

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

DISTRIBUCE MÍSTNĚ ZÁVISLÝCH INFORMACÍ V MOBILNÍCH APLIKACÍCH

DIPLOMOVÁ PRÁCE

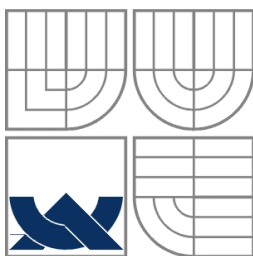
MASTER'S THESIS

AUTOR PRÁCE

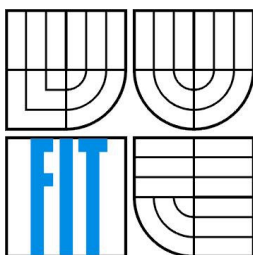
AUTHOR

Bc. Tomáš Bátrla

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DISTRIBUCE MÍSTNĚ ZÁVISLÝCH INFORMACÍ V MOBILNÍCH APLIKACÍCH

LOCATION BASED INFORMATION DISTRIBUTION IN MOBILE APPLICATIONS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. Tomáš Bátorla

VEDOUCÍ PRÁCE
SUPERVISOR

Doc. Dr. Ing. Petr Hanáček

BRNO 2007

Zadání diplomové práce

Řešitel: **Bátrla Tomáš, Bc.**

Obor: Počítačová grafika a multimédia

Téma: **Distribuce místně závislých informací v mobilních aplikacích**

Kategorie: Počítačová grafika

Pokyny:

1. Seznamte se s mobilními technologiemi a metodami zjišťování polohy využitelnými pro distribuci multimediálních dat.
2. Zhodnoťte možnosti použití různých typů určení polohy.
3. Navrhněte klient-server architekturu pro získávání lokačně závislých multimediálních dat.
4. Implementujte prototyp navržené architektury.
5. Ověřte funkčnost implementace nad ukázkovými daty.

Literatura:

- Podle pokynů vedoucího

Při obhajobě semestrální části diplomového projektu je požadováno:

- Body zadání 1-3

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

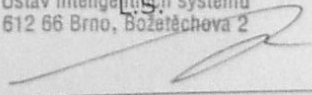
Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Hanáček Petr, doc. Dr. Ing., UITS FIT VUT**

Datum zadání: 28. února 2006

Datum odevzdání: 22. května 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav inteligentních systémů
602 00 Brno, Božetěchova 2



doc. Dr. Ing. Petr Hanáček
vedoucí ústavu

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Bc. Tomáš Bátorla**
Id studenta: 49274
Bytem: Jiráskova 1813, 755 01 Vsetín
Narozen: 02. 10. 1982, Vsetín
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

Článek 1

Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
diplomová práce

Název VŠKP: Distribuce místně závislých informací v mobilních aplikacích
Vedoucí/školitel VŠKP: Hanáček Petr, doc. Dr. Ing.
Ústav: Ústav inteligentních systémů
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v:

tištěné formě počet exemplářů: 1
elektronické formě počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2 Udělení licenčního oprávnění


1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užit, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3 Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejimu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....
Nabyvatel


.....
Autor

Abstrakt

Tato diplomová práce se zabývá získáváním multimediálních dat ze serveru v závislosti na poloze mobilního telefonu a vyfotografovaného dvojrozměrného grafického kódu. V práci jsou popsány platformy mobilních zařízení, technologie pro zjištění polohy a pro rozpoznávání grafických kódů. Dále je zde popsán návrh klient-server architektury, která umožňuje mobilním zařízením získat multimediální informace závislé na poloze. Na konci práce jsou shrnutky mé poznatky při reálném testování vytvořeného systému a nastíněny možnosti dalšího zlepšení systému pro distribuci místně závislých informací.

Klíčová slova

GSM, SOAP, RSS, LBS, GPS, 2D kódy, Java

Abstract

Master thesis is focused on retrieval of multimedia information based on location of mobile device. Paper describes mobile platforms technology and technology behind location sensing and two-dimensional barcode recognition. There is also included design of client-server architecture for information retrieval of location based multimedia data. Master thesis includes report of system testing and some ideas about new features for implemented location based information distribution system.

Keywords

GSM, SOAP, RSS, LBS, GPS, 2D barcodes, Java

Citace

BÁTRLA, Tomáš. *Distribuce místně závislých informací v mobilních aplikacích*. [s.l.], 2007. 62 s. VUT-FIT. Diplomová práce.

Distribuce místně závislých informací

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Doc. Dr. Ing. Petra Hanáčka.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Tomáš Bátorla
20.05.2007

Poděkování

Děkuji svému vedoucímu diplomové práce panu Hanáčkovi za odbornou pomoc a cenné rady, které mě pomohli poznat oblast mobilních aplikací a získat tak velmi atraktivní zkušenosti.

© Tomáš Bátorla, 2007.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Úvod.....	2
1 Mobilní, lokalizační a webové technologie	3
1.1 Mobilní technologie	3
1.1.1 J2ME	3
1.1.2 Flash Lite	4
1.1.3 Multimediální data	5
1.1.4 Grafické kódy	6
1.2 Webové technologie.....	9
1.2.1 SOAP	9
1.2.2 RSS	10
1.3 Lokalizace mobilních zařízení	11
1.3.1 Obecné principy.....	12
1.3.2 Lokalizace na úrovni mobilní sítě.....	17
1.3.3 Satelitní lokalizace.....	20
2 Aplikace pro distribuci místně závislých dat.....	24
2.1 Specifikace požadavků	24
2.2 Návrh aplikace	25
2.2.1 Webové rozhraní	25
2.2.2 Klient mobilního zařízení.....	27
2.2.3 Serverová část.....	33
2.2.4 Vzájemná komunikace komponent.....	36
2.3 Implementace.....	37
2.3.1 Mobilní aplikace.....	37
2.3.2 Serverová část.....	38
3 Testování	40
3.1 Použité výpočetní prostředky.....	40
3.1.1 Mobilní platformy	40
3.1.2 Serverová platforma.....	42
3.2 Testování implementovaného systému.....	43
3.2.1 Test příjmu GPS signálu	43
3.2.2 Test kvality dekodování 2D čárových kódů.....	44
3.2.3 Test rychlosti zpracování	47
4 Zhodnocení	49
4.1 Výhody a nevýhody implementace	49
4.2 Návrh vylepšení	50
5 Závěr	53

Úvod

Aplikace využívající znalost polohy jsou užitečné v každodenním životě. Mohou navigovat náš automobil a zjednodušit naši dopravu, stejně jako nás při procházce cizím městem mohou upozorňovat na zajímavé objekty okolo nás. Navigační systémy do auta či pro osobní použití jsou dnes populární aplikace. Služby závislé na poloze nám pomáhají zkoumat svět okolo nás. Stejně jako lokační služby získávají své příznivce, vzniká vedle nich nová technologie - dvojrozměrné čárové kódy, které můžeme stále častěji nacházet vytisknuté na věcech každodenních potřeby.

Diplomová práce uvádí čtenáře do problematiky mobilních i webových technologií a metod zjišťování polohy mobilních zařízení a rozpoznávání grafických kódů. Cílem práce je navrhnout a implementovat systém, který bude jako klíčový prvek využívat znalost polohy mobilního telefonu a internetovou adresu, kterou mobil získá na základě rozkódování vyfotografovaného grafického kódu. Na základě těchto informací získá klientský software relevantní multimediální data, která jsou vztažena k danému místu. Data mohou být reprezentována jako obrázky, text nebo zvuk. Klientská část bude implementována v prostředí mobilního telefonu. Z důvodu kompatibility vezmeme při návrhu a implementaci ohled na skutečnost, že klientská verze bude pracovat na mnoha zcela odlišných typech mobilních telefonů. Nyní se tedy pojďme seznámit s technologiemi, které nám umožní požadovanou aplikaci vytvořit.

1 Mobilní, lokalizační a webové technologie

1.1 Mobilní technologie

Mobilní telefony se za posledních deset staly užitečnou pomůckou a můžeme se s nimi setkat téměř u kohokoliv. Mobil je možno rozšířit doinstalováním různých aplikací, přičemž k tomuto účelu jsou terminály vybaveny například prostředím pro spuštění programů jazyka Java nebo FlashLite. Nyní se trochu více seznámíme s technologiemi, se kterými se můžete setkat i ve svém mobilním telefonu.

1.1.1 J2ME

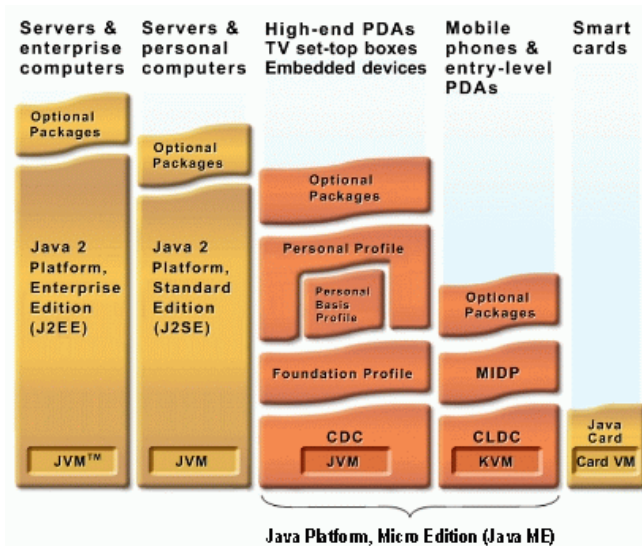
J2ME (Java 2 microedition) byla vyvinuta firmou Sun Microsystems. Důvodem byla potřeba vytvářet aplikace pro prostředí mobilních telefonů, které nabízeli pouze malý výpočetní výkon a paměťovou kapacitu. Java ME platforma je kolekce technologií a specifikací, které jsou zkombinovány takovým způsobem, aby vytvořili kompletní Java Runtime prostředí bezchybně fungující v mobilních zařízeních. Technologie je založena na několika elementech [1].

- Konfigurace poskytuje základní množinu knihoven a schopností virtuálního stroje pro širokou paletu mobilních zařízení.
- Profil je množinou API (aplikační programové rozhraní), které jsou podporovány užší skupinou mobilních zařízení.
- Rozšiřitelná knihovna je množina programovacích API, které jsou úzce spjata s daným mobilním zařízením.

Časem se vydělily dvě základní konfigurace [1]. Konfigurace CLDC (konfigurace mobilního zařízení s omezeným připojením) je navržena pro zařízení s malou paměťovou kapacitou a výpočetním výkonem. Konfigurace CDC (konfigurace mobilního zařízení s plným připojením) je určena pro lépe vybavené mobilní telefony a set-top boxy (zařízení pro připojení k televizi a příjem digitálního signálu). CLDC definuje základy programovacích API a virtuální stroj pro zařízení, která nemají dostatek zdrojů. Mezi tato zařízení patří mobilní telefony, pagery a PDA (personální digitální asistent).

Jakmile je CLDC zkombinováno s MIDP profilem, poskytuje solidní Java platformu k vytváření aplikací pro zařízení s omezenými zdroji výpočetního výkonu, energie a paměti. Široce rozšířeným příkladem je kombinace konfigurace CLDC a profilu MIDP (profil mobilní edice Javy).

MIDP poskytuje kompletní prostředí pro mobilní telefony a zařízení s podobnými schopnostmi. Prostor CLDC a MIDP poskytují aplikace nazývané se MIDlety [2], které jsou vytvořeny v prostředí J2ME.



Obrázek 1 - Struktura platformy Java ME
grasia.fdi.ucm.es/j2me/images/j2meLayers.jpg

1.1.2 Flash Lite

Flash Lite byl původně vyvinut firmou Macromedia, která jej vydala v roce 2003. Tato mobilní platforma poskytuje vývojářům Flash aplikací možnost programovat pro mobilní telefony v prostředí Flash. Zanedlouho po uvedení přišlo vylepšení ve formě nových funkcí jazyka ActionScript. Flash Lite 1.0 a 1.1 jsou původně založeny na Flash Player 4.0. Technologie Flash Lite 2.1 je postavena na Flash player 8.0. Flash lite podporuje skriptovací programovací jazyk ActionScript.

Flash Lite 2.x podporuje zpracování XML (rozšiřitelný značkovací jazyk), ukládání dat do perzistentní paměti a podporu SVG (škálovatelná vektorová grafika) a další. Velkou výhodou je skutečnost, že jakýkoliv program vytvořený v prostředí Flash je možno spustit i na mobilním telefonu. Flash je vhodný pro práci se streamovanými multimediálními daty. Existují dvě základní implementace použitelné s mobilními telefony. První je „stand alone“ model, který umožňuje spouštění Flash aplikací uložených v mobilním zařízení. Tento přístup je vhodný pro větší aplikace jako jsou například mobilní hry.

Druhou variantou, která je použita v mobilních telefonech firmy Sony Ericsson, je model založený na spouštění flash aplikace v internetovém prohlížeči mobilního telefonu. Zde Flash Player funguje jako rozšíření standardního internetového prohlížeče. Uživatel k aplikaci přistupuje jejím nahráním přes mobilní datovou síť.

1.1.3 Multimediální data

Multimédia mohou být považována za formu informačního obsahu a zpracování informací, který má za úkol informovat nebo pobavit uživatele. V prostředí mobilních aplikací se setkáváme i s multimediálními formáty, které byli speciálně vytvořeny pro zařízení s malou paměťovou a výkonnostní kapacitou. V dalším textu je uveden výčet několika základních datových formátů pro obraz. Konfigurace MIDP2.0 umožňuje mobilním zařízením pracovat s grafickými formáty JPEG, GIF a PNG. Prostředí Flash Lite poskytuje stejné datové formáty. Nyní si popíšeme základní formáty, které se pro ukládání grafických dat v prostředí mobilních aplikací používají.

GIF

GIF je zkratkou sousloví Graphics Interchange Format. Jedná se o grafický formát využívající 8 bitovou barevnou hloubku. Obraz ve formátu GIF může obsahovat maximálně 256 barev z palety 16.7 miliónu barev. Formát implementuje bezztrátovou kompresi obrazových dat a také umožňuje uložit v jednom souboru informace o více obrazech, které lze postupně zpracovávat ve formě animace.

JPEG

Jedná se o velmi rozšířený obrazový formát, který nabízí schopnost vysoké komprese za cenu určité ztráty obrazových dat. Grafický formát JPEG definuje vlastní kompresní algoritmus, který převádí obraz na řetězec bytů. Stejně tak je definován datový formát pro uložení komprimovaných dat, který se nazývá JPEG interchange format. Výhodou JPEG je skutečnost, že může být implementován i do jiných grafických formátů [4]. Můžeme se tak setkat s JPEG kompresí v obrazovém formátu TIFF.

