty uživatelů a další zajímavá statistická data.

V editaci jednotlivých informačních majáků je možné upravovat jejich obsah, kterým může být text, obrázky nebo videa s odkazem na server YouTube.

Kromě toho se zde definuje, jakým způsobem bude uživatel daný maják moci nalézt. Způsobů, jak lze maják nalézt, je několik.

Možnosti jsou:

- QR kód
- speciální číselný kód
- fyzická poloha
- WiFi otisk
- Bluetooth otisk

Server komunikuje s mobilními klienty prostřednictvím API, které je postaveno na HTTP požadavcích a odpovědích. Je tedy dostatečně obecné, aby podporovalo všechny mobilní platformy.

5.2 Návrh aplikace

Mobilní klient systému Tagiee by měl splňovat následující funkčnost.

5.2.1 Komunikace s Tagiee API

Aplikace bude schopná prostřednictvím internetu komunikovat se serverovou částí systému Tagiee prostřednictvím API. Komunikace probíhá na základě HTTP požadavků, které posílají informace ve formátu JSON a odpovědí ve stejném formátu. Konkrétní podoba implementace jednotlivých požadavků je uvedena kapitole 5.3.1.

Každé zařízení, které komunikuje se serverem, bude jednoznačně identifikovatelné podle speciálního kódu. Tím se zajistí obnovení některých dat (seznam nalezených majáků) i v případě odinstalování a znovu nainstalování aplikace.

5.2.2 Zobrazení dostupných institucí

Na úvodní obrazovce se zobrazí instituce, které nám systém nabídne jako relevantní (v naší blízkosti). Je zapotřebí zjistit polohu zařízení a na základě odeslané polohy zobrazit poskytnutý seznam institucí. V seznamu bude možné zvolit instituci a prohlížet její obsah.

V případě, že nejsou dostupné lokační služby, může uživatel otevřít detail již navštívené instituce i za pomoci speciálního textového kódu, který může být zveřejněn například u vchodu dané aplikace.

Pokud už uživatel danou instituci navštívil, bude si jí moci její detail prohlédnout i v režimu bez internetového připojení. To znamená uložení dat o instituci i o jejích majácích a také textových textového obsahu jednotlivých majáků.

Mimo to bude mít uživatel také možnost stáhnutí a uložení obrázkového materiálu jednotlivých majáků dané instituce do vnitřní paměti mobilního zařízení. To se hodí především, když je zařízení připojeno k WiFi síti a nespotřebovává tak data při přenosu po mobilních sítích GSM. Stažený obsah bude uživatel také moci kdykoliv smazat.

5.2.3 Zobrazení detailu instituce

Po výběru instituce zobrazí aplikace základní údaje jako je název, krátký popis, kontakt, adresu, otevírací hodiny, výši vstupného. Polohu instituce zobrazí také na mapě.

Uživatel také uvidí hodnocení instituce od ostatních uživatelů. Sám bude moci přidat jednoduché textové ohodnocení společně s hvězdičkovým ohodnocením v rozsahu 1-5.

V případě, že uživatel danou instituci již v aplikaci navštívil, jsou data uložena v zařízení. Při opětovné návštěvě zkontroluje aplikace aktuálnost dat podle čísla revize a v případě, že data jsou aktuální, použije data v zařízení. Toto šetří celkový datový provoz aplikace se serverem a i zrychluje běh aplikace. V případě neaktuálnosti dat a dostupnosti připojení se vždy stáhnou aktuální data.

5.2.4 Hledání informačních majáků

Aplikace v zobrazení detailu aplikace nabídne možnosti pro hledání informačních majáků. Z důvodu omezených možností Windows Phone API jsou k dispozici tři způsoby identifikace majáků.

- QR kód - Uživatel bude moci využít funkce fotoaparátu k naskenování a vyhodnocení QR kódu. Při vyhodnocení správného kódu umožní aplikace přístup k obsahu majáku.
- poloha - Aplikace na základě lokačních služeb, ať už signálu GPS, triangulace GSM sítě nebo dostupnosti WiFi sítí, zjistí aktuální polohu zařízení. Na základě polohy se vyhodnotí, zda se nachází v blízkém okolí (cca 20metrů) některého z majáků a pokud ano, zobrazí se mu poloha i majáky na mapě.
- speciální číselný kód - Poslední variantou je možnost zadat speciální trojmístný číselný kód, který je jedinečný pro každý maják.

Využití při identifikaci technologie Bluetooth, NFC a WiFi neumožňuje vývojáři omezené API.

Mimo výše zmíněné možnosti přístupu k obsahu informačního majáku se uživateli bude zobrazovat ještě seznam již navštívených majáků. Uživatel si je tedy bude moci prohlédnout i kdykoliv později, i když už se poblíž nich nevyskytuje.

5.2.5 Zobrazení obsahu majáku

V případě nalezení majáku se uživateli zobrazí jeho obsah. Maják může obsahovat textový popis, hypertextové odkazy, obrázky, odkazy na video na serveru YouTube. Dále u majáku bude přístup ke streamovaným videím z lokálního VLC serveru, které poskytne provozovatel instituce.

5.2.6 Odesílání statistik

V průběhu používání aplikace bude sama posílat statistické údaje. Jedná se o informaci o navštívení instituce nebo nalezení majáku společně s informací, jakou technologií maják našel. V případě nedostupnosti připojení se odešlou statistiky při jeho dostupnosti.

5.3 Implementace

Aplikace byla implementována pomocí jazyka C# a vytvořena pro platformu Windows Phone ve verzi 7. Důvod, proč nebyla vyvíjena nejnovější verzi 8, je nedostupnost testovacího zařízení v době vývoje.

5.3.1 Komunikace s API Tagiee

Ke komunikaci se serverovou částí systému Tagiee komunikuje aplikace pomocí HTTP požadavků a odpovědí (`HttpRequest`, `HttpResponse`) ve formátu JSON. Všechny požadavky jsou vyřizovány na adrese `http://api.tagiee.com`. Veškerá komunikace se serverem probíhá ve třídě `Communicator`.

API reaguje na tyto požadavky:

get_institution_list - tímto požadavkem aplikace žádá seznam institucí v systému Tagiee, které se nacházejí ve 100 km okolí od zařízení. V požadavku je odeslána zeměpisná šířka a délka polohy zařízení. Jako odpověď server odešle seznam institucí se základními informacemi a revizí informující o aktuálnosti dat.

get_institution_by_uid - tímto požadavkem aplikace žádá o veškeré informace k jedné konkrétní instituci, která se identifikuje podle unikátního parametru UID. Kromě základních údajů posílá server i kompletní seznam informačních majáků dané instituce i s jejich kompletním textovým obsahem a odkazy na multimediální obsah.

get_institution_revision - tímto požadavkem aplikace žádá o číslo revize jedné konkrétní instituce. Aplikace tak může zjistit, zda má u sebe uložená nejaktuálnější data o instituci.

device_visit_institution - tímto požadavkem aplikace odesílá informaci o navštívení konkrétní instituce v aplikaci. Informace se na serveru shromažďují a vytváří se z nich statistika.

device_found_beacon - tímto požadavkem aplikace odesílá informaci o nalezení konkrétního majáku identifikovatelného podle jedinečného parametru UBID v určité instituci. Informace se na serveru shromažďují a vytváří se z nich statistika a také se zpětně posílá informace o nalezených majácích na konkrétním zařízení v případě například přeinstalování aplikace.

get_reviews - tímto požadavkem aplikace žádá o seznam hodnocení jedné konkrétní instituce.

add_review - tímto požadavkem aplikace odesílá hodnocení aplikace ke konkrétní instituci. Jedno zařízení může odeslat pouze jedno hodnocení ke každé instituci.

Kompletní přehled požadavků a odpovědí ve formátu JSON je uveden v Příloze A.3.

5.3.2 Databáze a Isolated storage

Pro ukládání informací pro použití v režimu bez připojení, využívá aplikace vnitřní databázi a vnitřní soukromé datové úložiště.

Databáze má tyto 4 samostatné entity:

BeaconDB

Entita představuje informaci o nalezení majáků v určité instituci.

