

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

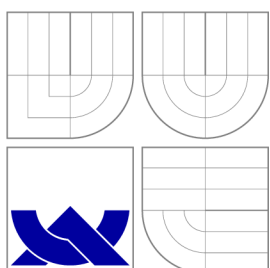
PLÁNOVAČ UDÁLOSTÍ ZALOŽENÝ NA POLOZE PRO N900

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

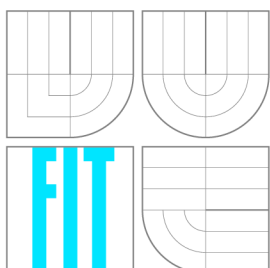
AUTOR PRÁCE
AUTHOR

LUKÁŠ ARON

BRNO 2011



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

PLÁNOVAČ UDÁLOSTÍ ZALOŽENÝ NA POLOZE PRO N900

GEOREMINDER FOR N900

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

VEDOUCÍ PRÁCE

SUPERVISOR

LUKÁŠ ARON

Ing. JOZEF MLÍCH

BRNO 2011

Abstrakt

Tato práce se zabývá návrhem aplikace pro mobilní zařízení Nokia N900, která umožňuje naplánování sady činností vykonávaných v předem definovaných oblastech. Tyto oblasti či konkrétní pozice definuje uživatel dle souřadnic systému GPS. Při dosažení definované oblasti je vykonána sada akcí, které mohou zahrnovat spuštění upozornění ve formě alarmu nebo spuštění aplikace. Práce popisuje teorii k mobilnímu zařízení a možnosti jak získat pozici tohoto zařízení. Součástí je popis implementace aplikace, kde je rozepsán postup vývoje a návrh grafického uživatelského rozhraní. Rovněž jsou zde jsou shrnuty dosažené výsledky a návrhy na pokračování.

Abstract

This work deals with a design of an application for the mobile device Nokia N900, which enables the planning of a set of actions, which shall be carried out in a predefined region. The region or specific position is defined by the user based on the coordinates of the GPS system. Upon reaching of the target location a set of actions, which is defined when creating the notification, is carried out. The work describes the theory of the mobile device and the options of obtaining the position of such a device. A description of the implementation of the application follows, where the method and the graphical user interface are described. In the conclusion of the work, the achieved results and a proposal for continuation of the work are summarised.

Klíčová slova

Nokia, N900, Maemo, WGS-84, GPS, A-GPS, Qt, Mobility API

Keywords

Nokia, N900, Maemo, WGS-84, GPS, A-GPS, Qt, Mobility API

Citace

Lukáš Aron: Plánovač událostí založený
na poloze pro N900, bakalářská práce, Brno, FIT VUT v Brně, 2011

Plánovač událostí založený na poloze pro N900

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Jozefa Mlícha. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Lukáš Aron
12. května 2011

Poděkování

Děkuji za odborné a trpělivé vedení mé bakalářské práce vedoucímu Ing. Jozefu Mlíchovi.

© Lukáš Aron, 2011.

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

1 Úvod	3
2 Mobilní zařízení	5
2.1 Nokia N900	5
2.2 Vývojové nástroje pro mobilní zařízení	7
2.3 Operační systém Maemo 5	10
3 Geografické systémy	12
3.1 WGS-84	12
3.2 Zjištění pozice	13
3.3 Výpočet vzdálenosti mezi dvěma body	15
4 Implementace	16
4.1 Grafická část aplikace	17
4.2 Aplikace běžící v pozadí	21
4.3 Data	21
4.4 Ostatní části	27
5 Dosažené výsledky	28
5.1 Přesnost získání pozice	28
5.2 Uživatelské rozhraní	30
5.3 Možné rozšíření	31
6 Závěr	34
A UML diagram tříd	37
B Obsah CD	39

Seznam obrázků

2.1	Zařízení Nokia N900 z různých pohledů	6
2.2	Hardwarová část mobilního zařízení Nokia N900	7
2.3	Schéma vývoje s využitím cross-kompilace	9
2.4	Softwarová část mobilního zařízení Nokia N900 (jádro operačního systému Maemo)	11
3.1	Standard WGS-84	13
3.2	Dráhy satelitních družic na oběžné dráze	14
4.1	Návrh hlavní grafické části aplikace	17
4.2	Návrh základního menu grafické části aplikace	18
4.3	Návrhu grafické části pro práci s pozicemi	18
4.4	Návrh grafické části pro práci se šablonami	19
4.5	Návrh grafické části pro práci s akcemi	19
4.6	Návrh grafické části pro práci s upozorněními	20
4.7	Návrh grafické části pro nastavení aplikace běžící v pozadí	20
4.8	UML diagram tříd aplikace běžící na pozadí	22
4.9	ER Diagram navržené databáze	23
4.10	Popis informací při získávání pozice	24
5.1	Horizontální přesnost GPS systému u statického bodu	29
5.2	Horizontální přesnost GPS systému při jízdě automobilem	29
5.3	Horizontální přesnost GPS systému při chůzi	30
5.4	Návrh dialogu pro manipulaci s pozicemi	32
5.5	Návrhu dialogu s mapou pro korekci pozice	32
A.1	UML diagram tříd grafické části aplikace	38

Kapitola 1

Úvod

Při dnešním technologickém pokroku a rozvoji mobilních zařízení, disponujících GPS modulem, je kladen důraz na výkon a výdrž baterie či mobilitu zařízení. Změna vzhledu a funkční výbavy je obvyklá u každého výrobce. Jsou ovšem nastaveny určité představy, až téměř standardy dnešní doby, které specifikují, co by měl mobilní telefon obsahovat. Většina výrobců se je snaží dodržet a přidat mnohem více dalších doplňků. Pokud se pominou základní služby, jako je volání či zasílání krátkých textových zpráv (sms) a zasílání multimediálních zpráv (mms), je trendem moderní doby umožnit přehrát hudbu nebo video v jakémkoliv formátu, možnost fotografovat v co největším rozlišení včetně využití blesku, být neustále v kontaktu s internetem za použití bezdrátových technologií Wi-Fi a v neposlední řadě určit pozici za pomoci GPS modulu.

Výkon mobilních zařízení stále stoupá a lze tedy na telefonu používat modernější operační systémy s velmi komplikovaným grafickým uživatelským rozhraním reagujícím na velké množství podnětů a samotné aplikace, které využívají hardwarové doplňky. Výhodou tohoto nárůstu je využití telefonu jako multimediálního centra na cestách. Před nemnoha lety sloužil telefon jen na volání a posílání krátkých textových zpráv. Dnes je možné spustit několik aplikací najednou. Například využívat připojení na internet přes bezdrátové rozhraní a zároveň poslouchat hudbu z přehrávače a současně psát krátkou textovou zprávu. V souvislosti s výkonem telefonu je požadavek na co nejdelší výdrž baterie. Tato snaha je v rozporu s výkonem. Kompromis je obvykle nalezen tak, že většinu doplňkového hardwaru, jako je například bezdrátové připojení Wi-Fi či GPS modul, lze vypnout a zapnout pouze v případě potřeby.

Pozice mobilního zařízení je získána za pomoci GPS modulu, který je většinou součástí těchto zařízení. Tento modul patří mezi nejznámější systémy jak získat pozici. S využitím systému pro získání pozice je dnes spojeno mnoho aktivit, jako například navigace na cestách či velice oblíbený geocaching. Nabízí se mnoho možností využití, ale i přes funkční hardwarovou jednotku (modul) je stále mnoho nápadů a ideí, jak využít pozici ve speciálních aplikacích. Ty se zabývají jiným přístupem, než jen navigací z bodu *A* do bodu *B*.