Nyní trochu popíši proces kódování použitý ve formátu JPEG. Na počátku je vstupní obraz konvertován z barevného prostoru RGB (červená, zelená, modrá) do YCbCr (žlutá, chrominance modrá, chrominance červená). Y obsahuje informace o intenzitě a CbCr je hodnota zvaná chrominance. Nový barevný prostor redukuje množství dat, která nesou informaci o skutečné barvě. Dále je obraz doplněn prázdnými hodnotami do velikosti, kdy je jeho šířka i výška dělitelná 8. Obraz je rozdělen do bloků o velikosti 8x8 pixelů. Běžně se přistupuje k podvzrokování chrominance na poloviční velikost.

V dalším kroku nastupuje dvojrozměrná diskrétní kosinová transformace, která je aplikována na každý dříve rozdělený blok. Tímto postupem získáme bloky, které popisují frekvenční oblast zpracovávaného obrazu. Určité frekvenční oblasti se zanedbají a tímto způsobem můžeme zredukovat množství informace popisující obraz. Po redukci barevného prostoru jsou bloky zkomprimovány.

PNG

PNG je obrazový datový formát, který implementuje bezeztrátovou kompresi dat. Na rozdíl od grafického formátu GIF, obsahuje PNG vždy pouze jeden obraz a nepodporuje uložení animací [3]. PNG podporuje 24 bitovou barevnou paletu a obrázky v odstínech šedi. Výhodou formátu PNG je skutečnost, že použitý kompresní algoritmus není licencován.

1.1.4 Grafické kódy

Dvojrozměrné čárové kódy - grafické značky, jsou pro výpočetní systémy jednoduše čitelné. Ve světě existuje mnoho standardních i proprietárních dvojrozměrných kódů. V Japonsku jsou velmi rozšířené tzv. kódy QRCode, které jsou speciálně navrženy pro kódování japonských znaků Kanji. Ve světě je pak velmi rozšířen standard Datamatrix, který má vysokou hustotu záznamu na jednotku plochy. Kódy Datamatrix se využívají převážně v průmyslu při sledování strojních součástí. Zajímavostí je, že kódy používané v průmyslu jsou na ocelových částech embosované, vytlačené. V těchto situacích dvojrozměrné kódy poskytují levnější alternativu k technologii RFID (radiofrekvenční identifikace). Kódy Datamatrix i QRcode obsahuje kódování opravy chyby Reed-Solomon nabízející opravy kódu, jež je až ze 60 procent poškozen. Nevýhodou standardizovaných kódů Datamatrix či Qrcode je skutečnost, že nebyly vyvinuty speciálně pro kamery mobilních telefonů. Nejmenší datová jednotka se nazývá modul. Modul má u standardu Datamatrix i Qrcode velikost blížící se krajním možностям rozpoznání za pomoci obyčejného mobilního telefonu. V poslední době se začínají objevovat grafické kódy speciálně vyvinuté pro kamery mobilních telefonů. Tyto kódy mají rozličné tvary. Existují kruhové grafické kódy, Shotcode, RevoKód i kódy využívající hexagonální mřížku, Beetag.

Grafické kódy mohou ukládat jakékoliv data. Zde se nabízejí dvě možnosti zakódování informace. V jednom případě je v kódu uložena internetová adresa a v druhém případě je v kódu uloženo unikátní identifikační číslo. V případě unikátního identifikátoru je potřeba zaslat dotaz na server, který odpoví internetovou adresou přiřazenou k unikátnímu číslu.

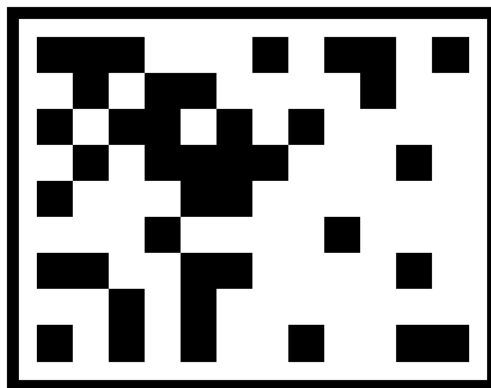


Obrázek 2 – Kruhový grafický kód Shotcode

Kruhovými kódy se zabývá firma Shotcode. Tento systém grafických kódů má výhodu v tom, že je speciálně vyvinut pro dekódování na mobilních telefonech. Výhodou systému Shotcode je skutečnost, že nezáleží na úhlu natočení a je tedy možno použít jeden obecný algoritmus a natočení kódu podle středové osy již dále neřešit.

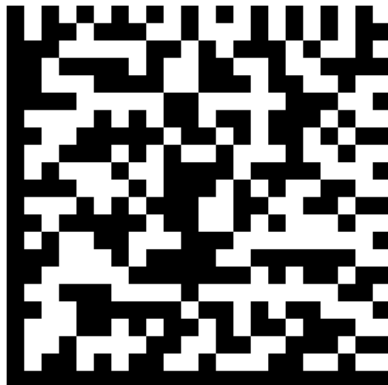
Dalším systémem grafických kódů jsou hexagonální kódy Beetag. Tyto kódy mohou být nalezeny na adrese www.beetag.com. Zajímavostí Beetag kódů je možnost vkládat další informace jako text a obrázky přímo do předem připraveného pole v kódu.

Na fakultě FIT na VUT v Brně jsou vyvíjeny dvojrozměrné čárové kódy, které je možno dekódovat za pomoci obyčejného mobilního telefonu s nainstalovaným čtecím software. Tyto kódy je možno nalézt na adrese <http://www.pepik.net/gkody/>. Zajímavostí proprietárních kódů z FIT je možnost uložení nejen unikátního identifikátoru, ale i IP adresy serveru, na který se čtecí zařízení může připojit po správném dekódování kódu.



Obrázek 3 – Grafický kód z FITu

Průmyslové grafické kódy Datamatrix je možno přečíst za pomoci dekodéru několika výrobců. Na internetu také existují generátory pro kódy Datamatrix a QRCode. Jeden z generátorů je možno nalézt na internetové adrese <http://datamatrix.kaywa.com>.



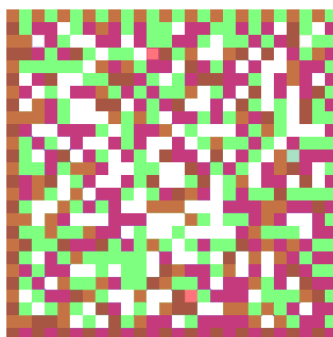
Obrázek 4 – Grafický kód standardu Datamatrix

V Japonsku jsou velice populární grafické kódy standardu Qrcode. Tyto kódy byly vyvinuty firmou Denso, která je má patentovány. Firma však specifikaci těchto kódů uvolnila pro veřejnost. Generátor kódů QRCode je možno nalézt na internetové adrese <http://qrcode.kaywa.com>.



Obrázek 5 – Grafický kód standardu Qrcode

Grafické kódy je možno i proložit mezi sebe a získat tak grafický formát se zvětšenou kapacitou. Na obrázku je možno vidět kombinace tří Datamatrix kódů, které jsou barevně odlišeny a otočeny o devadesát stupňů. Zkombinovány jsou kódy v modré, zelené a červené barvě.



Obrázek 6 – Barevný kód Datamatrix

1.2 Webové technologie

1.2.1 SOAP

Webové služby spolu mohou komunikovat pomocí protokolu SOAP [6] (protokol jednoduchého přístupu k objektům). Protokol nabízí unifikované rozhraní založené na XML, které umožňuje jednotnou a snadnou komunikaci. Každá zpráva SOAP se nachází v tzv. obálce, která volá vzdálené metody serveru. Jak si můžeme všimnout, dotaz ve formátu SOAP definuje adresu serveru, který nám na požadavek odpoví. V požadavku je přenášén název metody (getProductDetails) a název parametru (productID). Ukázka zprávy SOAP je níže [7]. Klient se dotazuje webové služby na informace, které se vztahují k produktu s identifikačním číslem 827635.

```
<SOAP:Envelope xmlns:SOAP="http://schemas.XMLSOAP.org/SOAP/envelope/">
  <SOAP:Body>
    <getProductDetails xmlns="http://warehouse.example.com/ws">
      <productID>827635</productID>
    </getProductDetails>
  </SOAP:Body>
</SOAP:Envelope>
```

Server odpoví zpět a klientské zařízení získá strukturované informace o požadovaném produktu. Informace jsou ve formátu SOAP a zašle detailní informace o požadovaném produktu. V odpovědi získáme adresu serveru, který s klientem komunikoval. Informace o samotném produktu jsou získána ve strukturovaném formátu, který zjednodušuje jejich zpracování.


```
<SOAP:Envelope xmlns:SOAP="http://schemas.XMLSOAP.org/SOAP/envelope/">
  <SOAP:Body>
    <getProductDetailsResponse xmlns="http://warehouse.example.com/ws">
      <getProductDetailsResult>
        <productName>Toptimate 3-Piece Set</productName>
        <productID>827635</productID>
        <description>3-Piece luggage set. Black Polyester.</description>
        <price>96.50</price>
        <inStock>true</inStock>
      </getProductDetailsResult>
    </getProductDetailsResponse>
  </SOAP:Body>
</SOAP:Envelope>
```

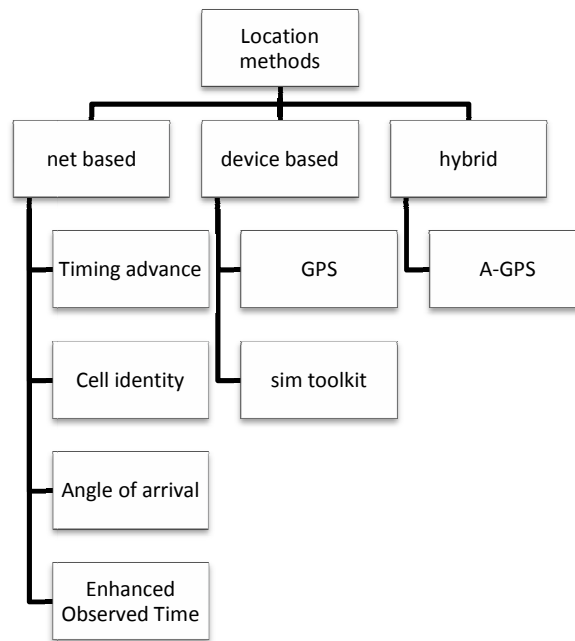
1.2.2 RSS

Formát RSS (syndikace obsahu) je stejně jako SOAP založen na XML [5] a můžeme se s ním setkat na webu. Soubor ve formátu RSS je dokument, který v sobě obsahuje sémanticky definované položky. Formát je konstruován za účelem snadné čitelnosti pro automatizované systémy. V následující ukázce je zachycen příklad souboru RSS, který poskytuje informace z webového článku. RSS definuje základní položky, které musí být povinně obsaženy. Jedná se o položky channel, item, title a link. Položky mají jasný význam a informace v nich obsažené mohou být sémanticky zařazeny. Níže je ukázka, kterou jsem získal ze zpravodajského serveru NASA. Ve zprávě je definován jeden channel a item. Celá zpráva informuje o nutnosti vyvinout nové motory před tím, než se člověk vydá na Mars.

```
<?XML version="1.0"?>
<RSS version="2.0">
  <channel>
    <title>Liftoff News</title>
    <link>http://liftoff.msfc.nasa.gov/</link>
    <description>Liftoff to Space Exploration.</description>
    <item>
      <title>The Engine That Does More</title>
      <link>http://liftoff.msfc.nasa.gov/news/2003/news-VASIMR.asp</link>
      <description>Before man travels to Mars, NASA design new engines
        </description>
    </item>
  </channel>
</RSS>
```

1.3 Lokalizace mobilních zařízení

K lokalizaci mobilního zařízení je možné využít metody, které využívají vlastností mobilních sítí nebo metody, kde mobilní zařízení měří polohu samo. Každý z těchto přístupů má své výhody i omezení. Na obrázku níže je možno spatřit stromovou strukturu, která vyobrazuje rozdělení metod pro zjištění polohy mobilního zařízení. Je vidět, že u obou metod máme k dispozici několik technologií, které jsou na daném přístupu založeny. Zcela vpravo je pak třetí metoda, která kombinuje výhody jak systémů zjištění polohy závislé na síti, tak i systémů zjištění polohy přímo v mobilním zařízení.



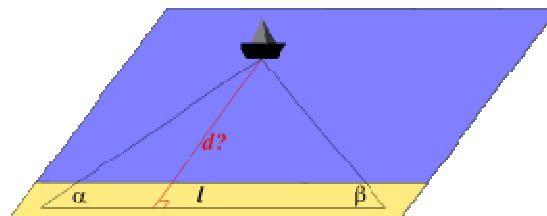
Obrázek 7 - Rozdělení metod lokalizace mobilních zařízení

1.3.1 Obecné principy

Systemy, které umožňují lokalizovat mobilní zařízení využívají ke zjištění polohy několik základních metod. První metoda je triangulace, druhá trilaterace a třetí se nazývá multilaterace.

1.3.1.1 Triangulace

V elementární geometrii je triangulace způsob zjišťování souřadnic a vzdáleností. Provádí se trigonometrickým výpočtem. Sestrojí se pomyslný trojúhelník, jehož jedna strana je strana již známého jiného trojúhelníku s dvěma koncovými referenčními body a třetím bodem je místo, jehož souřadnice se zjišťuje.



Obrázek 8 - Vysvětlení principu triangulace

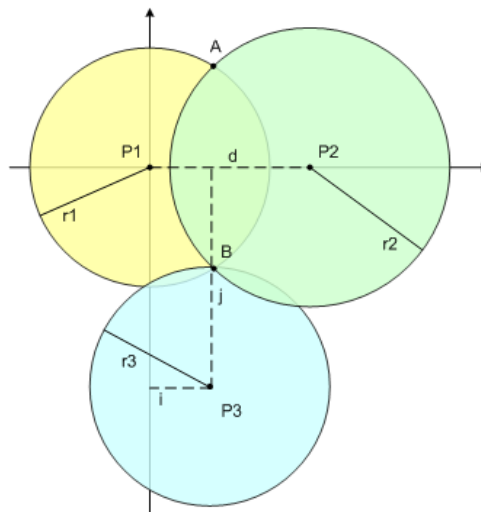
Jako příklad si můžeme vzít výpočet polohy a vzdálenosti lodi od pobřeží [8]. V místě A je změřen úhel α mezi pobřežím a lodí. V místě B je změřen úhel β .

Jestliže máme k dispozici souřadnice bodů A a B, nebo známe vzdálenost l , můžeme vypočítat souřadnice C a vzdálenost d .

Obrázek ukazuje postup výpočtu vzdálenosti lodi d od břehu. V bodě A změříme úhel α , v bodě B změříme úhel β . Předpokladem je známá vzdálenost l . Podle pravidla součtu vnitřních úhlů vypočítáme úhel θ u lodi: $\theta = 180^\circ - \alpha - \beta$. K zjištění délek zbývajících stran trojúhelníků dále využijeme sinovou větu: $\sin(\alpha) / a = \sin(\beta) / b = \sin(\theta) / l$. Nakonec můžeme podle definice goniometrické funkce sinus vypočítat vzdálenost d : $\sin(\alpha) = d / a$ nebo $\sin(\beta) = d / b$.