- `string Ubid` - identifikátor majáku
- `string Uiid` - identifikátor instituce, do které maják patří.
- `boolean Save` - informace o tom, jestli byla informace odeslána na server

- `string Time` - čas nalezení majáku
- `string Tech` - technologie, kterou byl maják nalezen

InstitutionDB

Entita představuje informaci o aktuálnosti dat o dané instituci. Ve vnitřním úložišti je pak uložen textový soubor s názvem odpovídající UUID instituce, který obsahuje kompletní JSON odpověď ze serveru.

- `string Uuid` - identifikátor instituce
- `int Revision` - číslo revize dat

ResourceDB

Entita představuje informace o obrázcích, které jsou obsahem majáků a které byly stáhnuty do vnitřního úložiště, aby se nemusely stahovat při každém prohlížení.

- `int ResourceId` - identifikátor obrázku
- `string Path` - cesta k uloženému obrázku
- `string Uuid` - identifikátor instituce, které patří maják, jehož obsahem je daný obrázek.

ReviewDB

Entita představuje hodnocení, které napsal uživatel o instituci a to se v případě nedostupnosti připojení nemohlo odeslat ihned.

- `string Uuid` - identifikátor instituce
- `boolean Save` - informace o úspěšném odeslání na server
- `string Time` - čas vytvoření hodnocení
- `string Text` - text hodnocení
- `int Rating` - číselné hodnocení

Ke každé entitě existuje třída `DataContext`, která zajišťuje operace nad vzniklými tabulkami.

Na vnitřní úložiště se ukládají mimo textových souborů s informacemi o institucích také stažené obrázky, které jsou obsahem majáků.

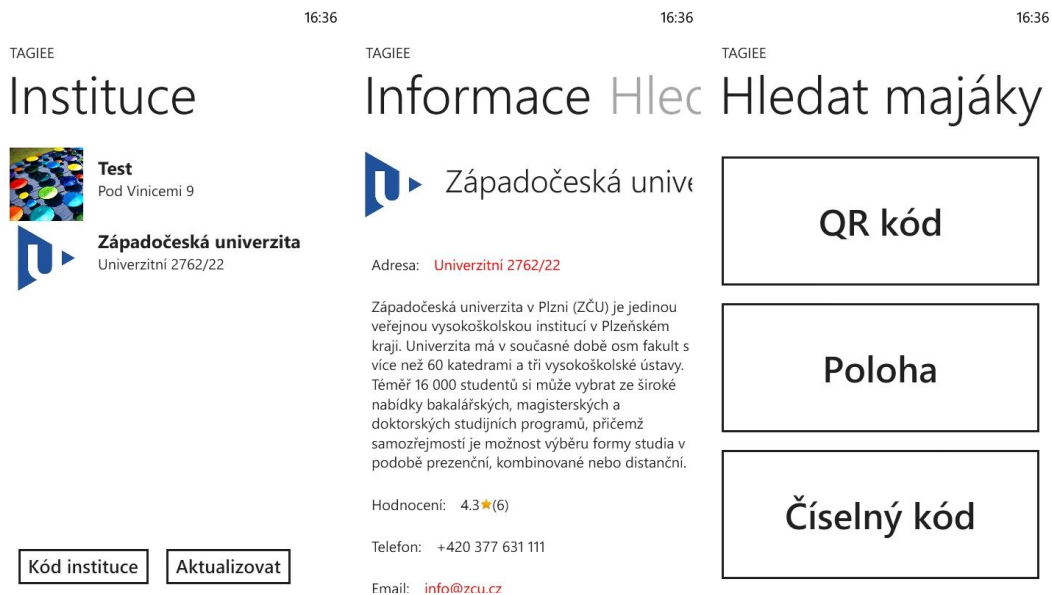
5.3.3 Jedinečná identifikace

Jelikož si systém Tagiee uchovává informace o tom, jaké majáky byly na konkrétním zařízení nalezeny, je nutné každé zařízení jednoznačně identifikovat. Jelikož aplikace existuje na 3 mobilních platformách, byla pro identifikaci zvolena externí knihovna, která je dostupná pro všechny 3 platformy. Knihovna `openUDID` [26] generuje jedinečný 32 znakový identifikátor napříč všemi třemi nejpoužívanějšími mobilními platformami. Použití v aplikaci je velice jednoduché (viz následující ukázka).

```
// použití knihovny
using OpenUDIDPhone;
//vygenerování identifikátoru
string udid = OpenUDID.value;
```

5.3.4 Návrh uživatelského prostředí

Uživatelské prostředí se skládá z jednotlivých stránek (viz Obrázek 5.1 a 5.2), do kterých je někdy vloženo vyskakovací okno. Na stránkách je použita šablona `Pivot` a stránky `InstitutionPage` a `BeaconPage` mají více záložek.



Obrázek 5.1 Ukázka úvodní obrazovky, detailu instituce a možností hledání majáků.



Obrázek 5.2 Ukázka seznamu nalezených majáků, obsahu majáku a možnosti uploadovat fotografii.

Stránky `MainPage`, `InstitutionPage` a `LoctionBeaconMapPage` jsou navíc propojené s vlastním `ViewModelem` z důvodu měnících se dat na těchto stránkách. Při změně není třeba pak měnit jednotlivé položky v uživatelském rozhraní, ale ty se pomocí `bindingu` změní automaticky.

Ukázka ukazuje MainPage, která je spojena s MainViewModel. Je nutné, aby položky a seznamy položek byly definovány jako typ DependencyProperty, respektive ObservableCollection<T>

```
//nastavení DataContext stránky na MainViewModel
DataContext = App.MainViewModel;
this.InstitutionList.ItemsSource = App.MainViewModel.InstitutionList;

//načtení dat z ViewModelu
App.MainViewModel.LoadData(this);

//metoda načtení dat v MainViewModel
public void LoadData(MainPage mainPage)
{
    App.Comunicator.requestInstutionList();
    //data přijdou z Comunicator, kde jsou přijata ze serveru
    this.InstitutionList = App.Comunicator.InstitutionList;
    this.IsDataLoaded = true;
}

//data binding v seznamu institucí
<ListBox x:Name="InstitutionList" Margin="0,0,0,100"
SelectionChanged="InstitutionList_SelectionChanged"
    <ListBox.ItemTemplate >
        <DataTemplate>
            <StackPanel>
                <Image Name="logo" Source="{Binding logo}" Grid.Column="1/>
                <StackPanel Orientation="Vertical" >
                    <TextBlock x:Name="name" Text="{Binding name}"/>
                    <TextBlock x:Name="address" Text="{Binding address}/>
                </StackPanel>
            </StackPanel>
        </DataTemplate>
    </ListBox.ItemTemplate>
</ListBox>
```

Na stránkách MapPage a LocationBeaconMapView je použita komponenta Maps. Pro plnohodnotné zobrazení mapových podkladů Bing map je potřeba zaregistrovat aplikaci na webu <http://www.bingmapsportal.com>. Po registraci je aplikaci přidělen API klíč, který je potřeba v aplikaci uvést při použití map.

Kompletní návod na zprovoznění Bing map v aplikaci je dostupný na adrese [27].

5.3.5 Hledání majáků

Hledání majáků je aplikaci možné třemi způsoby:

QR kód

Pro rozpoznání QR kódů byla zvolena externí knihovna, která rozpoznává kódy v reálném čase. Jedná se o knihovnu Windows Phone 7 Silverlight ZXing Barcode Scanning Library [28]. Aplikace využívá komponentu fotoaparátu a každých 250 milisekund zaznamená scénu. Knihovna se snaží na scéně naleznout QR kód a v případě úspěchu poskytne výsledek (viz ukázka). Výsledek je následně porovnán s identifikátory dostupných majáků. V případě shody umožní vstup do majáku.

```
//inicializace časovače
_timer = new DispatcherTimer();
_timer.Interval = TimeSpan.FromMilliseconds(250);
_timer.Tick += (o, arg) => ScanPreviewBuffer();
//inicializace kamery
_photoCamera = new PhotoCamera();
_photoCamera.Initialized += OnPhotoCameraInitialized;
_previewVideo.SetSource(_photoCamera);
//inicializace QR čtečky
_reader = new QRCodeReader();

//metoda analyzující snímky
private void ScanPreviewBuffer()
{
    try
    {
        _photoCamera.GetPreviewBufferY(_luminance.PreviewBufferY);
        var binarizer = new HybridBinarizer(_luminance);
        var binBitmap = new BinaryBitmap(binarizer);
        var result = _reader.decode(binBitmap);
        //v metodě foundQR už se pracuje s výsledkem analýzy
        Dispatcher.BeginInvoke(() => foundQr(result.Text));
    }catch(){...}
}
```

Poloha

Při hledání majáků pomocí polohy se zjistí aktuální poloha zařízení a vykreslí se na mapu. Ze známých poloh jednotlivých majáků se vypočítají vzdálenosti od zařízení a na mapu se vykreslí takové, které mají vzdálenost menší než cca 20 metrů. Mimo mapy zobrazí blízké majáky v seznamu, ze kterého lze na jednotlivé majáky přejít.