Většina uživatelů mobilních zařízení s GPS modulem používá vyhledávání pozice minimálně nebo vůbec. Existuje několik možných důvodů proč to tak je. Příkladem je, že uživatel má v zařízení implicitně nahrán software, který umí pouze vyhledat aktuální pozici či trasu mezi danými body. Není zde jiná možnost jak využít systém pro získání pozice? Funkčnost tohoto modulu je přesně daná a nelze ji upravovat po hardwarové stránce. Samotná aplikace (software) může na souřadnice nahlížet i jinak, než zjištění trasy mezi dvěma danými body.

Příkladem jiného použití se nabízí například kniha jízd, kde se při určitém časovém okamžiku uloží daná pozice do databáze, odkud se po daném období data vyzvednou a zpracují dalším softwarem. Další využití je například souhrnná databáze pozic kamarádů, kde se pozice nahrávají na server. Zároveň se stahují data obsahující aktuální pozice kamarádů, kteří data nahráli dříve, lze tedy vidět, kde se nachází nebo případně jak daleko jsou další uživatelé stejného softwaru a stejné komunikační skupiny.

Cílem této bakalářské práce je vytvořit aplikaci pracující s GPS modulem pro mobilní zařízení Nokia N900, která by umožňovala vytvářet upozornění na dané pozice. Tyto pozice by si uživatel sám spravoval. Aplikace by měla obsahovat jednoduché a přívětivé grafické uživatelské rozhraní, které by uživateli umožňovalo spravovat samotné geografické pozice, vytvářet upozornění a akce, které budou vykonané při dosažení konkrétní pozice. Další součástí by mělo být řízení spouštění událostí, které ovládá vykonávání akcí při dosažení pozice, na které je vytvořeno upozornění. Aplikace by měla být distribuována formou balíčku, který lze stáhnout z vývojového repozitáře balíčků oficiálního zdroje společnosti Nokia.

Součástí je vytvoření demonstračního videa pro ukázkou funkčnosti aplikace na zařízení Nokia N900. Při vývoji je využit Qt Framework od společnosti Nokia, který slouží k vytvoření grafického uživatelského rozhraní samotné aplikace. Po vytvoření aplikace bude možné stáhnout instalační balíček z oficiálního zdroje, jak bylo zmíněno dříve. Zároveň se bude vytvářet zpětná vazba od uživatelů této aplikace na oficiálním fóru společnosti Nokia. Použití aplikace je víceúčelové. Mezi základní použití lze uvést uložení příchoďů a odchodů ze zaměstnání, nákupní seznam, hledání dárečků v dnešní době velice populárním geocachingem. Věřím, že uživatel této aplikace nalezne i velké množství jiných možností využití.

Pro lepší orientaci v textu je publikace členěna následovně. Kapitola 2 obsahuje základní informace o mobilním zařízení Nokia N900 a jeho přednostech. Součástí je základní popis operačního systému Maemo 5. V kapitole 3 lze nalézt popis standardu WGS-84, ze kterého vychází dnes nejpoužívanější systém GPS, a popis samotné funkčnosti systému polohování na Zemi. Tato kapitola obsahuje také výpočet vzdálenosti bodu, který je v aplikaci hojně využíván. Návrh aplikace a popis implementace je popsán v kapitole 4, kde je zobrazen i grafický návrh uživatelského rozhraní a souhrn samotných částí aplikace tak, jak byly následně vyvíjeny.

Výsledek je prezentován v kapitole 5, kde je shrnuto, jak bylo splněno zadání. Nachází se zde i případná odezva na výslednou aplikaci od uživatelů. Součástí této kapitoly je shrnutí dosažených výsledků. V závěru lze nalézt celkové hodnocení práce a její přínos s případnými možnostmi, jak aplikaci v budoucnu vylepšit či co by bylo vhodné řešit jinak.

Kapitola 2

Mobilní zařízení

Obsahem kapitoly je základní popis mobilního zařízení Nokia N900 [12]. Mobilní zařízení je popsáno nejprve po hardwarové stránce. Je zde uveden základní popis vybavení telefonu a vše poté ještě jednou zopakováno formou schématu.

Následující popis je věnován softwarové stránce, kde je jednoduchý popis základní výbavy. Nemá smysl rozebírat softwarovou tematiku mobilního zařízení příliš do detailů z důvodu možného přizpůsobení, které je individuální každému uživateli. Přesto je zobrazeno schéma kernelu (jádra) operačního systému Maemo 5 Fremantle [2, 16], které by se nemělo v jednotlivých mobilních zařízeních odlišovat.

Dalším zajímavým tématem kapitoly je popis vývojových nástrojů umožňujících tvorbu vlastních aplikací pro mobilní zařízení.

2.1 Nokia N900

Mobilní zařízení Nokia N900 je smartphone (chytrý telefon, kapesní počítač) vyrobený společností Nokia a uvedený na trh v roce 2010. Vzhled telefonu z různých pohledů je zobrazen na obrázku 2.1, který je převzat z [15]. Nokia N900 vychází z předchozího modelu Nokia N810. Popis hardwarové části mobilního telefonu Nokia N900 lze nalézt na obrázku 2.2, který je přejat z [16]. Obrázek zahrnuje téměř úplný výpis všech hardwarových součástí a zároveň jejich propojení do jednoho celku.

Hlavní součástí zařízení je například procesor ARM Cortex-A8, taktován na frekvenci 600 MHz, který je postaven na architektuře ARMv7 poskytující výjimečný výkon a umožňující spouštět všechny aplikace rychle, hladce a současně.

Další důležitou součástí je čip TWL4030, poskytující funkcionalitu jako je například rozhraní pro hardwarovou klávesnici, real-time hodiny, audio kodeky, zesilovače apod.

Paměťovou náročnost splňuje 256 MB flash čip a 32 GB eMMC čip, který je dostačující pro uložení až 7 000 oblíbených skladeb nebo 40 hodin videa v DVD kvalitě. Paměť lze rozšířit o externí microSD kartu rozšiřující kapacitu až na 48 GB.

Interaktivita s uživatelem je zajištěna pomocí 3,5" dotykového displeje a plnohodnotné hardwarové klávesnice, Bluetooth rozhraním či bezdrátovou technologií Wi-Fi. Poprvé je použito vysokorychlostní širokopásmové připojení, které může nabývat takových rychlostí, jaké jsou připojení internetu v domácnostech.

Mobilní zařízení nabízí využití bohatého interaktivního prohlížení webového obsahu s využitím nejmodernějšího softwaru a velké dotykové obrazovky. Telefon nabízí mnoho multimediálních prvků, které nejsou běžné na všech zařízeních. Příkladem může být mož-

nost zachytit obraz nebo video v DVD kvalitě pomocí širokoúhlého integrovaného 5 Mpx objektivu značky Carl Zeiss. Optické čočky přináší přidanou jasnost, zatímco dvojitý LED blesk umožňuje pořizovat snímky ve dne i v noci.

Vestavěný asistovaný A-GPS přijímač najde rychle a přesně pozici zařízení. V případě využití dodávané aplikace Ovi Maps lze vyhledat nejrychlejší trasu z bodu *A* do bodu *B*.

Výkonný hardware je zabalený do designově podobného pouzdra. Nokia N900 je mobilní počítač vybavený ostrým dotykovým displejem disponujícím rozlišením 800 x 480 pixelů a plně integrovanou QWERTY klávesnicí.