1.3.1.2 Trilaterace

Trilaterace je metoda zjištění relativní polohy objektů využívající geometrii trojúhelníků v podobném duchu jako triangulace. Na rozdíl od triangulace, která používá měření úhlů ke zjištění polohy, trilaterace využívá známé polohy referenčních bodů a vzdálenost mezi referenčními body a subjektem, jehož polohu měříme. K tomu, abychom přesně a jednoznačně určili polohu bodu v 2D prostoru, potřebujeme mít k dispozici minimálně 3 referenční body. U 3D prostoru pak potřebujeme ke změření 4 reference. Řešení pro určení polohy bodu ve 2D prostoru může být nalezeno jako průnik 3 kružnic. K tomu však musíme mít splněny některé podmínky. Jeden referenční bod musí být umístěn v počátku souřadnicového systému. Druhý referenční bod musí být umístěn na ose x . Všechny referenční body musí být na ose z v 0. Tyto podmínky jsou vždy splněny, protože můžeme souřadný systém transformovat do námi požadované polohy. Nyní si uvedeme jednoduchý příklad výpočtu trilaterace pro soustavu tří bodů.



Obrázek 9 - Trilaterace ve 2D prostoru

Začínáme se třemi koulemi, jejichž středy jsou body P1, P2, P3 a poloměry r1, r2, r3. Hledáme polohu bodu B, který přijímá signál z okolních vysílačů.

$$\begin{aligned}r_1^2 &= x^2 + y^2 + z^2 \\r_2^2 &= (x - d)^2 + y^2 + z^2 \\r_3^2 &= (x - i)^2 + (y - j)^2 + z^2\end{aligned}$$

Odečteme první kouli od druhé a získáme x

$$x = \frac{r_1^2 - r_2^2 + d^2}{2d}$$

Substitucí x zpět do původní první rovnice získáme

$$y^2 + z^2 = r_1^2 - \frac{(r_1^2 - r_2^2 + d^2)^2}{4d^2}$$

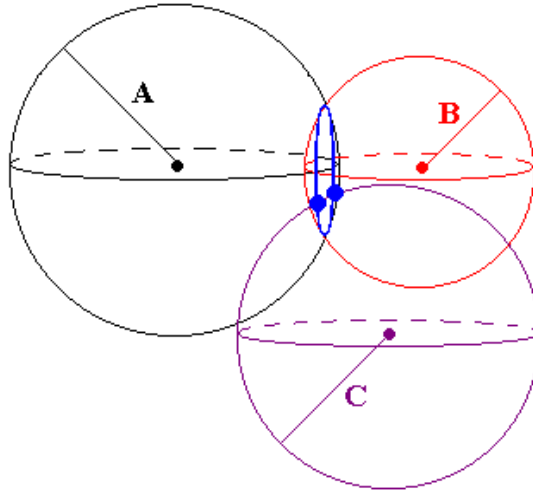
Novou rovnici položíme rovnu třetí rovnici koule a získáváme výpočet polohy na souřadnici y.

$$y = \frac{r_1^2 - r_3^2(x - i)^2}{2j} + \frac{j}{2} - \frac{(r_1^2 - r_2^2 + d^2)^2}{8d^2j}$$

Nyní máme k dispozici x a y. Můžeme přistoupit k výpočtu souřadnice z

$$z = \sqrt{r_1^2 - x^2 - y^2}$$

Je vidět, že výpočet z má 0, 1 nebo 2 řešení. To lze zobrazit tak, že dvě koule A a B vytváří svým průnikem kruh. Koule C se pak nemusí kruhu vůbec dotknout. Tím pádem nemá naše rovnice řešení. Pokud se C dotkne kruhu, má rovnice jedno řešení. A konečně, pokud stejně jako na obrázku C kruh protne na dvou místech, máme více řešení. V případě kdy máme dvě řešení, musí nastoupit metoda, která nám umožní rozlišit správné řešení. To může být například zajištěno průnikem s další koulí v prostoru.



Obrázek 10 - Trilaterace ve 3D prostoru může mít více řešení

1.3.1.3 Multilaterace

Známa jako hyperbolické polohování. Je to proces vypočtu polohy za pomoci rozdílu časů, kdy byl signál vyslán z bodu A a dorazil do bodů B, C, D, Tento princip se nazývá Time difference of arrival (TDOA). Existují dva základní principy. První z nich je případ, kdy měřený objekt vyšle signál ke třem či více přijímačům a ty změří rozdíl časů mezi vysláním signálu a jeho příjmem. Naopak může měřený objekt přijímat signál od více přijímačů a rozdíl časů si počítat u sebe. Jestliže je signál vyslán do prostoru, k přijímačům na různých geografických pozicích, dorazí signál v rozdílných časech. Jestliže víme přesnou polohu dvou přijímačů a TDOA, jako výsledek získáme hyperboloid, ve kterém se zjišťovaná poloha nachází [9]. Jestliže máme další přijímač, můžeme vypočítat další TDOA a získat další hyperboloid.

Průnikem obou vypočtených geometrických útvarů získáme křivku, která popisuje zjišťovanou polohu. Čím více máme přijímačů, které zjišťují TDOA, tím lépe můžeme zjistit polohu vysílače signálu. N přijímačů vytváří N-1 hyperboloidů. Jestliže je $N \geq 4$ přijímačů, měli bychom tedy dostat velmi přesnou polohu. Avšak v praxi dochází k mnoha chybám a občas hyperboloidy nemusí vytvořit průnik. Také můžeme přistoupit k většímu počtu měření. Jestliže získá každý přijímač více TDOA, tato hodnota se může průměrovat a získat větší přesnost určení polohy.

Výpočet multiaterace

Uvažujme vysílač na neznámé pozici $V(x, y, z)$ v prostoru. Naším cílem je zjistit polohu vysílače. Dále uvažujme o systému se 4 přijímači C, L, Q, R.

Získáme časy potřebné k přenosu signálu od vysílače k přijímači jsou

$$T_L = \frac{1}{c}(\sqrt{(x - x_L)^2 + (y - y_L)^2 + (z - z_L)^2})$$

$$T_R = \frac{1}{c}(\sqrt{(x - x_R)^2 + (y - y_R)^2 + (z - z_R)^2})$$

$$T_Q = \frac{1}{c}(\sqrt{(x - x_Q)^2 + (y - y_Q)^2 + (z - z_Q)^2})$$

$$T_C = \frac{1}{c}(\sqrt{(x - x_C)^2 + (y - y_C)^2 + (z - z_C)^2})$$

Časový rozdíl TDOA může být spočítán jako

$$\tau_L = T_L - T_C = \frac{1}{c}(\sqrt{(x - x_L)^2 + (y - y_L)^2 + (z - z_L)^2} - \sqrt{x^2 + y^2 + z^2})$$

$$\tau_R = T_R - T_C = \frac{1}{c}(\sqrt{(x - x_R)^2 + (y - y_R)^2 + (z - z_R)^2} - \sqrt{x^2 + y^2 + z^2})$$

$$\tau_Q = T_Q - T_C = \frac{1}{c}(\sqrt{(x - x_Q)^2 + (y - y_Q)^2 + (z - z_Q)^2} - \sqrt{x^2 + y^2 + z^2})$$

Kde (x_L, y_L, z_L) je například poloha L přijímače. Poloha dalších přijímačů je analogická.

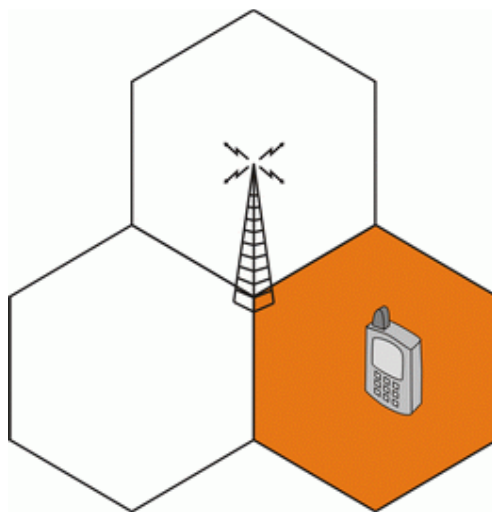
Konstanta C je rychlost světla.

1.3.2 Lokalizace na úrovni mobilní sítě

Mobilní a bezdrátové sítě nabízejí možnost zjistit polohu zařízení, které se v nich nachází. Zjištění polohy terminálu je možné uskutečnit několika způsoby. V dalším textu jsou popsány základní metody, které jsou používány v mobilních sítích.

1.3.2.1 Lokalizace pomocí BTS

Tato metoda je prvním a nejjednodušším řešením. Mobilní terminál se lokalizuje podle identifikačního čísla základnové stanice. Tento systém má tu výhodu, že nevyžaduje od provozovatele mobilní sítě náklady spojené se zavedením. Problém tohoto nejjednoduššího způsobu je, že umožňuje určit pouze okruh, kde se mobilní zařízení nachází. U vícesektorových BTS (základnová stanice) můžeme zjistit výseč kruhu. Přesnost této metody je závislá na hustotě stanic BTS. Nejlepší přesnosti se dosahuje u nízkovýkonných BTS, vykrývajících např. supermarkety, stanice metra, stadiony apod., i když i zde může být výjimečně přesah dosahu v řádu desítek až stovek metrů mimo daný objekt [10].

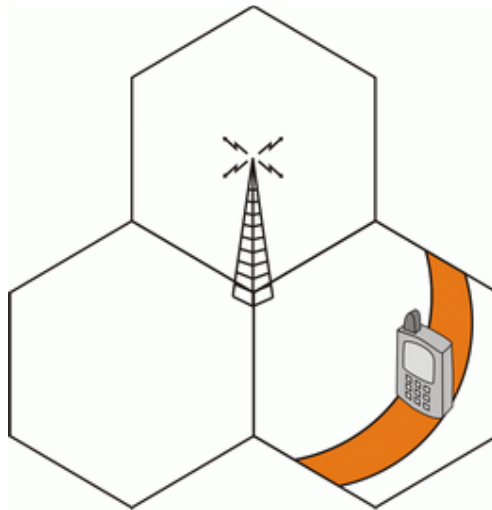


Obrázek 11 - Lokalizace mobilu dle sektoru BTS

http://access.feld.cvut.cz/image/200602281326_fig2.gif

1.3.2.2 Timing advance

Metoda Timing advance uvažuje dobu, jakou signál cestuje mezi základnovou stanicí a mobilním terminálem. Tímto způsobem můžeme zpřesnit určení polohy. Při výpočtu polohy se používá triangulace, která byla popsána již dříve.

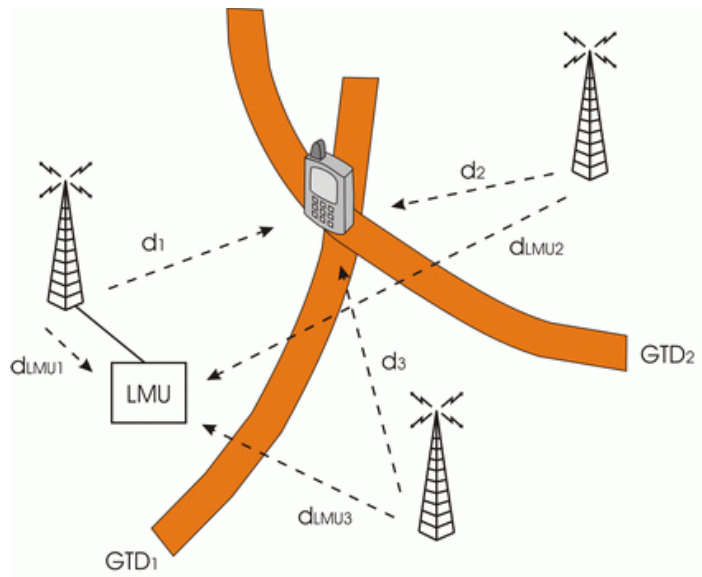


Obrázek 12 - Lokalizace s výpočtem časové prodlevy mezi vysíláním a příjmem signálu

http://access.feld.cvut.cz/image/200602281326_fig2.gif

1.3.2.3 Enhanced Observed Time

Tato metoda je založena na principu TDOA (rozdíl časů příchodu signálů) a k výpočtu se využívá multilaterace. Problémem metody EOT (vylepšené měření časového signálu) je nutnost časové synchronizace základnových stanic. Jestliže chce mobilní operátor využít tuto metodu, musí být jeho síť vybavena zařízením LMU (jednotka pro měření polohy). LMU provádí měření a výpočet reálných časových rozdílů synchronizace stanic BTS [11]. Ideálně je tato hodnota rovna nule. Na základě této informace se provedou korekce a pak se vypočítá poloha mobilního terminálu.

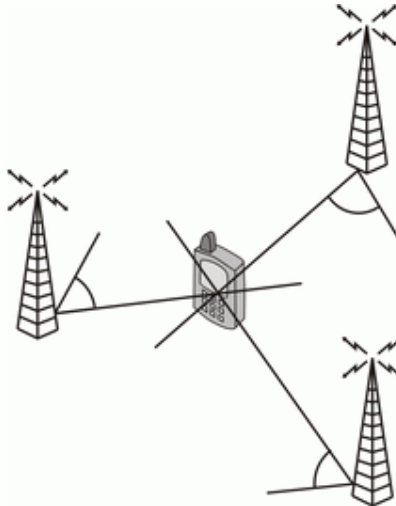


Obrázek 13 - LMU synchronizuje BTS

http://access.feld.cvut.cz/image/200602281326_fig3.gif

1.3.2.4 Angle of Arrival

Při této metodě se používá směrových antén. Je nutno znát vyzařovací charakteristiky antén. Měření úhlu, pod kterým je signál přijímán, se provádí v základnové stanici nebo v mobilním zařízení. Výsledkem měření jsou přímky, jejichž průnik určuje polohu mobilního terminálu.