Speciální číselný kód

Při hledání majáků pomocí číselného kódu se pouze nabídne uživateli formulář pro vložení kódu. Kód je následně porovnán s kódy dostupných majáků v instituci. V případě schody aplikace přejde na maják.

5.3.6 Stažení obsahu

Pro stažení multimediálních dat se využívá `WebClient`, který se připojí na URL se zdrojovým souborem a ten uloží do vnitřního úložiště jak je vidět na následující ukázce.

```
//odeslání požadavku webového klienta
_webClient.OpenReadAsync(new Uri(url));

//čtení URL
_webClient.OpenReadCompleted += (s1, e1) =>
{
    ...
//vytvoření nového souboru v úložišti a zapsání bytů.
using (var isfs = new
IsolatedStorageFileStream(directory+"/"+fileName, FileMode.CreateNew, isf))
    {
        long fileLen = e1.Result.Length;
        byte[] b = new byte[fileLen];
        e1.Result.Read(b, 0, b.Length);
        isfs.Write(b, 0, b.Length);
        isfs.Flush();
    }
    ...
}
```

5.3.7 Streamování videa

Streamování videí je v aplikaci řešeno dvěma způsoby.

Jedním způsobem distribuce videa je streamování videí ze serveru YouTube. V době psaní diplomové práce byla z Windows Phone Store stažena aplikace od společnosti Microsoft, která umožňovala pohodlné přehrávání videí v telefonu. Důvodem stažení aplikace byly neshody společnosti Microsoft s majitelem serveru YouTube společností Google. Google svou oficiální aplikaci pro platformu WP nevydal. Z důvodu neexistující oficiální aplikace pro tento server je nutné videa otevírat v prohlížeči navštívením mobilních stránek webu. Až zde je možné spustit požadované video v integrovaném přehrávači. Microsoft v době psaní začal vyvíjet novou aplikaci

pro portál YouTube, která by již neměla být v rozporu s podmínkami společnosti Google.

Druhá možnost streamování videí připadá na streamovací server VLC. Streamovací server VLC běží na lokální síti v instituci a návštěvníci s aplikací Tagiee se mohou pomocí technologie WiFi sítě připojit do této lokální sítě a prohlížet dostupná videa. Konfigurace VLC serveru je popsána v Příloze A.2. Server poskytuje videa na požádání (VOD). Seznam dostupných videí aplikace získává pomocí Webového rozhraní VLC serveru v podobě XML dokumentu, který je rozparsován a ve vhodné podobě zobrazen uživateli (viz následující ukázka).

```
//požadavek na obsah adresáře na adrese _src
_vlc = new VlcAccess(_src.Host + ":" + _src.Port);
_vlc.SendChangeDirCommand(_src.LocalPath.Substring(1));

public void SendChangeDirCommand(string dirPath)
{
    //sestavení požadavku
    string text = string.Format("http://{0}/requests/browse.xml?dir={1}",
        _host, dirPath);
    WebRequest webRequest = WebRequest.Create(text);
    webRequest.BeginGetResponse(new AsyncCallback(BrowseResponseCallback),
        webRequest);
    RaiseAsyncRequestStartedEvent();
}
//zpracování odpovědi
private void BrowseResponseCallback(IAsyncResult result)
{
    HttpWebRequest httpWebRequest = (HttpWebRequest)result.AsyncState;
    try
    {
        HttpResponseMessage httpResponse = httpWebRequest.EndGetResponse(result)
        as HttpResponseMessage;
        if (httpResponse.StatusCode == HttpStatusCode.OK)
        {
            using (Stream responseStream = httpResponse.GetResponseStream())
            {
                using (StreamReader streamReader = new StreamReader(responseStream))
                {
                    //čtení XML souboru
                    XmlSerializer xmlSerializer = new
                    XmlSerializer(typeof(BrowseInfo));
                    BrowseInfo = (xmlSerializer.Deserialize(streamReader) as
                    BrowseInfo);
                    var ms = new MemoryStream();
                    var sr = new StreamWriter(ms);
```

```

        //xml serializace
        xmlSerializer.Serialize(sr, new BrowseInfo() { Elements = new
element[] { new element() } });
        ms.Position = 0;
        var sr2 = new StreamReader(ms);
        var s = sr2.ReadToEnd();
        //výsledek
        element[] items = new element[BrowseInfo.size];
        BrowseInfo.Elements = items;
    }
}
}
}
}
catch (Exception ex)
{...}
}

```

Samotné přehrávání probíhá pomocí knihovny VLC Streaming Client [29] (viz následující ukázka). Knihovna je potřeba, protože platforma WP podporuje streamování videí pouze ve formátu VC-1 s kodekem WMV v kontejneru ASF, ale tento formát neumí VLC server vytvořit. VLC server tedy streamuje video zakódované H.264 a zvuk zakódovaný v MP3 v kontejneru MPEG-4. Video i audio zakódované těmito kodeky již zařízení přehrát umí, proto knihovna přijímá tento stream a následně kontejner otevře a rozdělí na jednotlivé složky a ty pak již přehrává v integrovaném přehrávači. Knihovna funguje jako dekodér MPEG-4 kontejneru.

```

//na stránku je potřeba přidat přehrávač z knihovny
<Player:PlayerCtrl Name="PlayerControl"/>

//spuštění přehrávání soubu uvedeného v parametru src.
PlayerControl.Start(src, server.StreamingSettings);

```

5.3.8 Nahrání fotografií

V rámci Oborového projektu byla aplikace a server Tagiee rozšířeny o funkčnost nahrávání fotografie od uživatele. Pro vyfocení snímku je použit `CameraCaptureTask`, který vyvolá výchozí aplikaci fotoaparátu a po vyfocení navrátí fotografii. Fotografie v podobě streamu bytů je převedena na řetězec bytů. Pomocí HTTP požadavku (viz následující ukázka) je řetězec bytů odeslán jako POST parametr na adresu "http://upload.tagiee.com/index.php?uiid=XXX", kde XXX představuje identifikátor instituce pro správné zařazení snímku. Na serveru Tagiee skript fotografii uloží do složky /upload.

```

//vytvoření a spuštění tasku pro vyfocení fotografie
ctask = new CameraCaptureTask();
ctask.Completed += new EventHandler<PhotoResult>(ctask_Completed);
ctask.Show();

//zpracování fotografie
void ctask_Completed(object sender, PhotoResult e){
if (e.TaskResult == TaskResult.OK && e.ChosenPhoto != null)
    {
        App.CapturedImage = PictureDecoder.DecodeJpeg(e.ChosenPhoto);
    }

//odeslání fotografie
public void uploadFile(string uuid, System.IO.Stream stream)
{
byte[] bytes = ReadToEnd(stream);
//vytvoření HTTP požadavku
HttpRequest request = (HttpRequest)WebRequest.Create(address +
uuid);
request.Method = "POST";
request.ContentType = "application/x-www-form-urlencoded";
string postData = String.Format("image={0}",
System.Convert.ToBase64String(bytes));
//odeslání dat
request.BeginGetRequestStream(result =>
    {
        using (var requestStream = request.EndGetRequestStream(result))
            {
                using (StreamWriter writer = new
StreamWriter(requestStream))
                    {
                        writer.Write(postData);
                        writer.Flush();
                    }
            }
    }
}
}

```

5.3.9 Oprávnění aplikace

Seznam oprávnění, které aplikace potřebuje ke svému běhu:

ID_CAP_IDENTITY_DEVICE - pro jednoznačnou identifikaci zařízení.