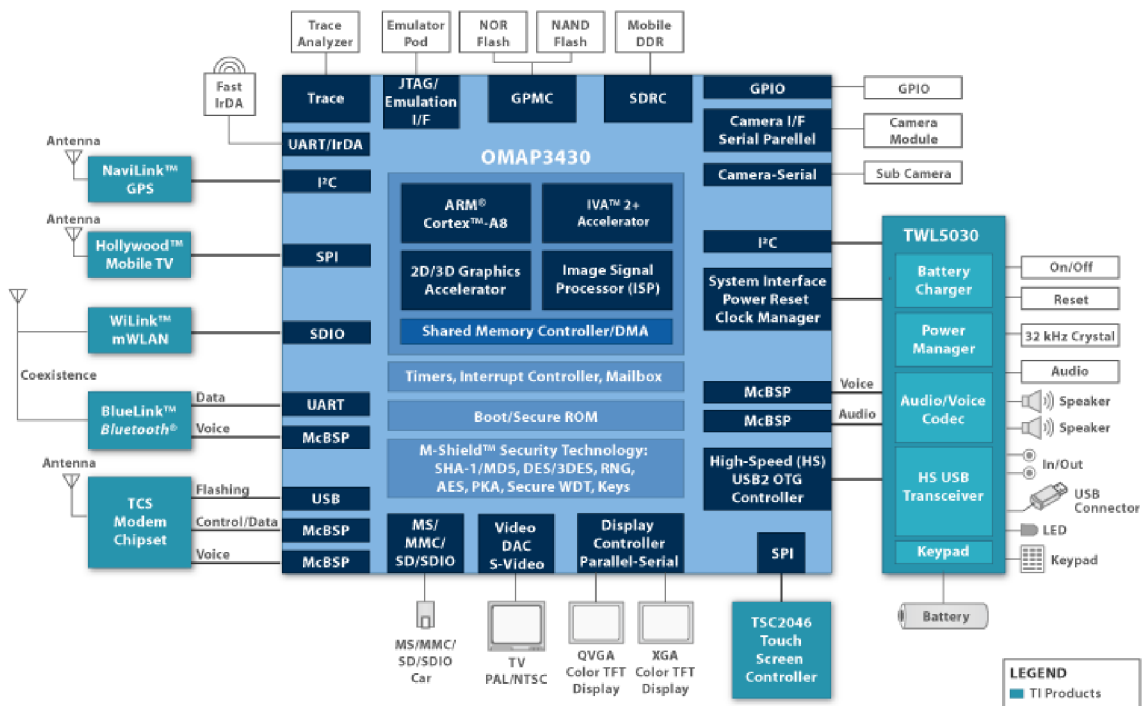


Obrázek 2.1: Zařízení Nokia N900 z různých pohledů

Softwarová výbava mobilního zařízení je individuální dle uživatele. Základní a implicitně předinstalovaný operační systém je Maemo 5 Fremantle, jež vychází z oblíbené linuxové distribuce Debian [6, 9].

Jelikož se jedná o otevřenou platformu, kterou lze volně upravovat, nelze se spolehnout na to, že každé mobilní zařízení bude obsahovat stejný software. Bohužel i přes to, že se jedná o otevřenou platformu (open source), jsou některé části uzavřené. Zejména se jedná o aplikace umožňující konektivitu či management napájení.

Předpokladem je, že většina uživatelů nebude měnit jádro (kernel) operačního systému Maemo. Lze jej tedy popsat v kapitole 2.3. Implicitně předinstalovaný operační systém Maemo není prázdný, ale obsahuje velké množství předinstalovaného softwaru. Ten umožňuje využít všechny hardwarové části. Příkladem může být internetový prohlížeč, multimediální přehrávač či Ovi Maps. Aplikace je možné odstranit a zároveň nainstalovat nové, včetně možnosti nainstalovat aplikace třetích stran. Informace byly čerpány z oficiálních internetových stránek mobilního zařízení Nokia N900 [15] a odborného časopisu [12].



Obrázek 2.2: Hardwarová část mobilního zařízení Nokia N900

2.2 Vývojové nástroje pro mobilní zařízení

Každý výrobce mobilních zařízení nabízí vývojové prostředí sloužící k tvorbě aplikací, které jsou specifické pro dané zařízení. Vývojové prostředí je nejčastěji nazýváno Software Development Kit (SDK). Vývoj aplikací pro mobilní zařízení je mírně odlišný od vývoje pro klasický počítač. Důraz je kladen na paměťové nároky a efektivitu výpočtu, aby zařízení nebylo příliš zatěžováno a nebyla zkracována doba výdrže baterie. Podporována je většina dnes používaných programovacích jazyků, ale přednostně jsou podporovány jazyky umožňující objektově orientovaný přístup. Při vývoji softwaru pro platformu Linux lze využít vývojová prostředí, která nejsou přímo určena výrobcem. Příkladem může být vývojové, konfigurační a kompilační prostředí Scratchbox.

Maemo SDK

Společnost Nokia poskytuje programovací prostředí Maemo SDK umožňující využít všechny možnosti a prvky platformy Maemo, které obsahuje například mobilní zařízení Nokia N900. Vývojové prostředí Maemo SDK je založeno na projektu Scratchbox, který je popsán jako samostatný vývojový nástroj pro mobilní zařízení.

Maemo SDK zahrnuje software a nástroje potřebné k vytváření aplikací psaných v jazyce C a využívajících knihovnu GTK+ a Qt tak, aby aplikace byly plně integrovány do platformy Maemo. Součástí jsou všechny nástroje, které vývojáři potřebují pro psaní, testování, kompilování a vytváření instalačních balíčků pro distribuci Maemo.

Vývojové prostředí se dělí na více menších částí, které lze nazývat systémy. Mezi největší systémy lze zařadit: knihovnu aplikací Hildon, systém multimedií, systém řešící připojení, systém komunikace v reálném čase, správu aplikací, jádro systému a aplikační rozhraní.

Maemo SDK podporuje vývoj aplikací, využívajících knihovnu Qt Mobility API, vyvinutu speciálně pro plné a jednoduché využití mobilních technologií, stejně jako jazyk C a C++. Vývojové prostředí umožňuje prozkoumat bohaté uživatelské rozhraní, aplikace a funkce, které jsou již implementovány v operačním systému. Dnes je doporučeno použít pro vývoj aplikací pro platformu Maemo vývojové prostředí Nokia Qt SDK, které je popsáno jako samostatný vývojový nástroj pro mobilní platformy. Informace jsou čerpány z [17].

Scratchbox

Scratchbox je konfigurační a kompilační prostředí pro vytváření aplikací Linuxu a tvorbu samotných distribucí Linuxu. Jedná se o univerzální kompilační nástroj pro překlad vestavěných linuxových aplikací. Uspodňuje práci při integraci a překladu pro jinou architekturu dané linuxové distribuce. Byl používán například k vývoji operačního systému Maemo, ale není striktně určen pouze k tomuto použití. Jeho hlavní předností je podpora mnoha typů architektur procesorů, například ARM a x86 (PowerPC, MIPS a další). Možnost kompilovat software pro rozdílné architektury, než na které je vytvářen, se nazývá cross-kompilace.