Obrázek 14 - Lokalizace dle úhlu dopadu signálu

http://access.feld.cvut.cz/image/200602281326_fig4.gif

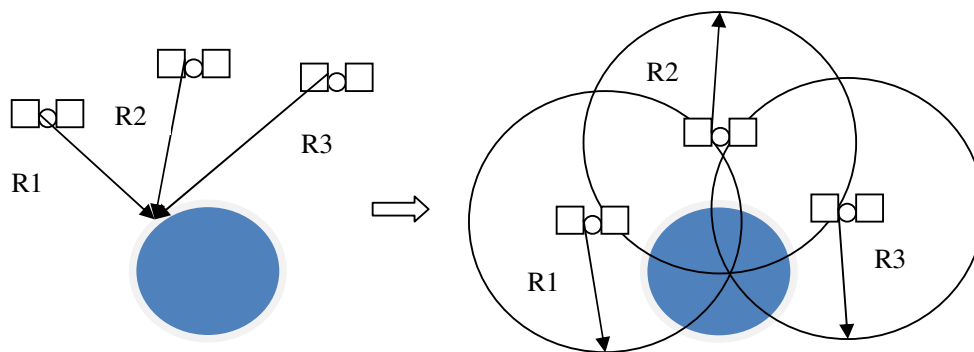
1.3.2.5 Enhanced Cell Global Identity

Lokalizační metody Cell ID a Timing advance jsou v případě metody Enhanced Cell Global Identity (E-CGI) vylepšeny o měření úrovně signálů. E-CGI používá pro výpočet vzdálenosti mezi mobilním zařízením a základnovou stanicí model šíření signálů. Podle naměřených úrovní signálů v místě mobilního zařízení a znalosti vysílacích výkonů základnových stanic jsou predikovány oblasti s nejpravděpodobnějším výskytem uživatele. Jeho poloha je poté obvykle stanovena jako těžiště této oblasti. Přesnost metody E-CGI je kolem 50-550m pro městské oblasti a 250m - 8km pro venkovské oblasti [11].

1.3.3 Satelitní lokalizace

1.3.3.1 Systém GPS

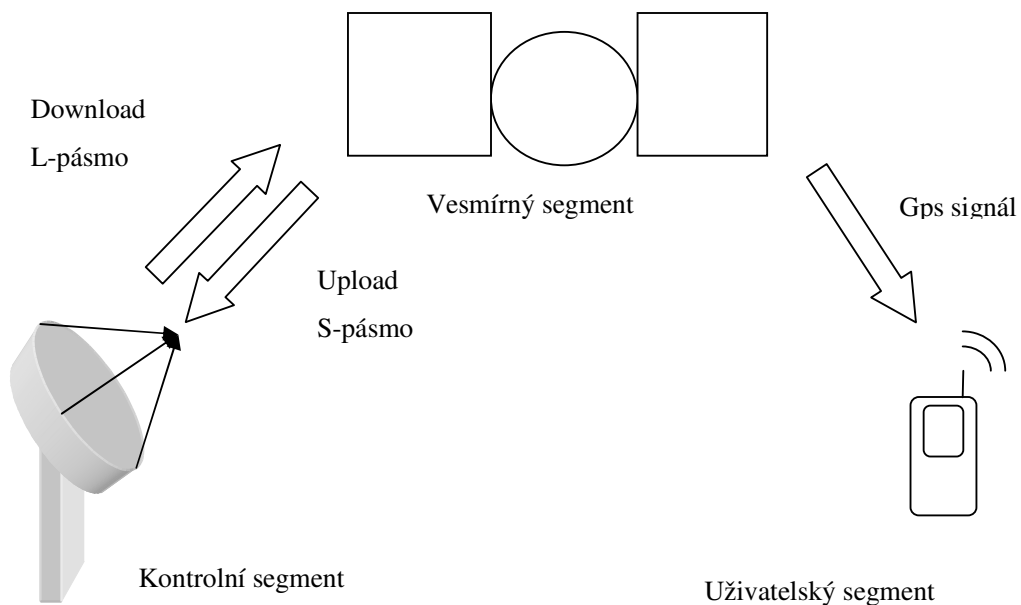
GPS je lokalizační systém provozovaný americkou armádou. Tento systém byl původně vytvořen pro armádu pod názvem NAVSTAR GPS (Navigation Signal Timing and Ranging Global Positioning System). V roce 1980 byl uvolněn i pro civilní účely. Systém se skládá ze 24 satelitů neustále obíhajících okolo Země ve výšce 36 000 km. Satelity jsou vybaveny atomovými hodinami, které dosahují přesnosti jedné miliardtiny vteřiny. Oběžnice neustále vysílají signál s informacemi o své poloze a času odeslání. Pozemní přijímače jsou schopny tyto signály přijímat a vypočítat tak svou polohu pomocí triangulace [12].



Obrázek 15 - Signály z GPS družic vytvářejí v prostoru pomyslné koule

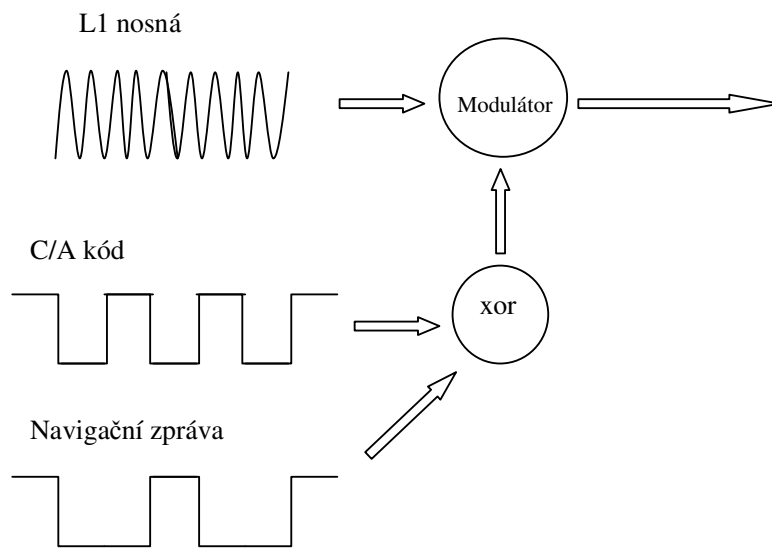
Technický popis

Systém GPS se skládá ze tří segmentů. Jedná se o kontrolní segment, vesmírný segment a uživatelský segment. Kontrolní segment se stará o chod systému. Vesmírný segment je tvořen družicemi a uživatelský segment využívá informace o poloze



Obrázek 16 - Segmenty systému GPS

GPS satelity vysílají dva nosné signály na frekvencích 1,575.42 MHz pro nosnou frekvenci L1 a 1,227.60 MHz pro nosnou frekvenci L2 [12]. L1 a L2 jsou modulovány digitálními kódy a navigační zprávou.



Obrázek 17 - Zakódování GPS signálu

Satelity vysílají dva typy informací o čase – digitální kódy. Coarse acquisition code C/A je běžně používán pro civilní účely. C/A se skládá z pseudonáhodného kódu o velikosti 1023 bitů, který je vyslán na frekvenci 1.023Mhz každou milisekundu. Každý satelit má svůj C/A kód, který je umožňuje rozlišit mezi sebou. P-Code (precisní kód) je podobný signál jako C/A. Korelovaná data jsou rozdělena do navigační zprávy o velikosti 37500 bitů. Zpráva je odesílána po dobu 12.5 minuty rychlostí 50 bitů za vteřinu. Pro výpočet polohy musí přijímač vědět přesný čas. Časová informace je generována krystalem podobným tomu, jaký máme v digitálních hodinkách. Čas je periodicky upravován podle referenčních časů ze satelitních signálů L1. Jakmile přijímač identifikuje satelity podle rozdílných C/A signálů, uloží si do paměti data. Poté vygeneruje svůj lokální C/A signál a porovná jej se signálem přijatým ze satelitu. Rozdíl mezi lokálním a přijatým signálem se spočítá zpožděním lokálního signálu. Při určitém zpoždění budou oba signály identické a my pak můžeme jednoduše vypočítat naši vzdálenost od satelitu.

1.3.3.2 Formát NMEA

Informace o poloze jsou z GPS modulu získána ve standardizovaném formátu NMEA[GPSintro] (mezinárodní námořní elektronická asociace). NMEA je specifikace dat pro komunikaci mezi námořními elektronickými zařízeními a obecněji GPS senzory. Tento protokol je definován organizací National Marine Electronics Association. Formát používá jednoduchý sériový protokol, který přenáší data od jednoho vysílače k jednomu či více přijímačům. Tento standard definuje strukturu každé věty (zprávy), takže každý přijímač může zprávu správně dekodovat.

- Každá zpráva začíná znakem \$
- Dalších 5 znaků definuje typ zprávy
- Všechny datové položky jsou odděleny čárkou
- První znak, který následuje po ukončení zprávy je hvězdička *
- Hvězdička je ihned následována dvojciferným kontrolním součtem

Ukázka formátu NMEA

\$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47

Vysvětlení

GGA	Global Positioning System Fix Data
123519	Zafixováno v 12:35:19 mezinárodního času
4807.038,N	Šířka 48 stupňů 07.038' Severně
01131.000,E	Délka 11 stupňů 31.000' Východně
1	Kvalita zafixování
08	Počet připojených satelitů
0.9	Horizontální odchylka
545.4,M	Výška v metrech nad mořem
46.9,M	Výška geoidu nad elipsoidem WGS84 (světový geodetický systém)

2 Aplikace pro distribuci místně závislých dat

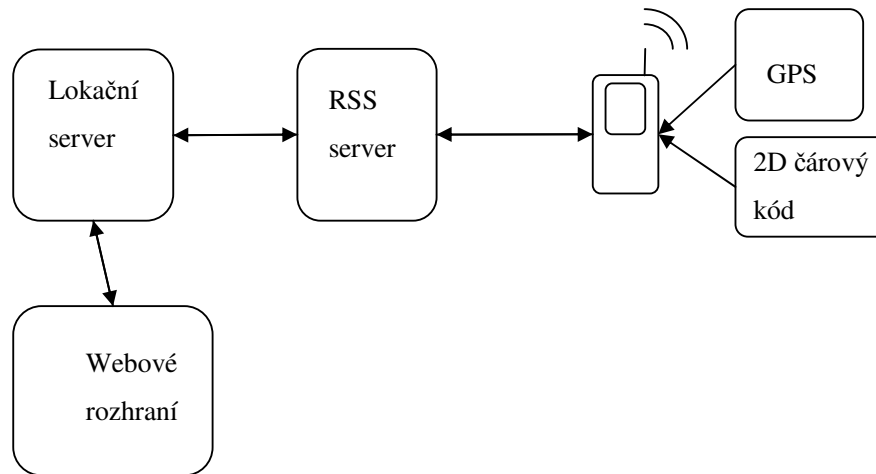
V této kapitole bude popsán návrh aplikace, která umožňuje distribuci místně závislých informací. K řešení našeho problému využijeme všechny technologie, které byly popsány v dřívější kapitole. Nyní se tedy pusťme do návrhu samotného systému.

2.1 Specifikace požadavků

Požadavkem na systém je schopnost distribuce multimediálních informací závislých na poloze. Je nutné, aby systém komunikoval skrz mobilní datovou síť se vzdáleným serverem. Dodatečným požadavkem je potom schopnost aplikace zpracovávat obrazová data získaná z fotoaparátu mobilního telefonu. Tato obrazová data budou rozkódována a z nich získána adresa serveru, na který se mobilní zařízení připojí. V konečném důsledku není aplikace omezena na pouze na připojení k předem danému serveru. Mobilní klient zjistí svou polohu pomocí některé z dříve popsaných metod. Zjištěná informace o poloze je předána na vzdálený server pro zpracování dat a zjištění požadované polohy. Je potřeba, aby byla aplikace kompatibilní s maximálním počtem mobilních zařízení. Informace o poloze je tedy získána univerzální metodou, která je nezávislá na mobilním operátorovi. Lokačně závislá data jsou přenášena zpět do terminálu ve standardizovaném formátu RSS, protože jej lze jednoduše číst. Pro zobrazení multimédií použijeme platformu Flash.

2.2 Návrh aplikace

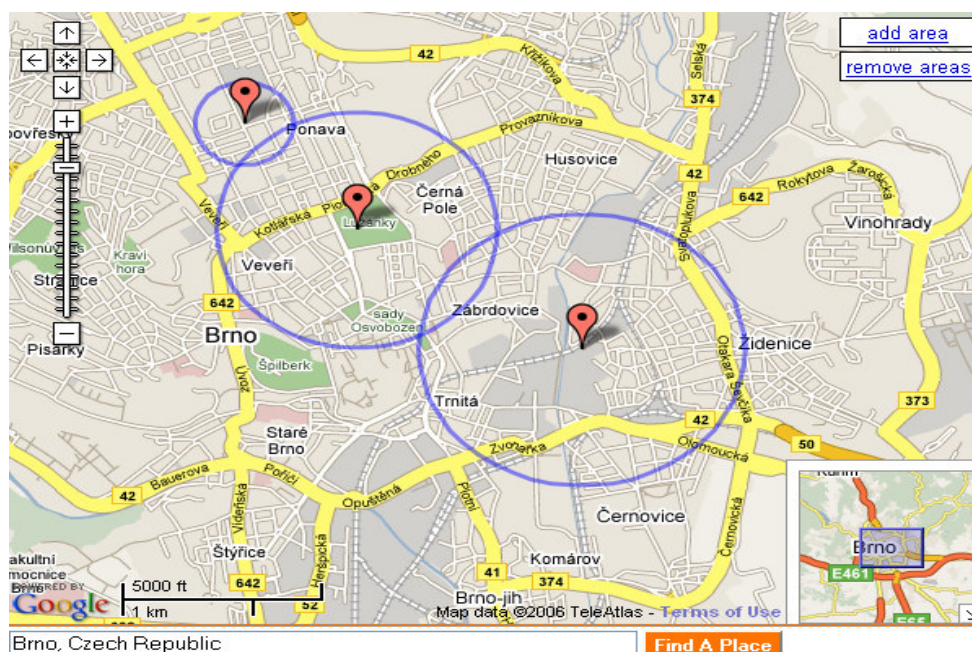
System se skládá z několika komponent, které je možné si prohlédnout na následujícím obrázku. V dalších podkapitolách práce je možno se blíže seznámit s komponentami mobilního zařízení, RSS serveru a Webového rozhraní. Lokační server není součástí mé diplomové práce a jeho bližší popis lze nalézt v práci mého kolegy Jiřího Musila [16]. Komponenty webového rozhraní a server RSS ve svém návrhu počítají s předem daným komunikačním rozhraním.



Obrázek 18 - Architektura aplikace

2.2.1 Webové rozhraní

Webové rozhraní je určeno pro vkládání informací k poloze. S jeho pomocí je možno interaktivně přiřadit k místu určitá data. Tato data pak budou v konečném důsledku zobrazována v mobilním zařízení ve formě souborů RSS. Klient je naprogramován za pomoci technologie JavaScript, protože na této technologii je postaveno mapové webové rozhraní od firmy Google. To nám umožní jednoduché a intuitivní zadání dat a jejich vztahů k místům.



Obrázek 19 - Ukázka z modulu pro vkládání a polohy dat a její definici
<http://maps.google.com>

Webový klient využívá mapové rozhraní firmy Google. Základním požadavkem je jednoduché ovládání mapy a možnost definice místa na mapě, kde se budou vyskytovat lokační data. K ovládání mapy slouží prvky pro změnu měřítka a pro pohyb po mapě. Ke správě polohy a oblastí pak slouží tlačítka pro přidávání a ubírání polohy. Každá poloha má definován poloměr, který určuje oblast její působnosti. Poloměry jsou vykresleny modrou kružnicí. Střed oblasti definuje značka, kterou je možno přesunout. Jestliže potřebuje uživatel změnit polohu značky a tím pádem i oblasti, může tak učinit pouhým přesunutím ikonky markeru stylem drag and drop.