ID_CAP_ISV_CAMERA - pro použití zadní kamery pro čtení QR kódů.

ID_CAP_LOCATION - pro určení polohy zařízení.

ID_CAP_MEDIALIB - pro přístup ke knihovně s médii pro streamování videí.

ID_CAP_NETWORKING a ID_CAP_WEBBROWSERCOMPONENT - pro zajištění komunikace se serverem.

5.3.10 Upgrade aplikace pro WP8

Pro bezproblémový běh aplikace na mobilních zařízeních s WP 8 by se musela aplikace upravit, jelikož nová verze SDK není zpětně kompatibilní. Změny proběhly u důležitých částí API, které aplikace využívá jako například určování polohy nebo síťová komunikace, což jsou stěžejní prvky aplikace Tagiee.

5.3.11 Schválení a publikování aplikace

Aplikace nebyla v prvním kole žádosti o publikaci certifikována pro zveřejnění na Windows Phone Store z důvodu nemožnosti aplikaci otestovat. Nemožnost je způsobená testováním aplikace mimo dostupnost jakékoli blízké instituce. Aplikace tedy zobrazovala prázdný seznam. Aplikaci byla upravena aby zobrazila hlášku o nedostupnosti institucí. Doporučená lokace pro testování byla při odesílání aplikace sdělena.

Možnosti jak aplikaci nainstalovat jsou vypsány v uživatelské příručce v Příloze A.1.

6. Ověření funkčnosti

Klientské aplikace pro mobilní platformu byla ověřena na instituci Západočeské univerzity v Plzni v systému informačních majáků Tagiee.

6.1 Instituce Západočeská univerzita v Plzni.

Instituce byla předem vytvořena pomocí webového rozhraní systému Tagiee na adrese <http://tagiee.com>.

- Přístupové jméno: "zcu"
- Přístupové heslo: "102030"

V instituci je celkem vytvořeno 12 majáků s vlastním obsahem. Každý maják lze identifikovat pomocí speciálního číselného kódu, QR kódu nebo NFC tagu. Majáky se dají rozdělit na dvě kategorie:

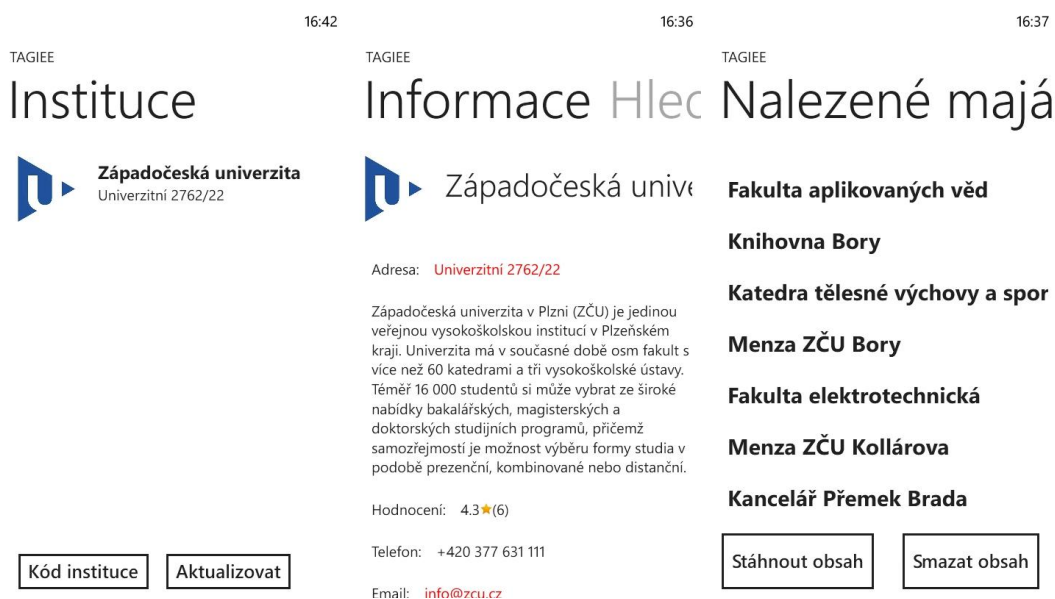
- **venkovní majáky** (odpovídající budovám) - tyto majáky se navíc dají nalézt pomocí souřadnic GPS a některé i pomocí WiFi sítí.
 - Fakulta aplikovaných věd (Univerzitní 22)
 - Katedra tělesné výchovy a sportu (Univerzitní 14)
 - Fakulta elektrotechnická (Univerzitní 26)
 - Menza ZČU Bory (trojúhelník Univerzitní)
 - Knihovna Bory (Univerzitní 20)
 - Menza ZČU Kollárova (Kollárova 19)
- **vnitřní majáky** (odpovídající místnostem ve Fakultě aplikovaných věd, 4.patro a 3. patro) tyto majáky se navíc dají nalézt pomocí WiFi sítí a některé pomocí Bluetooth signálu.
 - Katedra informatiky a výpočetní techniky
 - Učebna počítačové grafiky (UL-407)
 - Učebna počítačových sítí (UL-402)
 - EEG ERP učebna (UU-401)
 - Kancelář Doc. Ing. Přemysl Brada, MSc. Ph.D. (UK-422)
 - Laboratoř digitální fabriky (UL-309)

Každý maják má svůj obsah, který informuje o daném místě. Mimo popisu mohou obsahovat užitečné odkazy, obrázky či videa.

6.2 Klientská aplikace Tagiee pro platformu Windows Phone

Mobilní aplikace byla ověřena na zmíněné instituci Západočeské univerzity (viz Obrázek 6.1 a 6.2). Z důvodu omezenosti API pro vývoj aplikace na platformě WP byly použity a ověřeny tři možnosti pro hledání majáků. Aplikace všechny majáky, všemi dostupnými metodami, našla a zobrazila korektně jejich obsah.

Aplikace byla testována na zařízení HTC Maza s WP 7.5. Byly otestovány stavy, kdy aplikace nemá přístup k internetu nebo nejsou dostupné lokační služby. V případě nedostupnosti lokačních služeb lze nalézt instituci Západočeské univerzity pomocí speciálního kódu "zcu" na úvodní obrazovce aplikace za předpokladu, že instituci zařízení již dříve navštívilo.



Obrázek 6.1 Ukázka instituce v seznamu institucí, detail instituce a seznamu nalezených majáků.



Fakulta aplikovaných věd

Knihovna Bory

Fakulta aplikov, Fakulta aplikov,

Fakulta aplikovaných věd je jednou z fakult Západočeské univerzity v Plzni. Ve státem akreditovaných studijních programech připravuje vysoce kvalifikované odborníky, kteří nemají problém s uplatněním na trhu práce.



Fakulta aplikovaných věd je jednou z fakult Západočeské univerzity v Plzni. Ve státem akreditovaných studijních programech připravuje vysoce kvalifikované odborníky, kteří nemají

programy, doktorské studijní programy a mohou provádět i habilitační řízení a řízení ke jmenování profesorem.

Díky spolupráci a dobrým vztahům se zahraničními univerzitami nabízí fakulta svým studentům možnost zahraničních stáží v mnoha zemích celého světa.

Vědecko-výzkumná činnost

<http://fav.zcu.cz/vyzkum-a-vyvoj/>



Koukněte na video na Youtube

Obrázek 6.2 Ukázka vyhledávání majáků na mapě a zobrazení obsahu majáku Fakulta aplikovaných věd.

7. Závěr

V rámci práce byl prozkoumán trh se systémy poskytující informace. Především byly zkoumány systémy, které mají klientskou aplikaci pro platformu Windows Phone. Je potřeba podotknout, že takovýchto aplikací je stále méně než na ostatních mobilních platformách. Obecně jde ale říct, že se tyto systémy budou do budoucna rozšiřovat.