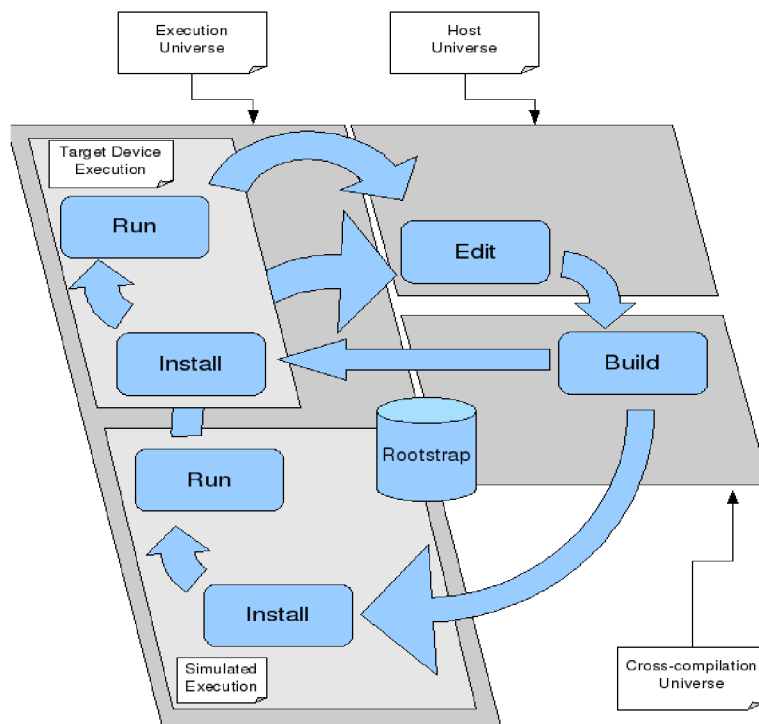
Cross-kompilace je známý pojem, který má dlouhou historii sahající až téměř do počátků rozvoje počítačů. V úplných počátcích rozvoje počítačů nebyl potřebný překlad na jiné architektury, jelikož software byl psán pro konkrétní stroje. Plnil pouze jeden účel či funkci. Používání stejného kódu opakovaně nebylo běžné. Čas strávený úpravou kódu pro danou architekturu procesoru byl nahrazen cross-kompilací, která použije stejný kód, ale přeloží jej tak, aby bylo možné výsledný software spustit na cílené platformě.

Základní myšlenkou cross-kompilace je použití procesoru k sestavení softwaru pro některé další procesory, které používají jinou architekturu. To znamená, že stroj, na kterém je sestavení softwaru provedeno, nemůže nativně tento software spouštět. Využití cross-kompilace je využíváno při snaze urychlit překlad pro cílovou architekturu, i když je možný nativní překlad. Příkladem může být překlad operačního systému pro mobilní zařízení, kde lze systém nativně přeložit pro architekturu ARM. Stejný překlad lze provést s využitím cross-kompilace na mnohonásobně rychlejším procesoru a ušetřit tak čas. Obrázek 2.3 demonstruje, jak probíhá vývoj softwaru s využitím cross-kompilace, obrázek je přejatý z [17]. Na počítači, kde probíhá vývoj (Host Universe), se napíše zdrojový kód aplikace. Tento zdrojový kód se přeloží pro cílovou platformu s využitím cross-kompilace a jeho spuštění lze simulovat ve virtualizovaném prostředí, nebo spouštět na cílové platformě (Execution Universe).

Část rootstrap je nástroj pro budování kompletního obrazu souborového systému Linux. Rootstrap nevyžaduje práva administrátora systému (root) či podobná privilegia toho druhu. Tvorba souborového systému s využitím rootstrap je rychlá a velice často využívaná. Podrobnější informace k nástroji rootstrap lze nalézt v [6].

V souvislosti s vývojem softwaru pro mobilní zařízení Nokia N900 a operační systém Maemo je vydáno vývojové prostředí, využívající znalosti Scratchbox, které se jmenuje Maemo Application Development and Debugging Environment (MADDE).

MADDE je vývojový nástroj, vycházející z projektu Scratchbox, vyvinutý společností Nokia ve spolupráci s komunitou vývojářů operačního systému Maemo. MADDE je využito pro překlad softwaru volaného z vývojového prostředí Nokia Qt SDK. Podrobnější informace k projektu Scratchbox nebo vývoji vestavěných linuxových systémů lze nalézt v literatuře [7].



Obrázek 2.3: Schéma vývoje s využitím cross-kompilace

Nokia Qt SDK a Qt Mobility API

Qt [8, 13] je komplexní C++ aplikační vývojový nástroj pro tvorbu aplikací včetně grafického uživatelského rozhraní a možnosti využití cross-kompilace. Qt umožňuje vývojářům použít stejný zdrojový kód softwaru a přeložit jej pro různé operační systémy. Podporovány jsou operační systémy Windows 98 až Windows 7, Mac OS X, Linux, Solaris, HP-UX či Unix obsahující X11. I když bylo vývojové prostředí vyvinuto pomocí C++, je poskytováno oficiální rozhraní pro ostatní programovací jazyky. Podporovány jsou zejména programovací jazyky Python, Ruby či technologie .NET.

Obsahem prostředí není pouze možnost jednoduše a rychle navrhnout grafické uživatelské rozhraní v editoru Designer, ale jedná se o naprosto přepracovanou standardní knihovnu pro implementační jazyk. Je zároveň optimalizovaná pro práci s grafickým rozhraním. Knihovna byla využita například při tvorbě grafického uživatelského rozhraní KDE používaného v mnoha linuxových distribucích.

Pro potřeby mobilních technologií bylo vytvořeno nové vývojové prostředí Nokia Qt SDK, které je zeštíhlenou verzí vývojového prostředí specializovaného na vývoj softwaru pro mobilní platformy. Nejvíce podporovány jsou platformy, které společnost Nokia sama vyvíjí, konkrétně se jedná o operační systém Symbian a Maemo. Pro realizaci výsledného překladu a tvorby instalačního balíčku pro cílovou platformu je využit upravený projekt Scratchbox. Scratchbox je upraven dle cílové platformy, která není součástí běžné instalace vývojového nástroje.

Součástí Nokia SDK je nástroj umožňující simulovat cílovou platformu včetně operačního systému, na které lze vyzkoušet vyvíjený software. Simulace probíhá v aplikaci Simulator. Ta umožňuje využití mnoha aspektů chování, které se mohou vyskytnout při

používání softwaru na cílovém mobilním zařízení, jako je například vybití baterie, připojení k internetu, změna pozice či porucha systému.

Důležitou a velice často vyhledávanou a používanou součástí vývojového prostředí Nokia SDK je Mobility API. Obsahuje velké množství velice často používaných postupů a metod využívaných pro ovládání specifických částí mobilního zařízení. Projekt Mobility přináší řadu nových rozhraní API, která jsou dobře známá ze světa mobilních zařízení. Tento projekt sjednocuje rozhraní pro komunikaci mezi hardwarem mobilního zařízení a softwarovou aplikací. Výhoda tohoto přístupu je vítána v případě tvorby softwaru pro více platform. Příkladem může být aplikace, která bez úpravy zdrojového kódu bude přeložitelná a zároveň spustitelná na mobilních zařízeních s operačním systémem Maemo, Symbian či na nově vznikající distribuci MeeGo.

Mobility API poskytuje vývojáři širokou škálu funkcí pro požadovanou mobilní platformu. Projekt je rozdělen na více jednotlivých částí, kde každá část zasahuje do jiné problematiky. Mezi nejpoužívanější jednotlivé části lze zařadit API pro kontakty, multimédia, management připojení, systémové informace, lokalizace a zprávy.

2.3 Operační systém Maemo 5

Maemo [2, 16, 7] je pokročilý operační systém založený na platformě Linux a určený pro nasazení do vyspělých mobilních počítačů. Především se jedná o zeštíhlenou verzi linuxové distribuce Debian přizpůsobenou pro potřeby přenosných zařízení. Komunita spravující operační systém Maemo umožňuje na základě platformy s otevřeným zdrojovým kódem upravovat a neustále vyvíjet software dle vlastního uvážení daného vývojáře. Podmínkou pro začlenění dané úpravy či přidání funkčnosti je přinést zlepšení a nové možnosti všem uživatelům tohoto systému bez nároku na finanční prostředky.