2.2.1.1 Načítání externích dat

Webový klient má k dispozici data ze serveru, která definují počáteční rozmístění lokačních informací a poloměr jejich působnosti.

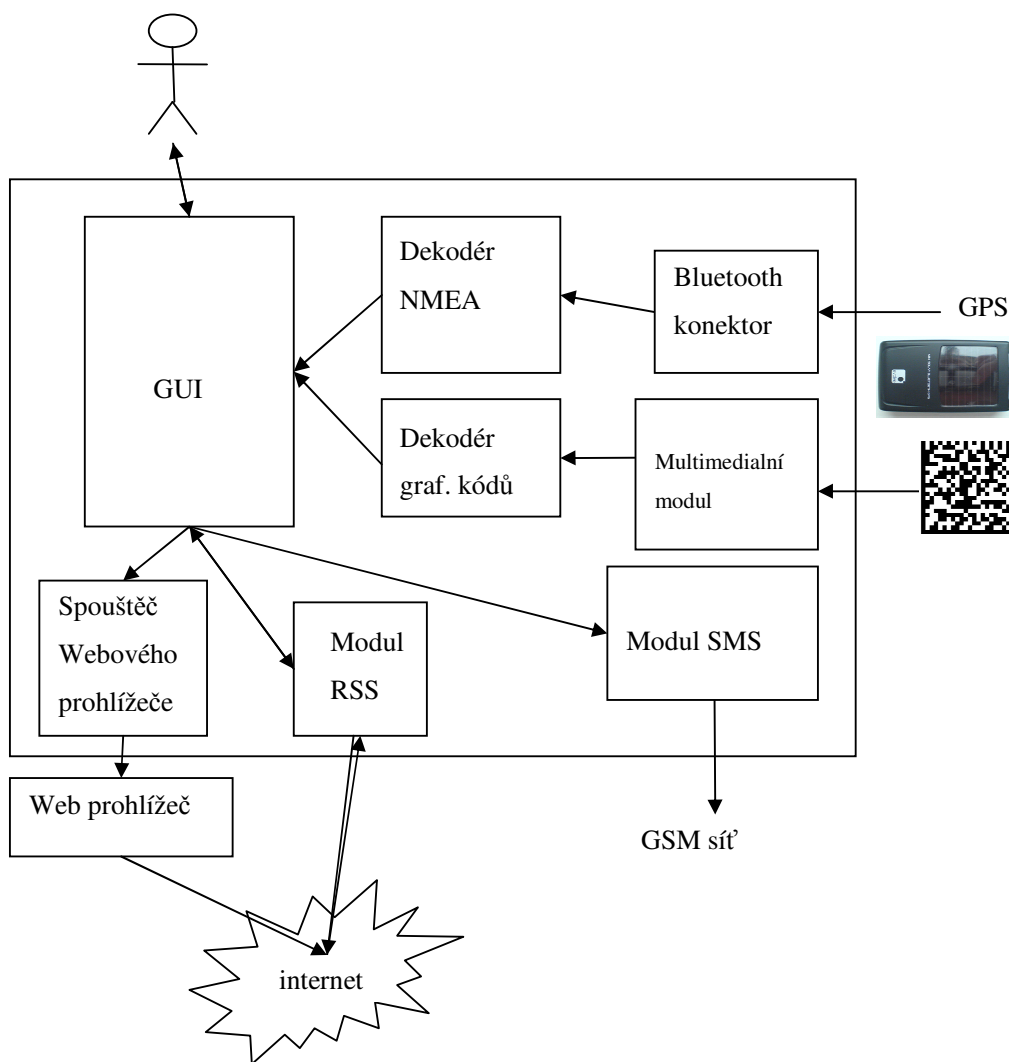
Vstupní data jsou uložena ve formátu XML a jsou definována následujícím způsobem

```
<adplaze_areas_list>
<error/>
<strings>
  <str id="add_area">Add area</str>
  <str id="remove_area">Remove area</str>
</strings>
<info id="1" html="" help="Testovací data" maximize="Maximalizováno">
  <areas>
    <area id="1" lat="16.359741" lon="49.11" round="10" label="data 1"/>
    <area id="2" lat="6.1741" lon="40.11" round="10" label="data 2"/>
  </areas>
</info>
</adplaze_areas_list>
```

Oddíl “strings” definuje text, který je zobrazen u tlačítek pro přidávání a odebrání oblastí. Je nutné mít definován text pro dvě tlačítka – přidání a ubrání oblasti. Oddíl “info” pak definuje vlastní informaci. Každá lokačně závislá informace může být obsažena ve více oblastech “areas”. Každá oblast “area” má přiřazeno unikátní číslo “id”, poloměr “round” a text “label”. Oblasti navíc obsahují informaci o své poloze, která je uložena jako zemská šířka a délka.

2.2.2 Klient mobilního zařízení

Klient bude pracovat v prostředí mobilního telefonu a bude naprogramován za pomoci jazyka Java. Úkolem klienta je získat informace ze senzoru GPS a dále je zpracovat. Informace o poloze mobilního telefonu bude poslána na vzdálený server, jehož adresu jsme získali vyfotografováním a rozkódováním dvojrozměrného kódu. V dalším textu provedeme rozdělení klientské aplikace na samostatné moduly a navrhne základní logiku aplikace.



Obrázek 20 - Modulární rozdělení klientské části

2.2.2.1 Moduly klientské části

Modul GUI

Tento modul má za úkol zobrazovat grafickou část mobilní aplikace. GUI (Grafické uživatelské rozhraní) je potřeba vytvořit nezávislé na mobilním telefonu. Z důvodu přenositelnosti jsou použity pouze základní formuláře, seznamy, atd.

Lokační modul

Za předpokladu, že je v mobilním zařízení přítomna rozšiřující knihovna JSR179, je možné získat lokační data i z jiných zdrojů než pouze z GPS. Konkrétně se jedná o použití mobilní sítě. Mobilní telefon může určit svou polohu pomocí blízkých majáků, tzv. short range beacons. Jestliže však lokační knihovna v mobilním telefonu není implementována,

můžeme na J2ME platformě získat lokační informace přímo ze senzoru GPS. K tomu musíme implementovat několik dalších modulů. Úkolem těchto modulů je nahradit funkce, které knihovna JSR179 (knihovna pro vývoj lokačních aplikací) implicitně implementuje. K dispozici máme několik možností, jak se připojit k přijímači GPS a získat data o poloze. Lze použít datový kabel, infračervený přenos, nebo Bluetooth. Pro své vlastnosti je Bluetooth nejlepší variantou. Je potřeba vytvořit modul se schopností vyhledat všechny zařízení nacházející se v blízkosti mobilního telefonu a připojit se na vybrané zařízení pomocí Bluetooth.

Bluetooth konektor

Jak již bylo řečeno dříve, je potřeba se připojit k GPS pomocí bezdrátové technologie Bluetooth. Bude nutné vyhledat dostupná zařízení a připojit se k senzoru GPS. Modul pro získání lokační informace má tedy za úkol vyhledání Bluetooth zařízení, připojení se k senzoru GPS a kontinuální čtení dat z senzoru GPS. Modul zajistí připojení mobilního zařízení k vybranému Bluetooth zařízení. Data, která můžeme z GPS číst jsou ve formátu NMEA. Je nutné počítat s různými okolnostmi, které mohou nastat při Bluetooth komunikaci. Bluetooth zařízení může přestat komunikovat z různých důvodů. V tomto případě pak musíme zajistit správné ukončení spojení a adekvátně reagovat na ztrátu lokačních dat.

Dekodér NMEA

Dekodér má za úkol z dat standardu NMEA zjistit potřebné údaje. Konkrétně nás budou zajímat atributy Latitude a Longitude. Data budou předávána modulu pro http komunikaci, který pak provede dotaz na Lokační RSS server.

Modul pro http komunikaci

Modul propojuje mobilní zařízení se vzdáleným RSS serverem. Komunikace probíhá na http komunikační vrstvě. Jakmile je zjištěna poloha mobilního telefonu, může být zaslán požadavek na vzdálený server. Zde je nutné počítat s možností nedostupného internetového připojení. V daném případě je nutné vhodně reagovat.

Modul čtení dat RSS

Tento modul je fyzicky umístěn v mobilním zařízení. Jeho funkcí je rozdělit příchozí data na bloky, které lze zobrazit na displeji mobilu. Vstupem bloku je soubor formátu RSS a výstupem kolekce typů, které typově odpovídají přenášeným informacím. Modul implementuje zpracování příchozích dat XML. Ukázka vstupního souboru RSS je

```
<?XML version="1.0"?>
```

```
<RSS version="2.0">
```

```
  <channel>
```

```
    <item>
```

```
      <title>Brno data</title>
```

```
      <link>http://www.adplaze.com/diplomka/batrla/brno.swf</link>
```

```
      <description>Multimediální data z Brna</description>
```

```
    </item>
```

```
  </channel>
```

```
</RSS>
```

Modul spouštění webového prohlížeče

Tento modul má za úkol spuštění webového prohlížeče v mobilním telefonu. Webový prohlížeč lze spustit přímo z aplikace J2ME. Spuštěnému prohlížeči je předána adresa odkazu, který byl přijat ze souboru RSS. Připojení k internetu je potenciálně nebezpečná operace a uživatel mobilního zařízení je vždy dotázán na povolení připojení. Může se stát, že mobilní klient nedostane od uživatele povolení k připojení webového prohlížeče. V tomto případě je nutné adekvátně zareagovat, vygenerovat upozornění a umožnit uživateli znovu se připojit k internetu.

Dekodér grafických kódů

Modul funguje jako rozpoznávač grafických dat. Poskytuje dekódování fotografie pořízené za pomoci fotoaparátu mobilního terminálu. Grafická data jsou přeložena na informaci o serveru, se kterým se bude komunikovat. Dekodér získá na vstupu obrázek a jako výstup je očekáván řetězec obsahující internetovou adresu lokačního serveru. Modul by měl mít schopnost přečíst základní průmyslové grafické kódy. Nabízí se použití standardů Datamatrix nebo Qrcode.

Modul SMS

Modul pro zasílání SMS zpráv. Uživatel může doporučit spuštěnou aplikaci svému kamarádovi. Z formulářů s vyplněným telefonním číslem a zprávou pro kamaráda je sestavena SMS zpráva. K této zprávě je přidána adresa odkazu pro stažení aplikace. Modul může sloužit v případě rozšíření aplikace na další doplňkové potřeby SMS komunikace. Zasílání SMS zpráv je potenciálně nebezpečná operace. Může se stát, že uživatel nedopatřením pošle placenou premium SMS zprávu a bude muset nechtěně zaslání hradit.

2.2.2.2 Logika aplikace

Při spuštění aplikace se objeví úvodní obrázek a ihned poté je uživatel přesměrován do základního grafického prostředí. Na začátku je vytvořeno připojení aplikace k výstupu z fotoaparátu integrovaného v mobilním telefonu. Výstup z fotoaparátu je zobrazen na obrazovce mobilního zařízení.

Vyfotografování

Po vybrání položky se provede pořízení fotografie ve standardním rozlišení a její dekódování na příslušný řetězec. Jestliže dojde k úspěšnému dekódování, je řetězec zobrazen na obrazovce mobilního telefonu a je nabídnuta možnost připojení mobilního telefonu k internetu nebo možnost uložení daného řetězce do záložek. Jestliže je připojení povoleno, je vytvořen a zapnut modul pro spuštění webového prohlížeče. Jestliže si uživatel vybere uložení do záložek, uloží se řetězec do interní databáze.

GPS připojení

Jestliže si uživatel vybere připojení k GPS, objeví se mu upozornění na otevření komunikace Bluetooth. Jestliže uživatel připojení nepovolí, je vytvořeno upozornění s textem o neúspěšném připojení a aplikace se dostává do základního menu. Jestliže je připojení povoleno, je vytvořen konektor Bluetooth, který vyhledává všechny Bluetooth zařízení v okolí. Aplikace se znovu dostává do hlavního menu.

Jakmile je vyhledávání ukončeno, uživateli se objeví upozornění, kde má možnost si vybrat jedno z nalezených zařízení a k tomuto zařízení se připojit. Jestliže je připojení k nalezené GPS povoleno, je dané připojení otevřeno a spustí se algoritmus, který jednou za předem daný čas nahraje data ve formátu NMEA z GPS. Z těchto dat je získána informace o poloze a uložena k dalšímu použití.

Záložky

Jestliže uživatel vybere položku záložky, má na výběr možnost nahrát dříve uloženou záložku. Záložky budou obsahovat samotná data a informace o vyfotografovaném. K dispozici je možnost záložky nahrávat z databáze i mazat.

Help

Jestliže uživatel vybere položku help, na obrazovce mobilního telefonu se objeví zobrazený textový a grafický popis práce s aplikací. Návod popisuje jak vyfotografovat čárový kód a jak se přes Bluetooth připojit k GPS.

About

V položce about je základní popis o lidech a open source knihovnách, které přispěli ke vzniku této aplikace.

Update

Jestliže chce uživatel zjistit, zda může využít vylepšenou verzi mobilního klientského software, vybere položku update. Po vybrání této položky se mobilní telefon připojí k internetu a zjistí dostupnost nové verze software.

Poskytovatel služby

Po vybrání této položky menu je k dispozici přepínač pro serverovou službu, která obsluhuje samotný proces získání informací z internetu. Implicitně je nastaven server, jehož popis bude následovat později. Lze však nastavit jiný server, na který může aplikace směřovat své dotazy.

Sdílení

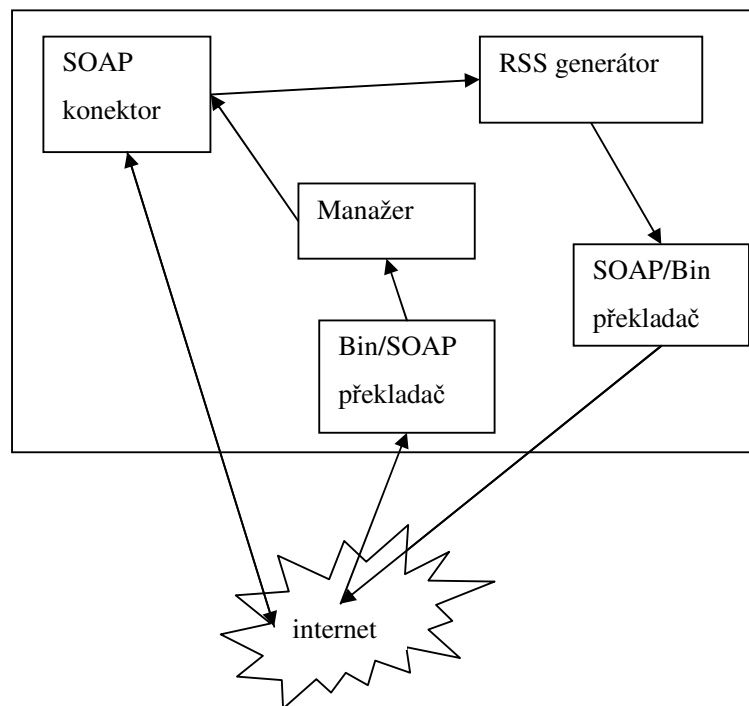
Po vybrání této sekce uživatelem je nabídnuta volba pro odeslání SMS kamarádovi s doporučením na aplikaci. Uživatel vyplní své jméno a telefonní číslo na něž chce doporučení odeslat.