Práce poskytuje přehled používaných formátů, kodeků a kontejnerů pro všechny typy multimédií a jejich podporu na mobilních platformách. Pro fotografie se obvykle na mobilních zařízeních používá formát JPEG a pro ostatní grafiky formát PNG. U práce se zvukem už není situace tak přehledná, přesto nejrozšířenější platformy podporují nejrozšířenější formát MP3 a jeho přímého nástupce AAC. U práce s videem je situace asi nejsložitější ze všech typů multimédií. Nejlepší volbou bude formát MPEG-4 ASP a MPEG-4 AVC známější pod názvem H.264.

Práce nabízí přehled možností distribuce multimédií ze serveru na mobilní zařízení. Nejvíce se pak zabývá streamingem, síťovými protokoly podporující přenos multimédií po síti a konkrétními streamovacími servery.

Programování pro platformu Windows Phone není složité, jelikož vše je dobře dokumentované společností Microsoft. Co vývojáře potěší trochu méně je omezenost API pro vývoj a všude přítomné limity. Nekompatibilita verzí 7 a 8 je také nepříjemná, jelikož je nutné ladění stejného kódu dvakrát.

Výsledkem práce je klientská aplikace pro platformu Windows Phone, která pracuje se systémem informačních majáků Tagiee. Funkčnost aplikace byla ověřena na instituci Západočeské univerzity v Plzni. Hledání informačních majáků je možné pomocí lokačních služeb, QR kódů a zadáváním speciálního číselného kódu. Aplikace také umožňuje přehrávat streamovaná videa ze serveru VLC.

Seznam zdrojů a literatury

- [1] Trip Advisor *Windows Phone Store* [online]. Poslední revize 2013. Dostupné z <http://www.windowsphone.com/en-us/store/app/tripadvisor/180b0f46-e753-e011-854c-00237de2db9e>
- [2] Trips in Paris *Windows Phone Store* [online]. Poslední revize 2013. Dostupné z <http://www.windowsphone.com/en-us/store/app/trips-in-paris/4ad3ff42-b3cb-4e4c-888f-c232a386a0a5>
- [3] Zoo Guide *Windows Phone Store* [online]. Poslední revize 2013. Dostupné z <http://www.windowsphone.com/en-us/store/app/zoo-guide/4b5fb14f-d70f-4dba-8013-013e723a14cc>
- [4] iZeeum *iTunes* [online]. Poslední revize 2013. Dostupné z <https://itunes.apple.com/gb/app/izeeum/id635025006?mt=8>
- [5] Central Park (Listen to the Light) *iTunes* [online]. Poslední revize 2013. Dostupné z <https://itunes.apple.com/us/app/central-park-listen-to-light/id468193258?mt=8>
- [6] STEPHEN WEINSTEIN. *The Multimedia Internet*. New York: Springer, 2005. ISBN: 0-387-23-681-3.
- [7] PAVEL STRACHOTA, Ukládání a komprese obrazu [online]. Poslední revize 15.9.2010. Dostupné z http://saint-paul.fjfi.cvut.cz/base/public-filesystem/admin-upload/POGR/POGR1/07.ukladani_a_komprese_obrazu.pdf
- [8] JPEG [online]. Poslední revize 2007. Dostupné z <http://www.jpeg.org/>
- [9] PAVEL TIŠŇOVSKÝ, Ztrátová komprese obrazových dat pomocí JPEG [online]. Poslední revize 15.9.2010. Dostupné z <http://www.root.cz/clanky/ztratova-komprese-obrazovych-dat-pomoci-jpeg/>
- [10] PAVEL TIŠŇOVSKÝ, PNG is Not GIF [online]. Poslední revize 14.12.2006. Dostupné z <http://www.root.cz/clanky/png-is-not-gif/>
- [11] Supported Media Formats *Android developers* [online]. Poslední revize 2013. Dostupné z <http://developer.android.com/guide/appendix/media-formats.html>

- [12] UIImage Class Reference *iOS Developer Library* [online]. Poslední revize 23.4.2013. Dostupné z http://developer.apple.com/library/ios/documentation/UIKit/Reference/UIImage_Class/Reference/Reference.html
- [13] Supported media codecs for Windows Phone *Windows Phone Dev Center* [online]. Poslední revize 2013. Dostupné z [http://msdn.microsoft.com/en-us/library/windowsphone/develop/ff462087\(v=vs.105\).aspx](http://msdn.microsoft.com/en-us/library/windowsphone/develop/ff462087(v=vs.105).aspx)
- [14] PETR LOBAZ, Uložení a komprese zvuku [online]. Poslední revize 12.3.2013. Dostupné z http://www.kiv.zcu.cz/~lobaz/mhs/prednasky2013/mhs_2013_05.pdf
- [15] Media Layer *iOS Developer Library* [online]. Poslední revize 19.9.2012. Dostupné z <http://developer.apple.com/library/ios/documentation/miscellaneous/conceptual/iphoneostechoverview/MediaLayer/MediaLayer.html>
- [16] WES SIMPSON. Video Over IP. London: Focal Press, 2008. ISBN: 978-0-240-81084-3.
- [17] Windows Media Components for QuickTime *Windows* [online]. Poslední revize 2013. Dostupné z <http://windows.microsoft.com/en-us/windows/windows-media-components-quicktime>
- [18] A. C. FONG, S. C. HUI. Multimedia Engineering. London: John Wiley & Sons, Ltd, Research Studies Press Limited, 2006. ISBN-13 978-0-470-03019-6 (HB), ISBN-10 0-470-03019-4 (HB).
- [19] Internet Information Services [online]. Poslední revize 2013. Dostupné z <http://www.iis.net/>
- [20] Comparing HTTP Streaming Protocol with RTSP *Developer Network* [online]. Poslední revize 28.8.2008. Dostupné z <http://msdn.microsoft.com/en-us/library/bb905764.aspx>
- [21] VLC Media Player [online]. Poslední revize 2013. Dostupné z <http://www.videolan.org/>

- [22] Android and iOS Combine for 92.3% of All Smartphone Operating System Shipments in the First Quarter While Windows Phone Leapfrogs BlackBerry, According to IDC *IDC* [online]. Poslední revize 16.5.2013. Dostupné z <http://www.idc.com/getdoc.jsp?containerId=prUS24108913>
- [23] Windows Phone Dev Center [online]. Poslední revize 16.5.2013. Dostupné z <https://dev.windowsphone.com/en-us/develop>
- [24] System requirements for Windows Phone Emulator *Windows Phone Dev Center* [online]. Poslední revize 2.8.2013. Dostupné z [http://msdn.microsoft.com/en-us/library/windowsphone/develop/ff626524\(v=vs.105\).aspx](http://msdn.microsoft.com/en-us/library/windowsphone/develop/ff626524(v=vs.105).aspx)
- [25] App certification requirements for Windows Phone *Windows Phone Dev Center* [online]. Poslední revize 2.8.2013. Dostupné z [http://msdn.microsoft.com/en-US/library/windowsphone/develop/hh184843\(v=vs.105\).aspx](http://msdn.microsoft.com/en-US/library/windowsphone/develop/hh184843(v=vs.105).aspx)
- [26] OpenUDID *GitHub* [online]. Poslední revize 7.7.2013. Dostupné z <https://github.com/ylechelle/OpenUDID>
- [27] Using Bing Maps *Develop Network* [online]. Poslední revize 2013. Dostupné z <http://msdn.microsoft.com/en-us/hh641334>
- [28] Windows Phone 7 Silverlight ZXing Barcode Scanning Library *CodePlex* [online]. Poslední revize 18.1.2011. Dostupné z <http://silverlightzxing.codeplex.com/>
- [29] VLC Streaming Client Library for Windows Phone *CodePlex* [online]. Poslední revize 17.7.2012. Dostupné z <https://v1cwp7.codeplex.com/>