Maemo je otevřená platforma, mající ve svém jádru Linux, ale zároveň udržuje velké množství dalších softwarových komponent u komunity open source vývojářů. Tato platforma je založena na známých technologiích, které jsou široce používány v komunitě svobodného softwaru. Tvůrčí tým, vytvářející a spravující systém Maemo, obsahuje více než 19 000 registrovaných členů, kteří přispívají k více než 800 projektům.

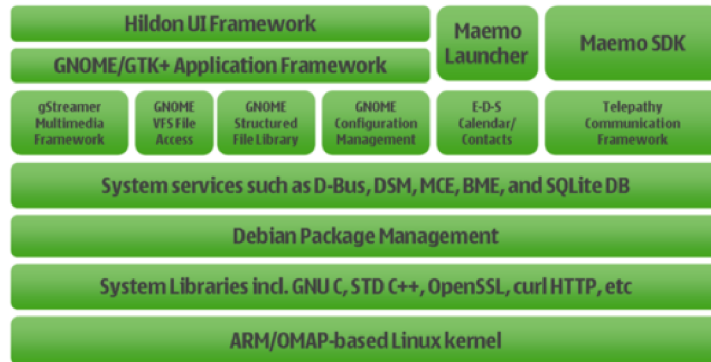
V současnosti jej implicitně používají mobilní zařízení Nokia N770, Nokia N800, Nokia N810, Nokia N810 wimax edition a Nokia N900 [15, 12]. Aktuální verze operačního systému je Maemo 5 Fremantle, která je s největší pravděpodobností poslední verzí operačního systému Maemo. Vývoj probíhal převážně ve společnosti Nokia ve spolupráci s mnoha open source projekty, kde mezi nejznámější patří jádro Linux, distribuce Debian či grafické uživatelské rozhraní GNOME.

Uživatelské rozhraní je rozděleno do dvou skupin. Na prvky, které jsou pevně dané a prvky, které jsou na volbě uživatele. Mezi pevně dané prvky patří například menu, které obsahuje hodiny, název operátora, nastavení budíku, indikace baterie a indikace dalších služeb (například BlueTooth). Veškerý další obsah je na volbě uživatele. K dispozici jsou celé čtyři obrazovky, které lze naplnit zástupci jednotlivých instalovaných programů, widgetů, odkazů na internetové stránky či RSS čtečky atd. Uživatel dostává naprostou volnost pro přizpůsobení přístroje svým potřebám, a tím i jistou konkurenční výhodu.

Pro vytvoření grafického prostředí je použit X Window System, který používá Xomap server a správce oken vycházející z projektu Matchbox. Pro grafické uživatelské rozhraní je využita knihovna Hildon sloužící převážně pro widgety a API. Knihovna Hildon je nadstavbou grafické knihovny GNOME/GTK, zahrnující vytvoření jednoduché plochy, sadu widgetů optimalizovaných pro přenosné zařízení, sadu nástrojů upravujících vzhled a velké

množství dalších doplňujících knihoven a aplikací. Primárním zvukovým serverem je použit ESD a GStreamer, který je využíván přehrávačem médií. Podporované formáty lze rozšířit využitím pluginů, které jsou dostupné z repozitáře společnosti Nokia, jedná se například o rozšíření podpory o audio formát OGG.

Přestože je Maemo postaveno na Linuxu a open source, zůstaly některé části uzavřené. Mezi uzavřené části lze zařadit hlavní status bar, taskbar aplety (jas displeje apod.) a aplikace řešící konektivitu a management napájení. Pro doplnění základních informací o operačním systému je na obrázku 2.4 zobrazen diagram základních komponentů operačního systému Maemo. Obrázek je přejetý z [15].



Obrázek 2.4: Softwarová část mobilního zařízení Nokia N900 (jádro operačního systému Maemo)

Protože je Maemo postaveno na linuxové distribuci Debian, lze využít při návrhu balíčků stejných principů a pravidel tak, jak je tomu u linuxové distribuce, ze které vychází.

Debian je jednou z nejstarších doposud vyvíjených distribucí GNU/Linuxu, kterou vyvíjí především velké množství dobrovolníků z celého světa. Přestože je tato distribuce velice konzervativní, patří mezi uživateli k nejrozšířenějším a zároveň nejoblíbenějším.

Obsah textu vychází z článku [2] a podrobnější informace lze hledat na oficiálních stránkách operačního systému Maemo [16].

Kapitola 3

Geografické systémy

Hlavním obsahem kapitoly je popis systému zjištění pozice, který je zejména zaměřen na systém GPS [3, 14]. Tento systém lze chápat i jako službu, která je poskytována zdarma pro všechna zařízení disponující GPS modulem. Tato služba není časově omezená a lze ji využívat bez ohledu na čas a počasí. Pro doplnění dalších možností jak získat pozici jsou popsány základy dalších systémů.

S pojmem pozice souvisí velice důležitá informace. Je to vzdálenost od jiné pozice. Vzdálenost dvou pozic lze vypočítat dle vzorce, který je popsán v kapitole. Výpočet je realizován za pomoci systému WGS-84 [4, 5].

Systém WGS-84 je celosvětově uznávaný geodetický standard, který definuje souřadnicový systém. Podrobnosti lze nalézt dále v textu, jedná se zejména o jeho funkčnost a účelnost.

V kontextu této práce je time management chápán jako plánování času v závislosti na geografické poloze mobilního zařízení. Následně je proto chápán pouze jako doplňující pojem, který byl využit při vývoji aplikace. Problematika time managementu zde není vysvětlena, protože spadá mimo zájem a obor tvořené práce.

3.1 WGS-84

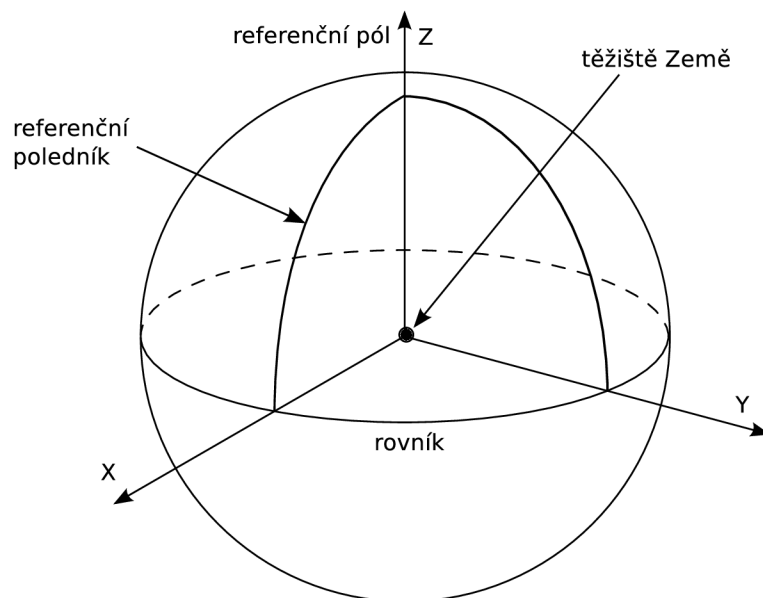
Vzorec pro výpočet vzdálenosti využívá geodetický standard World Geodetic System 1984 (WGS-84), který je celosvětově uznávaným standardem pro použití v kartografii, geodézii a navigaci. Vydalo jej Ministerstvo obrany USA roku 1984. WGS-84 definuje souřadnicový systém a referenční rotační elipsoid pro geodézii a navigaci.