Konec

Po vybrání položky konec je uživateli nabídnuta možnost ukončení programu, či návratu zpět do hlavního menu aplikace.

2.2.3 Serverová část

Serverová část se skládá z několika subkomponent, které se starají o komunikaci mezi klientským zařízením a lokačním serverem. RSS server je schopen přijmout data z mobilu a na základě nich se dotázat na lokační server a zjistit informace vztažené k dané poloze. Data, která jsou přijata ze serveru jsou převedena do formátu RSS a zaslána na mobilní zařízení.



Obrázek 21 - Rozdělení RSS serveru na moduly

2.2.3.1 Moduly

Modul generování RSS dat

Tento modul je schopen z odpovědi došlé z lokačního serveru tvořit textový soubor ve formátu RSS. RSS soubor obsahuje informace o verzi RSS a rozdělení dat na kanály. Každý kanál obsahuje kolekci položek item. Každá položka item definuje data vztahující se k dané poloze.

Item obsahuje následující položky

- Title
- Link
- Description

Title je krátký a výstižný název informace, která byla získána z lokačního serveru.

Link obsahuje URL (unifikovaný lokátor zdrojů) odkaz na vlastní data. Link může odkazovat jak na textová tak i obrazová či audio data.

Description popisuje vlastní data a umožní uživateli zjistit jaká data může očekávat po připojení mobilního telefonu k internetu.

Ukázka výstupního souboru

```
<?XML version="1.0"?>
<RSS version="2.0">
  <channel>
    <item>
      <title>Brno data</title>
      <link>http://www.adplaze.com/diplomka/batrla/brno.swf</link>
      <description>Multimedální data z Brna</description>
    </item>
  </channel>
</RSS>
```

Modul pro komunikaci SOAP

Úkolem tohoto modulu je komunikace mezi RSS serverem a serverem nabízejícím lokačně závislá data. WSDL (jazyk popisující webové služby) popisuje službu, která nabízí lokačně závislá data. Na zdrojovém textu níže je zachycena celá struktura souboru WSDL, ze kterého vývojové prostředí Netbeans vygenerovalo odpovídající modul pro komunikaci SOAP. V kódu jsou popsány názvy všech metod, které jsou klientem volány.

```
XMLns:tns="http://RSS/" XMLns:xsd="http://www.w3.org/2001/XMLSchema"
XMLns:SOAP="http://schemas.XMLSOAP.org/wsdl/SOAP/" targetNamespace="http://RSS/"
name="LocationRSSService">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://RSS/" schemaLocation="http://acer-
1f5adaf30f:8080/RSSLocationServer/LocationRSSService/__container$publishing$subctx/
WEB-INF/wsdl/LocationRSSService_schema1.xsd"
XMLns:wsdl="http://schemas.XMLSOAP.org/wsdl/"
XMLns:SOAP12="http://schemas.XMLSOAP.org/wsdl/SOAP12/">
    </xsd:schema>
  </types>
  <message name="getLocationRSS">
    <part name="parameters" element="tns:getLocationRSS"/>
  </message>
  <message name="getLocationRSSResponse">
    <part name="parameters" element="tns:getLocationRSSResponse"/>
  </message>
  <binding name="LocationRSSPortBinding" type="tns:LocationRSS">
    <SOAP:binding transport="http://schemas.XMLSOAP.org/SOAP/http" style="document"/>
    <operation name="getLocationRSS">
      <SOAP:operation SOAPAction=""/>
      <input>
        <SOAP:body use="literal"/>
      </input>
      <output>
        <SOAP:body use="literal"/>
      </output>
    </operation>
  </binding>
  <service name="LocationRSSService">
    <port name="LocationRSSPort" binding="tns:LocationRSSPortBinding">
      <SOAP:address location="http://acer-
1f5adaf30f:8080/RSSLocationServer/LocationRSSService"
XMLns:wsdl="http://schemas.XMLSOAP.org/wsdl/"
XMLns:SOAP12="http://schemas.XMLSOAP.org/wsdl/SOAP12/">
    </port>
  </service>
</definitions>
```

2.2.4 Vzájemná komunikace komponent

GPS modul komunikuje s mobilním telefonem přes rozhraní Bluetooth. Data, která vychází z GPS jsou k dispozici ve formátu NMEA. Získaná data jsou zaslána mobilním telefonem přes mobilní datovou síť. Mobilní telefon je propojen s RSS serverem přes protokol http. Data přicházející na server jsou uložena ve formátu RSS, který byl již dříve popsán. Při komunikaci směrem od mobilního telefonu k RSS serveru dochází k přenosu místně závislých dat. RSS server komunikuje za pomoci protokolu SOAP se serverem zajišťující lokačně závislá data.

2.3 Implementace

2.3.1 Mobilní aplikace

Mobilní aplikace je vytvořena ve vývojovém prostředí Netbeans 5.5 s nainstalovaným modulem Mobility Pack. Při implementaci jsem měl zapnutu podporu MIDP2.0 a CLDC 1.1. Logika aplikace byla editována přímo v prostředí Netbeans. Uživatelské rozhraní je implementováno návrhovým vzorem MVC (návrhový vzor Model2).

RSS

Ke zpracování souborů XML je použit open source projekt KXML, který je dostupý na internetu. KXML je knihovna s minimálními požadavky na paměť a výkon mobilního telefonu. S pomocí tohoto překladače jsem zpracovával příchozí data ve formátu RSS.

Multimediální operace

Vlastní čtecí cyklus probíhal ve smyčce, kdy se mobilní aplikace neustále snažila pořizovat fotografie a tyto fotografie ihned zpracovat a rozlišit případný zaznamenaný čárový kód. Smyčka byla implementována jako samostatné vlákno, které bylo po každém dekódovacím cyklu uspáno na 50 milisekund. K pořizení fotografie jsem použil příkaz

```
imageData = videoControl.getSnapshot(null);
```

s parametrem null, který říká že se pořídí fotografie s neurčitým rozlišením a ve formátu, který je pro danou implementaci Javy standardní. Na většině mobilních telefonů se jedná o fotografii zakódovanou pomocí JPEG komprese v rozlišení 120x160. Jestliže je potřeba pořídít fotografii ve větším rozlišení, je možno místo null v parametru předat řetězec definující dané rozlišení a použité kódování.

```
imageData = videoControl.getSnapshot("encoding=jpeg&width=480&height=640");
```

Za zmínku stojí, že ne každý mobilní telefon pracuje standardně s rozlišením 120x160. Například SE K750i fotí na šířku, protože poskytuje rozlišení 160x120.

Konektivita GPS a Bluetooth

Pro čtení dat z GPS je použito vlákno obstarávající vyhledávání okolních Bluetooth zařízení. Jakmile je dokončen proces vyhledání všech okolních Bluetooth zařízení, seznam dostupných zařízení se objeví na obrazovce mobilního telefonu. Uživatel si může vybrat jedno z jedno ze zařízení a aplikace se na něj připojí. Vytvoří se objekt pro čtení dat GPS a jejich zpracování. Tento objekt čte z GPS jednou za 60 sekund a aktualizuje polohu senzoru.

Standardní webový browser je možno spustit v Java za pomoci příkazu „System.platformRequest(URL)“, kde URL je je proměnná typu String a obsahuje URL řetězec.

Komunikace klient-server

Ke komunikaci se serverem jsem použil vygenerování zdrojového kódu wsdl služby. Tuto možnost poskytuje prostředí Netbeans a nabízí tak jednoduchou cestu jak vytvořit klienta webové služby na mobilním telefonu. Vlastní implementace této webové služby funguje tak, že je vytvořen servlet, který obaluje volání webové služby. Mobilní klient se zašle data na vzdálený servlet přes http. Servlet vytvoří obálku SOAP protokolu a zajistí komunikaci s webovou službou. Jakmile dostane servlet odpověď od webové služby, odstraní obálku SOAP a zašle získaná data do klientského zařízení. Použitá implementace umožňuje připojení mobilního klienta k jakékoliv webové službě. Jelikož je SOAP dotaz vytvářen až na serveru samotném, není mobilní telefon zatížen přenosem objemných dat spojených s použitím XML. Výhodou je tedy malá velikost přenášených dat.

2.3.2 Serverová část

K překladu lokačních dat do formátu RSS jsem využil volně dostupnou knihovnu jRSS. Tato knihovna je spolu se zdrojovými kódy k dispozici na internetové adrese <http://sourceforge.net/projects/jRSS/>. Serverová část je vytvořena jako webová služba.

Jakmile je z mobilního zařízení zaslán požadavek na data vztahující se k poloze, webová služba použije zadanou polohu pro dotaz na lokační server. Jako příklad dotazu na webový server je možno uvést

```
"http://www.adplaze.com/ws/ad_gate.rbx?ac=i&id=1&lat=0&lng=0&ua=&ip=";
```

Data, která server obdrží jsou uložena v následujícím formátu

```
**id_data_1*URL_data_1*popis_data_1*id_data_2*URL_data_2*popis_data_1*id_data_3*  
URL_data_3*popis_data_3*id_data_4*URL_data_4*popis_data_4*id_data_5*URL_data_5*  
popis_data_5**
```

Vstupní text jsem rozdělil separátorem * . Ukázka vytvořeného souboru RSS je

```
<?XML version="1.0"?>  
<RSS version="2.0">  
  <channel>  
    <item>  
      <title>id_data_1</title>  
      <link>URL_data_1</link>  
      <description>popis_data_1</description>  
    </item>  
    <item>  
      <title>id_data_2</title>  
      <link>URL_data_2</link>  
      <description>popis_data_2</description>  
    </item>  
    <item>  
      <title>id_data_3</title>  
      <link>URL_data_3</link>  
      <description>popis_data_3</description>  
    </item>  
    <item>  
      <title>id_data_4</title>  
      <link>URL_data_4</link>  
      <description>popis_data_4</description>  
    </item>  
    <item>  
      <title>id_data_5</title>  
      <link>URL_data_5</link>  
      <description>popis_data_5</description>  
    </item>  
  </channel>  
</RSS>
```

3 Testování

3.1 Použité výpočetní prostředky

3.1.1 Mobilní platformy

Velkou výzvou v prostředí mobilních aplikací je diverzita možností a schopností mobilních telefonů. Mnoho mobilních telefonů se od sebe navzájem liší a jsou něčím jedinečné. Některé telefony pracují s proprietárním operačním systémem, jiné naopak pracují s Symbian či Windows mobile. Některé mobilní telefony nabízejí základní rozlišení 120x160 bodů v J2ME, některé pak nabízejí rozlišení i 640x480 bodů. Omezením prostředí J2ME je neschopnost ovládat zaostření fotoaparátu. Dvojměrné čárové kódy pak musí být vytisknuty ve větší velikosti, abychom získali ostrý obraz kódu. Mobilní telefony se také liší v nabízeném připojení. Některé mobilní telefony pracují s GPRS nebo s EDGE, některé již podporují HSDPA či Wi-fi připojení. Při těchto rozdílech v rychlosti připojení je pak velký rozdíl v časech potřebných ke stažení požadovaných dat. V této kapitole blíže seznámíme čtenáře s hardwarem mobilních zařízení, na kterých bylo prováděno testování implementované aplikace. Mobilní telefony byly vybrány jako průřez trhu. Testoval jsem jak na špičkově vybavených mobilních telefonech, tak i na mobilních telefonech střední a nízké třídy.

Nokia N73

Mobilní aplikace byla testována na mobilním telefonu Nokia N73. Tento mobilní telefon je vybaven operačním systémem Symbian OS 9.1, který však platforma Java nevyužívá naplno. Mobilní telefon poskytuje rozlišení 3.15 megapixelů, tj. 2048x1536 pixelů a kvalitní optiku Carl Zeiss s autofocusem. Mobilní telefon N73 umožňuje spouštět aplikace naprogramované v prostředí Flash. Tento mobilní telefon je schopen pracovat s PIM, což je podpora prací s kontakty, kalendářem a tiskem. N73 umí pracovat v sítích 3. generace, poskytuje připojení GPRS a EDGE. Rychlost Javy dle testu v programu jBenchmark je 3828 bodů pro MIDP1.0 a 343 bodů pro MIDP2.0.

Sony-Ericsson K750i

Mobilní telefon SEK750i nabízí verzi Javy MIDP2.0 a profil CLDC1.1. Mobilní telefon umí pracovat s čísly s pohyblivou desetinou čárkou a nabízí podporu MMAPi (knihovna pro práci s multimédií). Mobilní telefon podporuje knihovnu WMA pro práci s SMS a knihovnu pro komunikaci Bluetooth. Mobilní telefon neumožňuje přehrávání Flash programů. Rychlost Javy dle testu v programu jBenchmark je 3892 bodů pro MIDP1.0 a 337 bodů pro MIDP2.0.

Sony Ericsson K700i

Mobilní telefon SEK700i nabízí verzi Javy MIDP2.0 a profil CLDC1.1. Stejně jako SEK750i nabízí podporu práce s multimédií a SMS. Tento model telefonu však neumožňuje práci s Bluetooth, což nám neumožňuje určovat polohu mobilního zařízení. Z tohoto důvodu byly při práci s tímto mobilním telefonem zadávány polohové údaje ručně. Rychlost Javy dle testu v programu jBenchmark je 2691 bodů pro MIDP1.0 a 132 bodů pro MIDP2.0.

MDA Pro

Mobilní telefon MDA Pro nabízí verzi Javy MIDP1.0 s profilem CLDC1.1. Na tomto mobilním telefonu pracuje operační systém Windows mobile verze 5.0. Telefon obsahuje Bluetooth a fotoaparát s rozlišením 1 280 × 960 (1,23 Mpx) pixelů. Mobilní telefon poskytuje podporu WMA pro práci s SMS zprávami. Mobilní prohlížeč na tomto telefonu je Internet Explorer verze 6.0 s nainstalovaným Flash 9.0 pluginem.

Senzor GPS

Měření bylo prováděno s GPS přijímačem Mediatek, který má 32 kanálů pro příjem GPS signálu. Jako zajímavost bych uvedl, že GPS senzor má k dispozici kombinované napájení z baterie a ze solárního článku.