Seznam zkratek

WP - Windows Phone

NFC - Near Filed Communication

GPS - Global Positioning System

QR - Quick Response

BTS - Base Transceiver Station

GSM - Globální Systém pro Mobilní komunikaci

SVG - Scalable Vector Graphics

RLE - Run-length encoding

LZ77 - Lempel-Ziv 77

LZW - Lempel-Ziv-Welch

BMP - Bitmap

GIF - Graphics Interchange Format

JPEG - Joint Photographic Experts Group

JFIF - JPEG File Interchange Format

ISO - International Organization for Standardization

DCT - discrete cosine transform

MDCT - Modified discrete cosine transform

EXIF - Exchangeable image file format

PNG - Portable Network Graphics

TIFF - Tagged Image File Format

ICO - ikona

ALAC - Apple Lossless encoder

FLAC - Free Lossless Audio Codec

MPEG - Motion picture experts Group

AAC - Advanced Audio Coding

ASF - Advanced Systems Format

RIFF - Resource Interchange File Format

ASP - Advanced Simple Profile

AVC - Advanced Video Coding

FUP - Fair User Policy

VoIP - Voice over Internet Protocol

HTTP - Hypertext Transfer Protocol

RTSP - Real Time Streaming Protocol

TCP - Transmission Control Protocol

UDP - User Datagram Protocol

RFC - Request for comments

RTP - Real-time Transport Protocol

RTCP - RTP Control Protocol

MMS - Microsoft Media Server

HTML5 - HyperText Markup Language 5

URL - Uniform Resource Locator

IIS - Internet Information Services

PDA - Personal digital assistant

ARM - Acorn RISC Machine

RAM - Random-access memory

SDK - Software development kit

API - Application Programming Interface

XAML - Extensible Application Markup Language

WPF - Windows Presentation Foundation

SQL - Structured Query Language

LINQ - Language Integrated Query

JSON - JavaScript Object Notation

XML - Extensible Markup Language

Seznam obrázků

Obrázek 2.1 Aplikace TripAdvisor. Zobrazení bodů zájmu v okolí. Zajímavosti v okolí Plzně.....	12
Obrázek 2.2 Aplikace Trip in Paris.....	13
Obrázek 2.3 Zoo Guide. Plánek Zoo. Menu. Detail zvířete s popisem a možností přehrát audio záznam.....	14
Obrázek 2.4 iZeeum. Princip použití: Vyfocení, odeslání, analýza a za závěr poskytnutí multimediálního obsahu.....	14
Obrázek 3.1 Přehled operací při kompresi JPEG. [9].....	19
Obrázek 3.2 Přehled protokolů na transportní vrstvě. [16].....	30
Obrázek 3.3 Přehled možností streamování serveru VLC. [21].....	33
Obrázek 4.1 Rozdělení mobilních operačních systémů od roku 2012 do prvního čtvrtletí roku 2013. [22].....	34
Obrázek 4.2 Ukázka úvodní obrazovky Windows Mobile, Windows Phone 7 a Windows Phone 8. [23].....	35
Obrázek 4.3 Ukázka rozložení v šabloně Pivot a Panorama.[23].....	40
Obrázek 4.4 Ukázka Toast a MessageBox notifikace.....	41
Obrázek 5.1 Ukázka úvodní obrazovky, detailu instituce a možností hledání majáků...53	
Obrázek 5.2 Ukázka seznamu nalezených majáků, obsahu majáku a možnosti uploadovat fotografii.....	53
Obrázek 6.1 Ukázka instituce v seznamu institucí, detail instituce a seznamu nalezených majáků.....	62
Obrázek 6.2 Ukázka vyhledávání majáků na mapě a zobrazení obsahu majáku Fakulta aplikovaných věd.....	63

Seznam tabulek

Tabulka 3.1 Podpora jednotlivých formátů obrázků na mobilních platformách. Kompletní podpora formátů a kodeků pro Android je uvedena na [11], pro iOS [12] a pro Windows phone [13].....	20
Tabulka 3.2 Podpora jednotlivých formátů a kodeků pro práci se zvukem na mobilních platformách. Kompletní podpora formátů a kodeků pro Android je uvedena na [11], pro iOS [15] a pro Windows phone [13].....	23
Tabulka 3.3 Podpora jednotlivých formátů a kodeků pro práci s videem na mobilních platformách. Kompletní podpora formátů a kodeků pro Android je uvedena na [11], pro iOS [15] a pro Windows phone [13].....	27
Tabulka 4.1 Minimální požadavky na hardware pro platformu Windows Phone 7.5. [23].....	36
Tabulka 4.2 Minimální požadavky na hardware pro platformu Windows Phone 8. [23].....	37

Přílohy

A.1 Uživatelská příručka mobilního klienta

1. Instalace aplikace

Instalace se provede stažením aplikace z Windows Phone Store. V době psaní této diplomové práce nebyla aplikace ještě certifikována, a proto nebyla v obchodě zveřejněna.

Druhou možností je stáhnout aplikaci z adresy:

<http://www.windowsphone.com/s?appid=c5bb7a9d-9353-419d-a9b6-c4ed3810c5af>

Zde je určená pro testování, a proto jde stáhnout pouze po přihlášení pod tímto Microsoft účtem.

- uživatelské jméno: tagiee@outlook.com
- heslo: gh564df684h

Na přiloženém CD je aplikace také dostupná v adresáři Tagiee, lze jí nainstalovat ručně podle návodu na adrese:

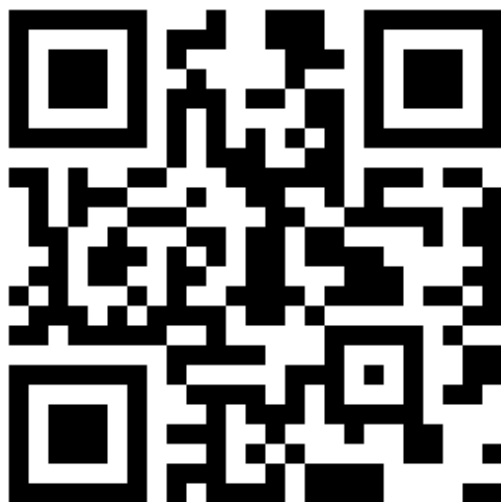
<http://www.windowsphone.com/cs-cz/how-to/wp8/apps/how-do-i-install-apps-from-an-sd-card>

2. Používání aplikace

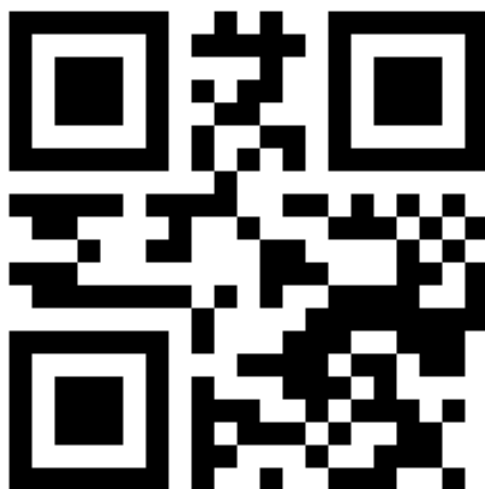
Pro plnohodnotný běh aplikace je nutné mít dostupné internetové připojení a mít povoleny lokační služby. Aby aplikace mohla nalézt instituci, nesmí se nacházet dále než 100 km od adresy instituce. Pro hledání majáku je možné využít polohy zařízení, QR kódy a speciální číselný kód.

Instituci Západočeská univerzita nalezneme v okolí Plzně. Po prvním navštívení instituce jí můžeme vyhledat pomocí kódu "zcu". Pro hledání majáku pomocí polohy jděte do areálu ZČU na Borech. Pro hledání pomocí QR kódu a speciálního číselného kódu použijte obrázky níže.

V prohlížení obsahu majáků lze měnit obrázky klikáním na obrázky. V záložce multimédia lze nahrát svojí fotografii na server Tagiee a v případě nakonfigurování VLC serveru a dostupnosti stejné lokální sítě, lze přehrávat streamovaná videa.