V současné době standard WGS-84 využívá modelu gravitačního pole Země definovaného v roce 1996. Model gravitačního pole Země (EGM96) byl naposledy revidován v roce 2004, kde došlo ke zpřesnění modelu k aktuálnímu stavu gravitačního pole.

Samotné spojení rotačního elipsoidu se Zemí je takové, že počátek souřadnic leží v těžišti Země, osa Z odpovídá ose rotace Země, osa Y leží v rovině rovníku 90° východně od osy X . Spojení referenčního rotačního elipsoidu s reálnou Zemí je dáno souřadnicemi základních stanic systému WGS-84 rozmístěných po celé zeměkouli.

Pro reálnější představu je zobrazen standard WGS-84 na obrázku 3.1, který je přejatým z [3]. Standard WGS-84 je výchozím a referenčním modelem pro Global Positioning System neboli GPS, který je schopný získat pozici zařízení s přesností ± 1 cm.

Poslední revidovaná verze standardu WGS-84 z roku 2004 začíná být zastaralá a neodpovídá nárokům dnešní moderní doby, což jsou především rychlost a přesnost. Z tohoto



Obrázek 3.1: Standard WGS-84

důvodu vzniká revize nová. Nová revize upřesňuje výpočty pozic, které jsou důležité pro orbitální satelity, ale mají praktický vliv na typické topografické použití. Zpřesnění lze očekávat také u modelu gravitačního pole Země, které bylo naposledy revidováno v roce 2004. Informace jsou čerpány z literatury [3], kde lze nalézt podrobnější informace k dané problematice. Velice zajímavé informace lze též nalézt v [4].

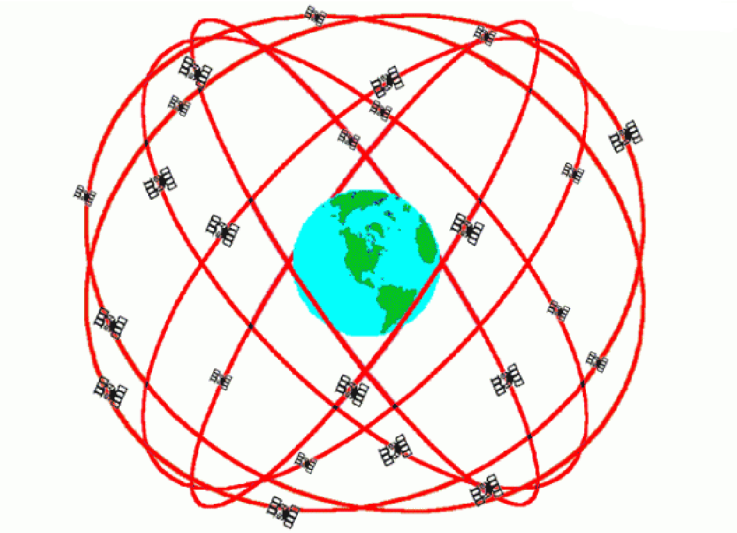
3.2 Zjištění pozice

GPS

Název systému GPS označuje zkratka NAVSTAR GPS [3, 5, 14], která je v celém názvu Navigation Satellite Timing and Ranging Global Position System neboli globální systém sloužící k určování pozice na Zemi. Systém je veden ministerstvem obrany USA a původně sloužil pouze pro potřeby armády Spojených států amerických. Technologie GPS byla na počátku využívána jen jako přesný vojenský lokalizační a navigační prostředek pro sledování pozic vojenských jednotek, zaměřování cílů, apod. V 80. letech 20. století americká vláda rozhodla o jeho uvolnění i pro civilní účely. Poté došlo k mohutnému rozšíření technologie GPS do všech oblastí lidské činnosti.

Pro pokrytí signálem GPS celé Země bylo projektováno využívání 24 satelitních družic. Potřeba získat přesnější pozici si ale vyžádala přidat dalších osm satelitních družic na aktuální počet 32. Tyto družice obíhají ve výšce 20 200 km nad povrchem Země na šesti kruhových drahách se sklonem 55° . Dráhy jsou vzájemně posunuty o 60° a na každé dráze je pět až šest nepravidelně rozmístěných pozic pro družice. Družice váží asi 1,8 tuny a na střední oběžné dráze se pohybuje rychlostí 3,8 km/s, s dobou oběhu kolem Země 11h 58min (polovina siderického dne).

Demonstrační obrázek 3.2 zobrazuje jak vypadají dráhy družic na oběžné dráze pro původní (projektovaný) návrh. Obrázek je přejat z [3].



Obrázek 3.2: Dráhy satelitních družic na oběžné dráze

Pro určení dvojrozměrné pozice (nejčastěji zeměpisná délka a šířka) postačí příjem signálu z minimálně tří družic (výpočet tří pseudovzdáleností), pro určení trojrozměrné pozice (navíc výška) minimálně ze čtyř družic. Příjem menšího počtu družic znemožňuje výpočet pozice, vyšší počet družic naopak určení pozice dále zpřesňuje. Na základě pasivity GPS přijímačů (pouze přijímají data, ale nevysílají zpět k satelitům) je systém GPS schopen obsloužit neomezený počet uživatelů a nelze jej zaměřit nepřitelem.

A-GPS

A-GPS [3] nebo někdy také aGPS [3, 5] je zkratka Assisted GPS a slouží k rychlejšímu počítačnickému výpočtu pozice. Protože signál navigačních družic špatně proniká do budov, byl vyvolán zájem o bezdrátovou podporu GPS – Wireless Assisted-GPS (A-GPS) a o vysoce citlivé přijímače GPS. To vše s požadavkem na HW/SW integraci do mobilních zařízení.

Je-li signál z družic slabý, nemusí se vždy podařit dekodování navigačního souboru dat. V takovém případě mají udržet pozici v běžícím navigačním systému údaje z mobilní sítě. Jedná se o ekvivalentní údaje družic vyslané do časoměrně-dálkoměrného systému z mobilní sítě.

Technologie A-GPS je založena na IP technologii. Lokalizační server komunikuje přímo s mobilním přístrojem prostřednictvím protokolu vyšší vrstvy IP, který je zaveden do mobilní sítě. Ústřední a rádiová část mobilní sítě tak zůstává beze změny. Výměna informací mezi telefonem a lokalizačním serverem probíhá přes běžná datová spojení, např. GPRS. Toto řešení tedy využívá protokoly a rozhraní, které již v ústřední a rádiové části sítě existují.

Již zmíněné lokalizační servery jsou schopny vyhodnotit pozici mobilního telefonu přímo na zemi, provést některé potřebné početní úkony a zjištěné výsledky spojit se signálem satelitu GPS. Tím tedy odpadne problém nedostupnosti satelitního signálu ve stíněných oblastech. Pozici v takovém případě nebude určovat satelit, ale telefon a odvodí se podle okolních stanic BTS. Tato informace se zpracuje a lokalizační server již zajistí komunikaci se satelitem.

Konečné údaje jsou poté poskytovány do mobilního telefonu opět po bezdrátové mobilní síti. Pomocí této technologie získává mobilní zařízení data o oběžných drahách, frekvencích a provozních schopnostech satelitů z mobilní sítě a dokáže tak analyzovat i slabší signály ze satelitů během několika sekund. U samotné GPS trvá zjištění informace poněkud delší dobu. S A-GPS uživatel pouze zadá požadavek, výpočty se provedou na lokalizačních serverech a zpět je poslán pouze výsledek. Vše trvá jen několik málo sekund, což šetří nejen čas, ale také energii.

Nevýhodou základní technologie A-GPS je, že její přesnost je závislá na pokrytí dané oblasti BTS stanicemi. Tato nevýhoda je vyřešena kombinací získání základní pozice mobilní sítě a následným zpřesněním pomocí signálu z orbitálních družic.