Protocol	NMEA0183 GGA, CHN, GSV, RMC, VTG
Baud rate	9600, N, 8, 1
Update rate	1 Hz
Datum	WGS84
Channel	32 channel
Frequency	L1, 1575.42 Mhz
Position accuracy	< 3m CEP
Antena type	Built-in ceramic
External antenna interface	MMCX
Bluetooth	Class 2, range 10m

Tabulka 1 – Specifikace hardware senzoru GPS
Získáno z technické specifikace senzoru



Obrázek 20 – Fotografie testovacího senzoru GPS

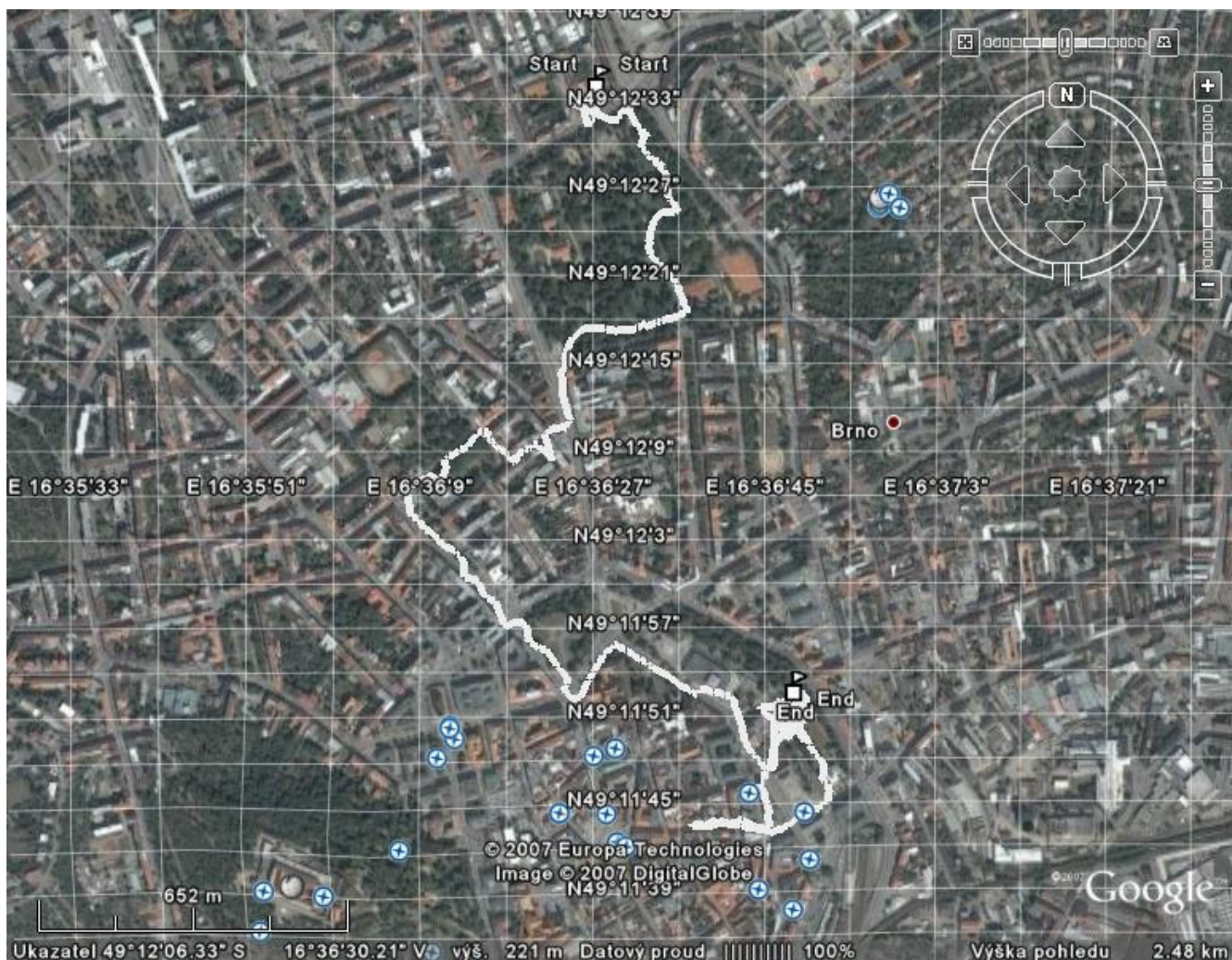
3.1.2 Serverová platforma

Server byl spuštěn na notebooku s procesorem AMD 3100+ . Notebook je vybaven operační pamětí o velikosti 512 MB a harddiskem s přístupovou dobou 11ms. Použitý operační systém je Windows XP SP2. Jako server byl při testu použit produkt Bundled Tomcat 5.5.17. Notebook byl při testu připojen k internetu přes linku s propustností 1Mb/s. Server je implementován a kompilován pro verzi Javy 1.6. Samotný výstup - war soubor byl umístěn na serveru platformy Tomcat.

3.2 Testování implementovaného systému

3.2.1 Test příjmu GPS signálu

Ke zjištění spolehlivosti a přesnosti implementovaného řešení jsem provedl dvě sady testů. K vlastnímu testování jsem použil program GPSTrack 1.0. Aplikace se umí připojit k GPS za pomoci Bluetooth a číst data o poloze senzoru. GPSTrack nabízí rozhraní pro ukládání a export lokačních dat a proto jsem jej zvolil jako nástroj pro testování. Program jsem nastavil na záznam hodnot po 10 sekundách. Testování jsem prováděl na velkém území města Brna. Trasu ze zastávky Pionýrská v Brně na zastávku Husitská nebo pak ještě dále na Kolejní, jsem absolvoval vícekrát. Měření jsem prováděl i na cestě do Nového Lískovce, kde lze nalézt vysoké budovy. Se senzorem jsem se nacházel i v Černých polích. Testovacích dat bylo téměř 1600. Jedná se tedy o téměř 4,5 hodiny záznamu. Výsledná data jsem získal spojením několika exportů dat ve formátu CSV (hodnoty oddělené středníkem). Formát CSV je vhodný pro čtení běžnými tabulkovými procesory. Avšak vhodnější by bylo data exportovat do formátu, který podporuje rozhraní Google maps. Výsledná data bych tak mohl jednoduše zobrazit v příjemném grafickém rozhraní. V městské zástavbě je signál ovlivněn odrazem od vysokých budov a také zde není k dispozici přímá viditelnost k většímu počtu satelitů. Při odchodu z domu a zapnutí senzoru ještě v blízkosti řady 5 patrových budov jsem často nebyl schopen dosáhnout fix satelitů. Zajímavostí je, že jsem v některých případech fix satelitů nezískal ani po celé cestě z ulice Střední na areál FIT. V Lužáneckém parku jsem však byl schopen získat přesná data a rychlé zafixování téměř kdykoliv, jelikož jsem měl běžně signál z 6 satelitů. V některých místech na kopci v Králově Poli jsem měl signál i z 10 satelitů. Občas se mě však stávalo, že jsem naměřil velikou odchylku od hodnot, které byly zaznamenány v měření předchozím. Jednalo se o výstřelový šum, který občas zcela změnil naměřenou hodnotu. Obrázek 21 byl získán v programu Google Earth po převodu dat z CSV do formátu vhodného pro program firmy Google.

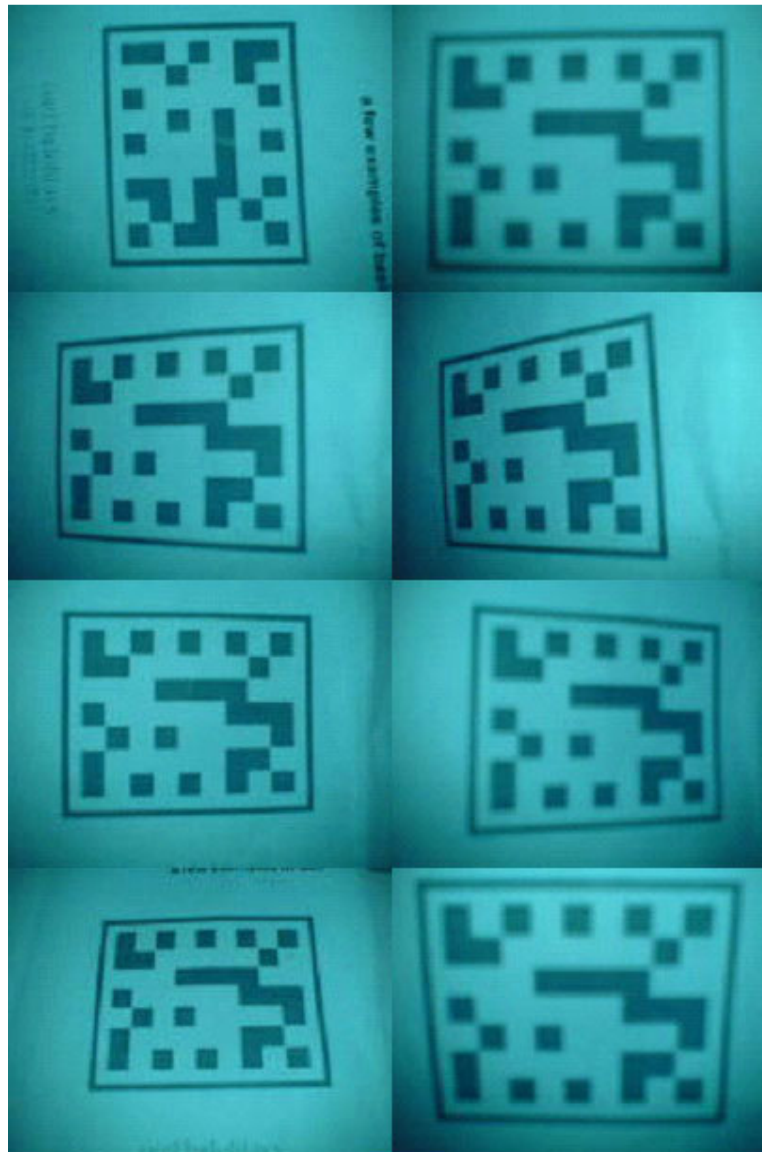


Obrázek 21 – Vyznačení testovací trasy ve městě Brně

3.2.2 Test kvality dekódování 2D čárových kódů

Tento pokus má za cíl zjistit úspěšnost dekódování dvojrozměrného čárového kódu při různých úhlech natočení, vzdálenosti, rozlišení a osvětlení. Celkem byla schopnost dekódovat grafický kód testována na vzorku 120 fotografií. Dvojrozměrný čárový kód byl vytisknut na bílém podkladě a kódoval URL adresu serveru, který zpracovával dotazy s obsahem GPS souřadnic. Adresa serveru je <http://www.adplaze.com/ws>. Velikost vytisknutého čárového kódu byla 40x30 mm.

Ke čtení dvojrozměrných čárových kódů jsem použil tři různé knihovny pro čtení dvojrozměrných kódů. Pro čtení kódů Qrcode jsem použil opensource knihovnu qrcode, jež může být nalezena na internetové stránce <http://qrcode.sourceforge.jp>. Knihovna umí spolehlivě číst kódy QRCode o rozměru 21x21 modulů. Při vyfotografování kódu pod úhlem 45 stupňů však není opensource verze schopna fotografii rozkódovat. Problémem je, že použitý algoritmus nedokáže v některých situacích poznat špatně přečtená data a neupozorní na chybu rozpoznání. Dále není dostatečně vyřešena schopnost dekodování kódů při zakódování velkého objemu dat.



Obrázek 21 – Ukázka reálných fotografií z mobilního telefonu při rozlišení 120x160 bodů

Pro testování schopnosti číst dvojrozměrné kódy Datamatrix jsem použil prostředky firmy Semacode, které je pro nekomerční využití k dispozici na adrese www.semacode.org. Tato knihovna umí číst spolehlivě kódy, které kódují větší počet dat. Knihovna Semacode dokáže při správném vyfotografování přečíst i kódy o rozměrech 52x52 modulů. Takovýto kód již poskytuje kapacitu, do níž je možné uložit celou SMS zprávu, dlouhé URL, či email. Semacode algoritmus se umí vypořádat s jakýmkoliv natočením kódu kolem středové osy a zvládne dekodovat i kódy vyfotografované našikmo.

Nakonec jsem testoval knihovnu vznikající na VUT-FIT. Knihovna umí číst relativně jednoduché kódy, avšak to nemá větší vliv na schopnost dekodovacího procesu. Grafické kódy z FIT jsou také pěkně čitelné i při horším zaostření a menší velikosti. Při malé velikosti grafického kódu však dochází ke zhoršení schopnosti rozpoznání grafického kódu, protože dekodovací algoritmus je založen na detekci rohů kódu, které mohou být nezřetelné.



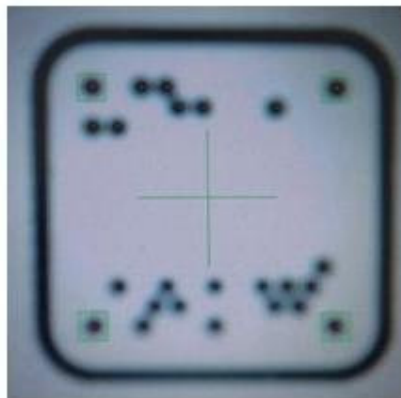
Obrázek 22 – Fotografie grafického kódu z FIT, speech

Z následujících dvou obrázků je jasně vidět problém standardizovaných kódů, které jsou při větším přiblížení a menší velikosti špatně čitelné. Velikost modulu u Qrkódu není velká a mobilní telefon, který pořizuje fotografie z prostředí Java má problémy s jeho zaostřením. Problém se dá odstranit spojnou čočkou, ale řešení to není vhodné. Ke správnému čtení je potřeba dodatečný hardware.



Obrázek 23 – Fotografie kódu QRCode

Naopak grafické kódy Beetag jsou rozeznatelné i při vyfotografování zblízky. Velikost modulu, což je nejmenší ploška nesoucí data, je větší než u Qrkódu či Datamatrix. Správnému dekódování napomáhá rozložení grafických dat na hexagonální mřížce. Podobného hexagonálního uspořádání se dnes užívá i při stavbě pokročilých čipů pro digitální fotoaparáty.



Obrázek 24 – Fotografie grafického kódu Beetag

3.2.3 Test rychlosti zpracování

Dalším testem, který jsem provedl bylo testování celkové rychlosti zpracování požadavku. Měřena byla průměrná délka doby od vyfotografování kódu přes jeho dekódování, odeslání na server a zpětného přijetí náležitých informací. Tyto informace jsem sbíral z toho důvodu, abych zjistil, kolik času ušetří práce s aplikací obyčejnému uživateli oproti využití přímo wapového browseru a nařukání informace do formuláře. Testoval jsem na dříve popsaných mobilních telefonech.