Fakulta aplikovaných věd, kód: 391

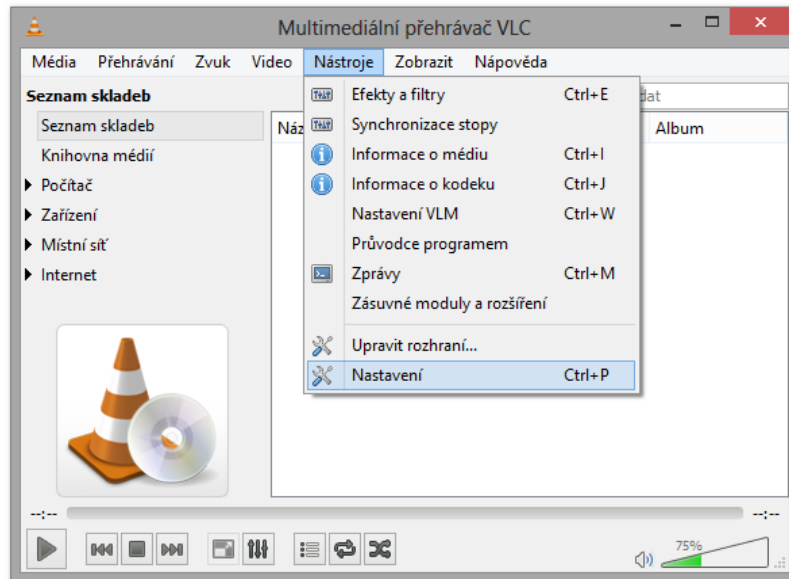


Knihovna bory, kód 384

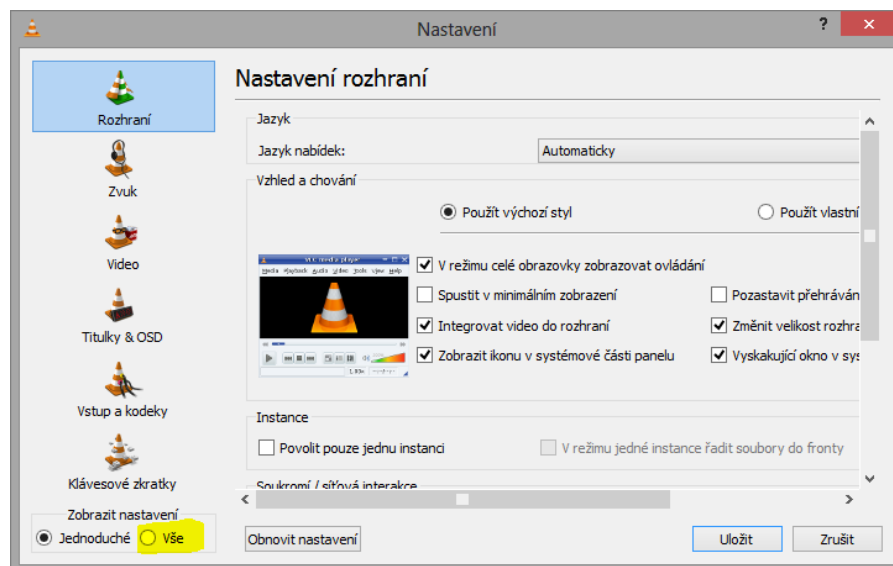
A.2 Návod na nastavení VLC

Zde je popsán návod na nastavení serveru VLC pro lokální streamování videí do mobilní aplikace Tagiee. Pro návod byl použit přehrávač ve verzi 2.0.7. nainstalovaný na systému Windows 8. Ke stáhnutí je dostupný na adrese <http://www.videolan.org/>.

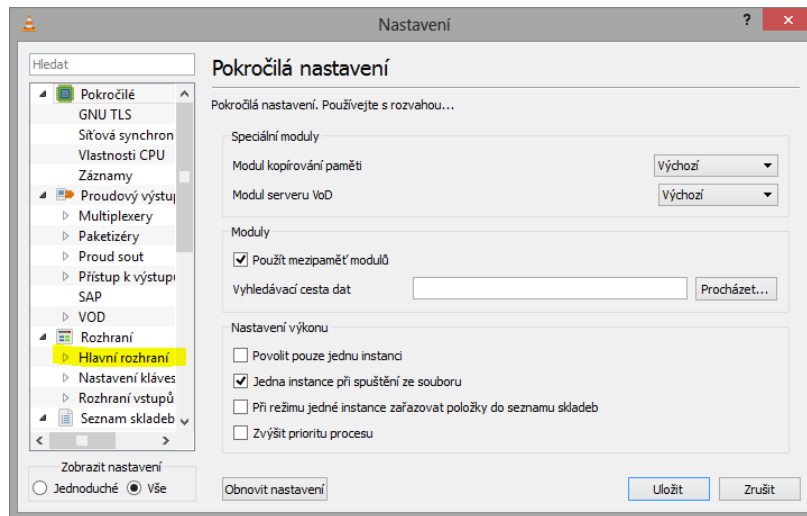
1. Spusťte VLC a v liště zvolte Nástroje → Nastavení.



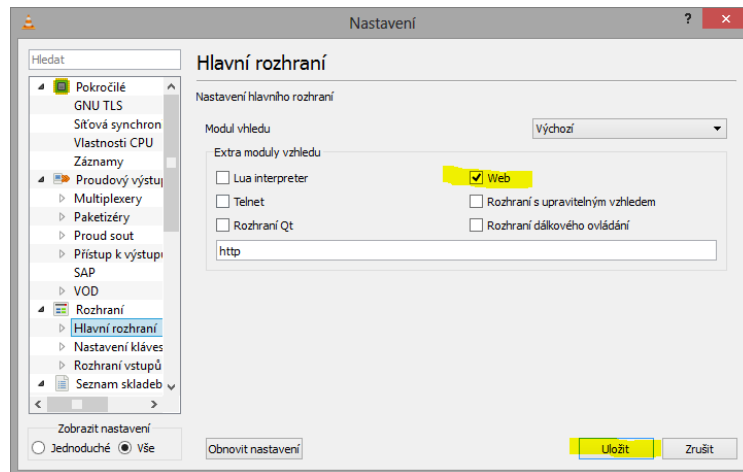
2. V Nastavení zvolte Zobrazit nastavení možnost Vše.



3. Poté položku Hlavní rozhraní.



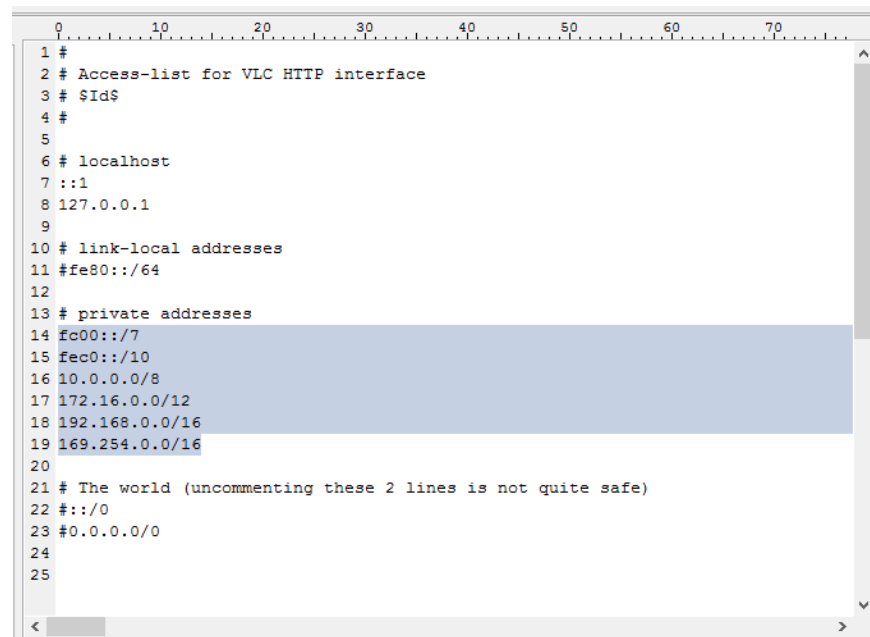
4. Zde zvolte možnost Web a uložte.