Další možnosti

Další možností jak definovat pozici zařízení je využití bezdrátových sítí - Wi-Fi. Při použití metody "otisku signálu", která je nejpoužívanější při hledání pozice s využitím bezdrátových sítí, lze narazit na nedostatky signálu těchto sítí. Tento způsob vyhledávání by byl vhodný do větších měst, kde lze předpokládat hustou síť bezdrátových technologií. Problém by se dostavil po použití v odlehlejších místech Země, jako jsou například hory, moře či údolí, kde není dostatek nebo dokonce není zavedeno žádné bezdrátové vysílání.

Poslední popisovaná metoda pro získání pozice je založena na základě znalosti IP adresy daného zařízení. Přesnost pozice je v tomto případě velice sporná. Ale přesto lze s jistotou lokalizovat územní příslušnost, kde je dané zařízení připojeno k internetu. Následující údaje jsou již závislé na konkrétním poskytovateli internetového připojení a mapování daných oblastí. V rámci České republiky lze velice dobře rozeznat velká města. Pro účel této práce je tento způsob identifikace pozice nedostačující.

3.3 Výpočet vzdálenosti mezi dvěma body

GPS modul je zařízení pracující s GPS signálem, který je přijímán z orbitálních družic. Ve všech aplikacích, využívajících informací o pozici dané zeměpisnými souřadnicemi, je zmiňován pojem vzdálenosti. Téměř vždy uživatele zajímá vzdálenost od pozice, kde se právě nachází, až do pozice cílového stanoviště.

Za cestou k výpočtu vzdálenosti stojí dlouhá teorie o zaoblenosti Země a o jejím skutečném tvaru. Historickým vývojem bylo pouze zjištěno, že Země není plochá, až k poznatku, že se jedná o útvar pojmenovaný geoid. Geoid je fyzikální model povrchu Země při střední hladině světových oceánů a zároveň prochází daným nulovým bodem. Tento útvar se matematicky velice špatně vyjadřuje. Velice často je nahrazován tzv. rotačním elipsoidem, který je dán dvěma základními parametry, a to délkou hlavní a vedlejší poloosy. Systém WGS-84, který je popsán v kapitole 3.1, používá právě geoid nahrazený rotačním elipsoidem. Při použití rotačního elipsoidu je zřejmé, že dochází k určitému zkreslení.

$$\text{distance} = \arccos[\sin(s_1) \cdot \sin(s_2) + \cos(s_1) \cdot \cos(s_2) \cdot \cos(d_1 - d_2)] \cdot d_z \quad (3.1)$$

Předpokladem pro správné použití vzorce je znalost hlavní poloosy neboli poloměru Země $d_z = 6\,378\,137$ m a zároveň co nejpřesnější hodnoty zeměpisné šířky a délky počátečního bodu (s_1 a d_1) a stejné informace o bodu cílovém (s_2 a d_2). Vzorec 3.1 je přejat z literatury [14], kde je popsán detailněji a je zde rozvedena myšlenka geodetického systému.

Kapitola 4

Implementace

Obsahem kapitoly je popis návrhu aplikace, včetně náčrtů vzhledu grafického uživatelského rozhraní. Popis samotného návrhu a implementace je rozdělen do více částí z důvodu přehlednosti a lepší orientace v dílčích částech aplikace.

Aplikace je členěna do menších logických částí, které samostatně vykonávají svou činnost a na ostatních částech aplikace nejsou závislé. Návrh je většinou doprovázen ilustračním obrázkem, který zobrazuje vizuální podobu řešeného problému.

Aplikaci bylo nejprve nutné důkladně navrhnout a poté implementovat, ale už i při plánování bylo nutné zvolit programovací jazyk a knihovnu pro návrh grafického uživatelského rozhraní. Bylo možné vybrat z několika možností. Dnes je většina knihoven pro grafické uživatelské rozhraní multiplatformní, což umožňuje podporu velkého množství programovacích jazyků.

Pro implementaci byl zvolen multiplatformní programovací jazyk C++ [11], který v současné době patří mezi nejrozšířenější programovací jazyky a zároveň umožňuje využít objektově orientovaný přístup k datům.

Výhoda tohoto přístupu k datům je například možnost abstrakce, zapouzdření a ukrytí dat, polymorfismus, dědičnost a opakovaná použitelnost kódu. Nespornou výhodou implementačního jazyka při použití standardních knihoven je přenositelnost téměř na všechny platformy bez dodatečné úpravy kódu.

Knihovna pro tvorbu grafického uživatelského rozhraní, byla zvolena Qt [1, 8]. Ta je velmi spjata s implementačním jazykem a zároveň umožňuje rychlý a pohodlný návrh vzhledu grafického uživatelského rozhraní. Dal jsem přednost knihovně Qt před GTK+ také proto, že obsahuje dokumentaci s příklady použití jednotlivých prvků. Pro podrobnější popis programovacího jazyka C++ lze využít literaturu [11]. Použití knihovny pro tvorbu grafického uživatelského rozhraní je popsáno v další části této kapitoly.

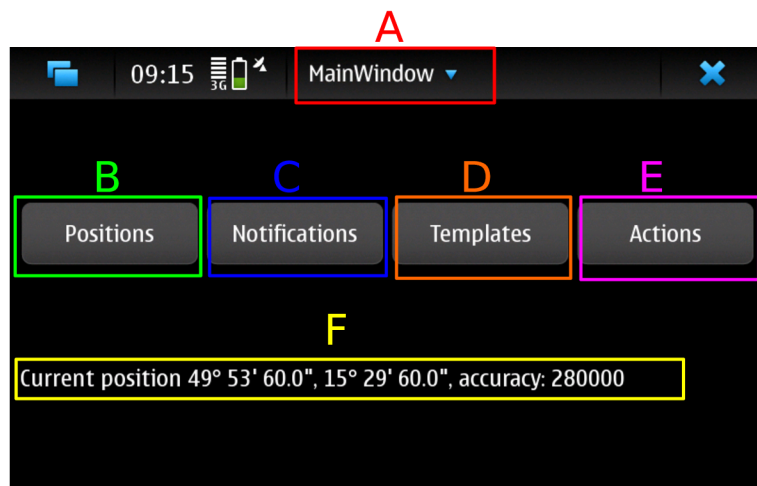
Na základě výběru grafické knihovny je mnohem lepší možnost vybrat vývojové prostředí Nokia SDK, které je již přizpůsobené a specializované na tvorbu softwaru pro mobilní zařízení. SDK navíc zahrnuje Mobility API umožňující jednodušší přístup k GPS modulu, který bude nutné využít pro zjištění pozice zařízení. Vývojové prostředí SDK bylo popsáno v kapitole 2.

Aplikace je navržena za pomoci objektově orientovaného přístupu a z tohoto důvodu byl do každé řešené části aplikace umístěn návrh UML diagramu tříd dané části aplikace. Diagram tříd zobrazuje strukturu a vztahy mezi objektovými třídami navrhované části aplikace. Součástí aplikace je práce s daty, kde problematika uložení dat a postup zpracování je vysvětlen v další části této kapitoly.

4.1 Grafická část aplikace

Hlavním úkolem grafické části aplikace je umožnit uživateli pracovat s daty a řídit aplikaci. Aplikace je tvořena několika částmi, které jsou jednotlivě popsány v této kapitole. Vytvoření grafické části usnadňuje práci s daty, která jsou téměř všechna uložena v databázi. Důvodem využití grafického uživatelského rozhraní je kontrola zadávaných dat a případné upozornění na chyby, které se mohou vyskytnout. Zároveň je rozhraní vytvořeno pro samotný účel aplikace tak, jak byla zadána.