Mobilní zařízení	Velikost přenášeného souboru	Rychlost spuštění
K750i	3KB	20 sekund
K750i	20KB	25 sekund
K750i	50KB	30 sekund
N73	3KB	25sekund
N73	20KB	27sekund
N73	50kB	31sekund

Tabulka 2 - Naměřené hodnoty času od spuštění aplikace po příjem dat ze server

4 Zhodnocení

4.1 Výhody a nevýhody implementace

Mezi výhody navrženého systému patří nezávislost na předem stanoveném serveru. Jelikož získáváme informace o internetové adrese serveru pomocí čtení dvojrozměrného kódu, je možno mít v čárovém kódu uloženou jakoukoliv internetovou adresu. Navržené řešení poskytuje do budoucna otevřenost systému pro další aplikace a nové projekty postavené na prototypu. Služby s přidanou hodnotou ocení přítomnost informace o poloze uživatele. Navigační služby jsou užitečné aplikace a ve spojení s grafickými kódy se otevírají zcela nové možnosti. Implementovaná podpora zasílání SMS zpráv patří také mezi výhody. V samotném kódu může být zakódována SMS zpráva i s číslem, na něž čtečka zprávu pošle. Tato funkce je vhodná i v případě možnosti zpoplatnění služeb. Výhodou je relativní přenositelnost grafického uživatelského rozhraní. Navrhoval jsem GUI s důrazem na jednoduchost a maximální nezávislost na výrobci či typu mobilního telefonu. Výhodné je použití dekodéru pro čtení kódů Datamatrix. Tyto kódy se již nyní objevují na potisku obalů například nápojů. V budoucnu lze očekávat větší rozvoj kódů Datamatrix z důvodu jeho otevřenosti.

Nevýhodou systému z technického pohledu je implementace na platformě Java. V první řadě jsou podporovány pouze mobilní telefony s MIDP2.0. Mobilní telefony s operačním systémem Symbian či Windows Mobile poskytují vysokou rychlost zpracování, přiblížení obrazu a v neposlední řadě větší komfort. Systém je omezen fyzickou velikostí grafických kódů. Jestliže jsou kódy moc malé, nejdou přečíst a jsou rozmazané. Při velkém přiblížení se ze čtvercových bodů stávají nezřetelné šmouhy. Nevýhodou je sférické zkreslení fotoaparátů. Fotoaparáty v mobilech způsobují při fotografování z velké blízkosti soudkovitost grafického kódu. Jestliže je zakřivení moc velké, může nastat problém s dekodováním kódu. Nevýhodou je použití externího modulu GPS. Protože senzor GPS není zabudován v mobilním telefonu, je nutné komunikovat s GPS přes Bluetooth. Použití GPS jako jediného zdroje lokačních informací má své nevýhody. Signál GPS je možno zachytit pouze v nízké zástavbě či na prostranství. Mezi vysokými budovami nebo přímo v budovách nebudeme mít k dispozici informace o poloze mobilního telefonu.

4.2 Návrh vylepšení

Implementovaný systém by bylo vhodné rozšířit o schopnost čtení jednorozměrných kódů. Problémem je však nedokonalost kamer zabudovaných v dnešních mobilních telefonech. Jako o prvním vylepšení by se dalo uvažovat o možnosti ovládat zaostření kamery mobilu. Jestliže máme k dispozici ostrý obraz, pak rozkódovat informaci v kódu nebývá problém. Obrázek ilustruje typickou fotografii z fotoaparátu s vypnutým makrorežimem. Na následujícím obrázku můžeme vidět fotografii zaostřeného čárového kódu při zapnutí makrorežimu ve fotoaparátu mobilu.



Obrázek 25 – reálná fotografie čárového kódu EAN

<http://www.smartphonethoughts.com/images/Kris-Aura-SP8bcno.jpg>

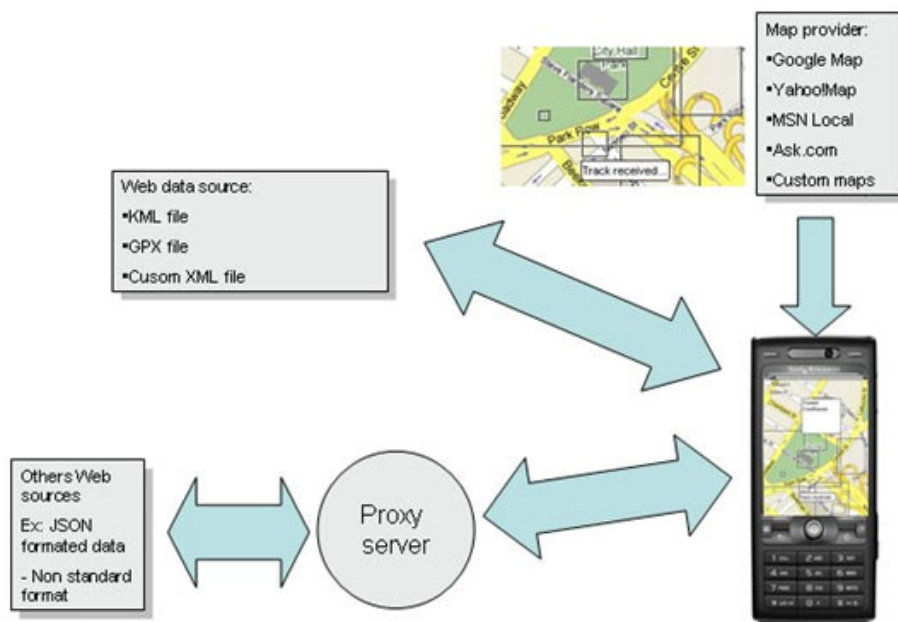


Obrázek 26 – Fotografie EAN kódu vyfotografovaná přes optickou soustavou

<http://www.smartphonethoughts.com/images/Kris-Aura-SP9sony.jpg>

Bohužel lze s pomocí MMAPi v Java získat pouze obrázek bez možnosti zaostření. V tomto případě se nabízí možnosti mobilů s Win Mobile nebo Symbian.

Jako další přirozené vylepšení bych navrhoval spojení implementovaného systému s mapami. Uživatel by získal možnost orientace ve svém okolí a například by mohl fotografováním dvojrozměrných kódů přímo do mapy zaznamenat informace o poloze kódu. Stejně tak by uživatel mohl získat informace o nejbližších místech, kde se kódy vyskytují. Ve spojitosti s implementací map bych rád upozornil na projekt <http://J2MEmap.landspurg.net/>. Tento projekt pracuje jako klient pro mapové podklady třeba od Google nebo od Yahoo či jiných poskytovatelů. Implementace J2MEmap systému je jednoduchá a ve spojení s již implementovanou podporou GPS bychom mohli získat zajímavý nástroj. Na obrázku je zobrazena základní struktura systému J2MEmap.



Obrázek 27 – Struktura systému J2MEmap

<http://J2MEmap.landspurg.net/pics/picsCdg.jpg>

Jako o dalším možném vylepšení bychom mohli mluvit o možnosti rozpoznávat více druhů grafických kódů. V kapitole, kde jsme popisovali dvojrozměrné kódy bylo ukázáno několik standardních i proprietárních kódů. Výhodou čtečky by byla možnost rozpoznat více možných variant kódů. Nabízí se podpora kódů Qrcode, Datamatrix, Shotcode a Beetag. Tyto kódy v dnešní době patří mezi nerozšířenější.

Dále bychom mohli uvažovat o zabudování text to speech technologie. Jestliže bychom získali ze serveru data v textové podobě, mohl by na nás mobilní telefon promluvit. Již dnes se v navigačních systémech používá hlasu pro navádění, a proto by mohl náš program použít podobných zpestření. Navíc jsou dvojrozměrné čárové kódy občas nečitelné a proto bývají označeny dodatečně i číselně. Mohli bychom použít techniky rozpoznávání řeči a uživatel by jednoduše mohl číslice nadiktovat přímo do mikrofonu mobilního telefonu.

5 Závěr

V diplomové práci jsme se seznámili s webovými a mobilními technologiemi. Stejně tak zde byly nastíněny technologie které využívají senzory obrazu a polohy. Zjistili jsme, jakým způsobem je možné zaměřit polohu mobilního telefonu.

V diplomové práci byli zmíněny možnosti zjištění polohy, které využívají vlastnosti mobilních. Stejně tak jsme se popsali technologie zjištění polohy založené na satelitech, GPS. V práci byly naznačeny výhody a nevýhody obou typů systémů a popsán systém hybridní, který využívá výhod výše zmíněných přístupů.

V textu jsme také seznámili čtenáře s grafickými kódy, které se stále častěji využívají v oblasti skladů a například v Japonsku jsou využívány i jako zkratka pro přístup na mobilní internet. Člověk pouze vyfotografuje grafický kód a ihned se připojí na internetovou stránku, která ke kódu náleží.

V kapitolách, které následovaly po úvodní teoretické části byl navržen systém, který umožňuje mobilnímu uživateli získat informace závislé na poloze mobilního telefonu a vyfotografovaném grafickém kódu. Tento systém byl navržen jako klient-server aplikace a maximálně využíval platformy Java pro mobilní telefony.

Při implementaci navrženého systému jsem využil komunikaci Bluetooth, práci s SMS zprávami i multimediální funkce mobilního telefonu. Naučil jsem se programovat grafická uživatelská rozhraní pro platformy mobilních telefonů a také jsem se jak propojit senzor polohy s mobilem.

Při testování jsem narazil na problémy s přesností přijímaného signálu. Tento problém s přesností nastává při pohybu v terénu, kde není přímá viditelnost mezi GPS a satelity. Jestliže se GPS senzor nachází například mezi vysokými budovami, může být zjištění polohy obtížné. V práci jsme také sledovali, jak kvalitní je knihovna, kterou jsem použil k dekódování grafických kódů. Zjistili jsme, že použitá knihovna je schopna dekódovat relativně veliké grafické kódy, které mohou nést internetovou adresu vzdáleného serveru nebo obsahovat informace například pro zaslání SMS na určité telefonní číslo. Při testech celkové rychlosti jsme zjistili, že rychlost dekódování kódu velmi záleží na kvalitě pořízené fotografie a celkovém výkonu mobilního zařízení. Dospěli jsme k názoru, že již s dnešními mobilními telefony je možno vytvořit systém, který kombinuje rozpoznávání obrazových informací a technologie zjištění polohy mobilního telefonu.

Na konci práce jsou uvedeny výhody a nevýhody navrženého přístupu a implementace. Výhody, které aplikace nabízí převažují nedostatky aplikace. Jako největší nedostatek bych však označil možná až přílišnou složitost celého procesu spuštění Java aplikace, spuštění vyhledávání okolních Bluetooth zařízení, připojení k GPS, vyfotografování kódu a následného otevření internetového prohlížeče. Pro mnohé mobilní uživatele se jedná o zbytečný a zdlouhavý proces. Nicméně systém je myšlenkově velmi zajímavý a při implementaci více spolupracující s hardwarem mobilu by bylo možné celý proces omezit na pouhé stisknutí tlačítka pro vyfotografování smímku grafického kódu. Nakonec práce jsme diskutovali o možných vylepšeních, které by usnadnili rozpoznání grafických kódů, nebo vylepšili zjišťování polohy mobilního telefonu. Při práci na popisovaném systému jsem se seznámil s mnoha technologiemi a vymoženostmi, které nabízí prostředí mobilních telefonů a prostředí internetu.

Seznam literatury

- [1] *Java ME Platform Overview* [online]. 2006 [cit. 2007-04-19]. Dostupný z WWW: <<http://Java.sun.com/Javame/technology/index.jsp>>.
- [2] *Java Overview* [online]. 2005 [cit. 2007-04-19]. Dostupný z WWW: <<http://developers.sun.com/techttopics/mobility/overview.html>>.
- [3] *PNG* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/PNG>>.
- [4] *JPG* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/JPG>>.
- [5] *RSS* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/RSS>>.
- [6] *SOAP* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/SOAP>>.
- [7] *SOAP Specification* [online]. 2004 [cit. 2007-04-19]. Dostupný z WWW: <www.w3.org/TR/soap/>..
- [8] *Trilateraton* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/Trilateration>>.
- [9] *Trilangulation* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/Triangulation>>.
- [10] *Multilateraiton* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/Multilateration>>.
- [11] ORLICH, Martin. *Základní lokační metody v GSM* [online]. 2006 [cit. 2007-04-19]. Dostupný z WWW: <<http://access.feld.cvut.cz/view.php?cisloclanku=2006022801>>.
- [12] RAHMANY, Ahmed. *Introduction to GPS The Global positioning system*. 2002th edition. [s.l.] : [s.n.], 2002. 749 s. ISBN 1-58053-183-0.
- [13] *QRCode* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/QRCode>>.

[14] *Datamatrix* [online]. 2007 [cit. 2007-04-19]. Dostupný z WWW:
<<http://en.wikipedia.org/wiki/Datamatrix>>.

[15] Migrating Wireless Applications to MIDP 2 WMA and MMA. *Nokia letter* [online]. 2006 [cit. 2007-04-19].

[16] MUSIL, Jiří. *Modulární architektura distribuovaných systémů*. [s.l.], 2007. 59 s. VUT-FIT. Diplomová práce.

Seznam tabulek

Tabulka 3 – Specifikace hardware senzoru GPS.....43

Tabulka 4 - Naměřené hodnoty času od spuštění aplikace po příjem dat ze server.....48

Seznam obrázků

Obrázek 1 - Struktura platformy Java ME grasia.fdi.ucm.es/j2me/image	4
Obrázek 2 – Kruhový grafický kód Shotcode.....	7
Obrázek 3 – Grafický kód z FITu	7
Obrázek 4 – Grafický kód standardu Datamatrix	8
Obrázek 5 – Grafický kód standardu Qrcode	8
Obrázek 6 – Barevný kód Datamatrix	9
Obrázek 7 - Rozdělení metod lokalizace mobilních zařízení	12
Obrázek 8 - Vysvětlení principu triangulace	12
Obrázek 9 - Trilaterace ve 2D prostoru	13
Obrázek 10 - Trilaterace ve 3D prostoru může mít více řešení	15
Obrázek 11 - Lokalizace mobilu dle sektoru BTS	17
Obrázek 12 - Lokalizace s výpočtem časové prodlevy.....	18
Obrázek 13 - LMU synchronizuje BTS	18
Obrázek 14 - Lokalizace dle úhlu dopadu signálu	19
Obrázek 15 - Signály z GPS družic vytvářejí v prostoru pomyslné koule.....	20
Obrázek 16 - Segmenty systému GPS	21
Obrázek 17 - Zakódování GPS signálu.....	21
Obrázek 18 - Architektura aplikace.....	25
Obrázek 19 - Ukázka z modulu pro vkládání a polohy dat a její definici	26
Obrázek 20 – Fotografie testovacího senzoru GPS	42
Obrázek 21 – Ukázka reálných fotografií z mobilního telefonu při rozlišení 120	45
Obrázek 22 – Fotografie grafického kódu z FIT, speech	46
Obrázek 23 – Fotografie kódu QRCode.....	47
Obrázek 24 – Fotografie grafického kódu Beetag.....	47
Obrázek 25 – reálná fotografie čárového kódu EAN	50
Obrázek 26 – Fotografie EAN kódu vyfotografovaná přes optickou soustavou.....	50
Obrázek 27 – Struktura systému J2MEmap	51