5. Otevřte soubor .host v adresáři:

- Windows: “%PROGRAMFILES%\VideoLAN\VLC\lua\http\.hosts”
- Linux: “/usr/share/vlc/lua/http/.hosts”
- MacOS “/Applications/VLC.app/Contents/MacOS/share/lua/http/.hosts”

6. V souboru odkomentujte řádky (smažte #) s lokálními adresami. Kompletní nastavení webového přístupu je dostupný zde: http://wiki.videolan.org/Documentation:Modules/http_intf



```
0 10 20 30 40 50 60 70
1 #
2 # Access-list for VLC HTTP interface
3 # $Id$
4 #
5
6 # localhost
7 ::1
8 127.0.0.1
9
10 # link-local addresses
11 #fe80::/64
12
13 # private addresses
14 fc00::/7
15 fec0::/10
16 10.0.0.0/8
17 172.16.0.0/12
18 192.168.0.0/16
19 169.254.0.0/16
20
21 # The world (uncommenting these 2 lines is not quite safe)
22 #::/0
23 #0.0.0.0/0
24
25
```

7. Zjistěte IP adresu počítače/serveru. (V systému Windows spustit program cmd a vložit příkaz ipconfig).
8. Zadejte IP adresu počítače/serveru společně s cestou k adresáři s videi do nastavení instituce na webu <http://tagiee.com/> jako položku streamovací server. Například <http://10.0.0.139:8080/C:/videa>, kdy jsou videa uložena v adresáři a podadresářích C:/videa.



- [🏠 Zpět na seznam institucí](#)
- [📄 Informace a Statistiky](#)
- [✎ Editovat Instituci](#)
- [📍 Majáky](#)
- [📍 Lokace](#)
- [🖼️ Obrázky](#)
- [★ Uživatelské hodnocení](#)

Logo Instituce:		Upravit
Název Instituce:	Západočeská univerzita	Upravit
Popis Instituce:	Západočeská univerzita v Plzni (ZČU) je jedinou veřejnou vysokoškolskou...	Upravit
Země:	Czech Republic	Upravit
Kraj:	Plzeňský	Upravit
Město:	Plzeň	Upravit
Adresa:	Univerzitní 2762/22	Upravit
PSČ:	301 00	Upravit
Tel:	+420 377 631 111	Upravit
Email:	info@zcu.cz	Upravit
Adresa stream serveru:	http://10.0.0.139:8080/Cs:/video	Upravit
Adresa stream serveru: <input type="text" value="http://10.0.0.139:8080/Cs:/video"/>		
Uložit Zrušit		
Editovat souřadnice:	Latitude: 49.7248467 / Longitude: 13.3509363	Upravit

A.3 Seznam JSON požadavků a odpovědí

V této příloze je uveden kompletní seznam požadavků a odpovědí ve formátu JSON pro komunikaci s API systému Tagiee (<http://api.tagiee.com>).

add_review - tímto požadavkem aplikace odesílá hodnocení aplikace ke konkrétní instituci. Jedno zařízení může odeslat pouze jedno hodnocení ke každé instituci.

```
[type] => add_review
[udid] => string - unikatni identifikator zarizeni
[uiid] => string - instituce, ke ktore se hodnoceni pridava
[review] => string - kratke slovni hodnoceni
[rating] => int
[platform] => string {ios|android|win}
[version] => string - ve formatu x.x.x
```

ODPOVED

```
[status] => boolean
```

device_found_beacon - tímto požadavkem aplikace odesílá informaci o nalezení konkrétního majáku identifikovatelného podle jedinečného parametru UBID v určité instituci. Informace se na serveru shromažďují a vytváří se z nich statistika a také se zpětně posílá informace o nalezených majácích na konkrétním zařízení v případě například přehodnocení aplikace.

```
[type] => device_found_beacon
[udid] => string - unikatni identifikator zarizeni
[uiid] => string - instituce, ke ktore se pozadavek tyka
[tech] => string {gps|qr|nfc|key} - pouzita technologie, neni povinna
[discovered_datetime] => datetime
[platform] => string {ios|android|win}
[version] => string - ve formatu x.x.x
```

ODPOVED

```
[status] => boolean
```

device_visit_institution - tímto požadavkem aplikace odesílá informaci o navštívení konkrétní instituce v aplikaci. Informace se na serveru shromažďují a vytváří se z nich statistika.

```
[type] => device_visit_institution
[udid] => string - unikatni identifikator zarizeni
[uiid] => string - instituce, ke ktore se pozadavek tyka
[discovered_datetime] => datetime
```

```
[platform] => string {ios|android|win}
[version] => string - ve formátu x.x.x
```

ODPOVED

```
[status] => boolean
```

get_institution_by_uid - tímto požadavkem aplikace žádá o veškeré informace k jedné konkrétní instituci, která se identifikuje podle unikátního parametru UIID. Kromě základních údajů posílá server i kompletní seznam informačních majáků dané instituce i s jejich kompletním textovým obsahem a odkazy na multimediální obsah.

```
[type] => get_institution_by_uid
[udid] => string - unikátní identifikátor zařízení
[uuid] => string - instituce, ke které se požadavek týká
[platform] => string {ios|android|win}
[version] => string - ve formátu x.x.x
```

ODPOVED

```
[status] => boolean
[response]=>
(
  [revision] = int
  [logo] => string - url obrázku
  [uuid] => string
  [name] => string
  [description] => string
  [address] => string
  [lat] => double
  [lon] => double
  [tel] => string
  [support_email] => string
  [prices] =>
    (
      [1..N array]
        [type] => string
        [price] => string
        [currency] => string
    )
  [opening_hours] =>
    (
      [1..N array]
        [days] => string
        [time] => string
    )
  [allow_offline] => boolean
  [allow_uploads] => boolean
  [allow_reviews] => boolean
  [number_reviews] => int
```

```

[rating] => float
[user_posted_review] => boolean
[beacons] =>
(
  [1..N array] =>
  (
    [ubid] => string
    [name] => string
    [description] => string
    [lat] => double
    [lon] => double
    [key] => int
    [additional] =>
    (
      [1..N array] => - sila signalu urcena poradim
      (
        [tech] => string
        [data] => string
      )
    )
  )
  [content] =>
  (
    [1..N array] =>
    (
      [type] => string {image|text|audio|video|youtube|href}
      [heading] => string
      [content] => string
      [resource_id] =>
      (
        [0..N array] => int
      )
    )
  )
  [discovered] => boolean
)
)
[resources] =>
(
  [1..N array] =>
  (
    [id] => int
    [type] => string
    [size] => double
    [path] => string
    [metadata] =>
    (
      [width] => int
      [height] => int
      [caption] => string
    )
  )
)

```



```
)  
)  
)
```

get_institution_list - tímto požadavkem aplikace žádá seznam institucí v systému Tagiee, které se nacházejí ve 100km okolí od zařízení. V požadavku je odeslána zeměpisná šířka a délka polohy zařízení. Jako odpověď server odešle seznam institucí se základními informacemi a revizí informující o aktuálnosti dat.

```
[type] => get_institution_list  
[udid] => string - unikatni identifikator zarizeni  
[platform] => string {ios|android|win}  
[version] => string - ve formatu x.x.x  
[lat] => double  
[lon] => double
```

ODPOVED

```
[status] => boolean  
[response]=>  
(  
    [1..N array]  
    [revision] = int  
    [logo] => string - url obrazku  
    [uuid] => string  
    [name] => string  
    [address] => string  
    [lat] => double  
    [lon] => double  
    [rating] => double  
    [number_reviews] => int  
)
```

get_institution_revision - tímto požadavkem aplikace žádá o číslo revize jedné konkrétní instituce. Aplikace tak může zjistit zda má u sebe uložená nejaktuálnější data o instituci.

```
[type] => get_institution_revision  
[udid] => string - unikatni identifikator zarizeni  
[uuid] => string - instituce, ke ktere se pozadavek tyka  
[platform] => string {ios|android|win}  
[version] => string - ve formatu x.x.x
```

ODPOVED

```
[status] => boolean  
[response]=>  
(  
    [revision] = int
```

```
)
```

get_reviews - tímto požadavkem aplikace žádá o seznam hodnocení jedné konkrétní instituce.

```
[type] => get_reviews  
[udid] => string - unikatni identifikator zarizeni  
[uiid] => string - instituce, ke ktere se pozadavek tyka  
[platform] => string {ios|android|win}  
[version] => string - ve formatu x.x.x  
[from] => int - spodni limit  
[to] => int - horni limit
```

ODPOVED

```
[status] => boolean  
[response]=>  
(  
  [1..N array]  
  [added] => string  
  [rating] => double  
  [review] => string  
)
```