Základní grafické uživatelské rozhraní, které lze vidět po spuštění aplikace, je hlavním rozcestníkem pro možnost výběru, s jakými prvky či tabulkami databáze bude nadále komunikováno. Návrh tohoto rozhraní je zobrazen na obrázku 4.1. Návrh obsahuje několik základních tlačítek, jež ovládají jednotlivé operace nad danou množinou dat. Oblast *A* umožňuje zobrazit menu pro nastavení aplikace běžící v pozadí. Písmeno *B* popisuje oblast pro tvorbu, úpravy a odstraňování pozic. Oblasti pro operace nad šablonami vyznačuje písmeno *D*. Operace nad upozorněními definuje oblast vyznačená písmenem *C*. Přidávání, úpravu a odstraňování akcí zajišťuje tlačítko, které označuje oblast *E*. Při spuštění aplikace je nezávisle na části běžící v pozadí zjištěna aktuální GPS pozice zařízení a je vypsána do oblasti *F*.



Obrázek 4.1: Návrh hlavní grafické části aplikace

Reakce na stisk tlačítek *B* – *E*, definovaných na obrázku 4.1, je totožná a spouští základní menu obsahující stejné typy operací pro všechny tabulky v databázi. Jedná se zejména o operace vkládání, úpravy a odstraňování záznamů. Vnitřní základní menu uchovává stav, který definuje danou tabulku na základě stisknutého tlačítka. Návrh, jak bude vypadat grafické rozhraní tohoto základního menu, je zobrazen na obrázku 4.2.

Obrázek obsahuje tři základní tlačítka definující jednotlivé operace. Oblast *A* označuje tlačítko pro přidávání, oblast *B* označuje úpravy a písmenem *C* je pojmenována oblast odstraňování. Zde není specifikován typ záznamu. Dané operace přidávání, úprava či odstraňování jsou definovány pro všechny typy záznamu, konkrétně tedy pro pozice, akce, šablony a upozornění.

Při práci s pozicemi uživatel vždy pracuje s grafickým uživatelským rozhraním, kde lze vyplnit zeměpisnou šířku a délku. Vyplnění zeměpisné šířky a délky je umožněno na základě

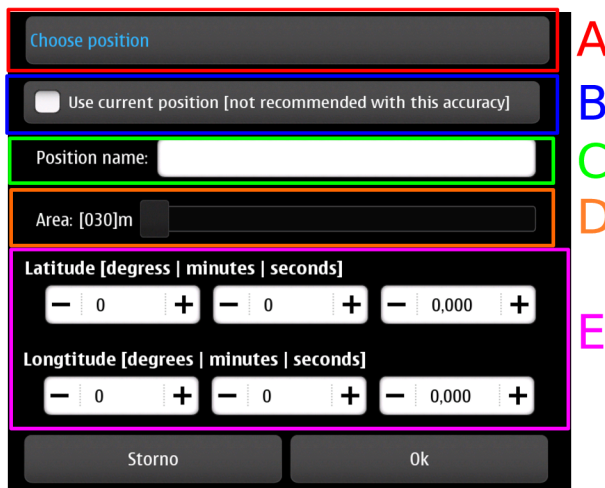


Obrázek 4.2: Návrh základního menu grafické části aplikace

tří nezávislých čísel, která reprezentují hodnoty známé z GPS zařízení - stupně, minuty, vteřiny. Navržená databázová struktura neakceptuje tři hodnoty pro určení zeměpisné šířky nebo délky. Pro uložení do databáze je nutné vstupní data od uživatele převést do jednoho reálného čísla. Pro načtení GPS souřadnice pro danou pozici z databáze, která může být vyvolána uživatelem při úpravách, je nutné převést hodnotu pozice, definované nyní reálným číslem, do formátu GPS. Ten je zobrazený pro uživatele v grafickém uživatelském rozhraní. Uživatelské rozhraní je totožné s rozhraním, při kterém bylo realizováno uložení.

Návrh vstupního dialogu, umožňující vkládání či úpravu GPS souřadnic, je zobrazen na obrázku 4.3. Vložení souřadnice dané pozice probíhá do oblasti *E*, která reprezentuje zeměpisnou šířku a délku.

Součástí uložení pozice do databáze je jméno, které je nutné vyplnit do oblasti označenou písmenem *C* a velikost tolerance oblasti zvýrazněné písmenem *D*. První část obrázku, zvýrazněna písmenem *A*, je viditelná pouze při úpravě GPS pozice a slouží pro výběr uložené pozice v databázi dle názvu. Poslední zvýrazněnou částí, písmenem *B*, je prvek identifikující doporučení, zda je vhodné uložit aktuální hodnotu GPS v závislosti na přesnosti určené pozice. Pokud je tento grafický prvek zaškrtnut, není viditelná oblast *E*, kde je provedeno uložení aktuálních údajů o zeměpisné šířce a délce automaticky.



Obrázek 4.3: Návrhu grafické části pro práci s pozicemi

Šablony shlukují jednotlivé akce do skupin, které lze použít pro vykonání reakce na danou pozici. Návrh rozhraní, přes které bude umožněno šablony přidávat a zároveň je upravovat, je zobrazen na obrázku 4.4.

Obsahem návrhu je rozdělení na několik oblastí, které jsou označeny písmeny v obrázku. Část označena písmenem *A* je viditelná pouze při úpravě šablon a slouží pro výběr

upravované šablony na základě jména. Písmeno *B* označuje oblast, kde je nutné zadat název šablony, který je povinný pro následnou identifikaci při tvorbě upozornění. Tlačítko, zvýrazněné na obrázku písmenem *C* slouží pro přidávání akcí do šablony, kde akce jsou vybírány dle názvu z uložených akcí v databázi. Poslední částí, označenou písmenem *D*, je oblast uložených akcí v dané šabloně. Uložené akce představují seznam akcí, které jsou seřazeny od shora dolů v pořadí, v jakém budou spouštěny při dosažení cílové pozice daného upozornění. Při tvorbě či úpravě šablony je nutné dodržet podmínku minimálního počtu akcí, který je stanoven na alespoň jednu akci. Při nedodržení této podmínky je vydáno upozornění a neprovede se žádná úprava nebo uložení do databáze.

Obrázek 4.4: Návrh grafické části pro práci se šablonami

Nedílnou součástí šablon jsou akce, které jsou shlukovány. Tvorba samotných akcí není součástí této práce. Uživatel dostává možnost do aplikace přiřadit programy třetích stran a použít je jako reakci na dosažení cílové oblasti. Uložení do databáze probíhá dle grafického uživatelského rozhraní, jehož návrh je zobrazen na obrázku 4.5. Obsahem návrhu je několik částí, které jsou zvýrazněny a označeny písmenem. První oblast, označená písmenem *A*, je viditelná pouze při úpravách akcí a slouží pro výběr upravované akce, dle jejího názvu. Oblasti, které je nutné vyplnit, jsou označeny písmeny *B* a *C*, kde *B* značí jednoznačný název akce, který musí být jedinečný v celé databázi a oblast *C* je cesta k programu, který bude představovat konkrétní činnost akce. Poslední označenou částí, oblast *D*, je volitelná část, která slouží pro uložení parametrů, s jejichž pomocí bude program spuštěn. Aplikace nerozlišuje, zda se jedná o program s grafickým uživatelským rozhraním či o program konzolového typu. Akce lze definovat jako uživatelem definované programy, včetně možnosti přidat parametry, které budou použity při spuštění.

Obrázek 4.5: Návrh grafické části pro práci s akcemi

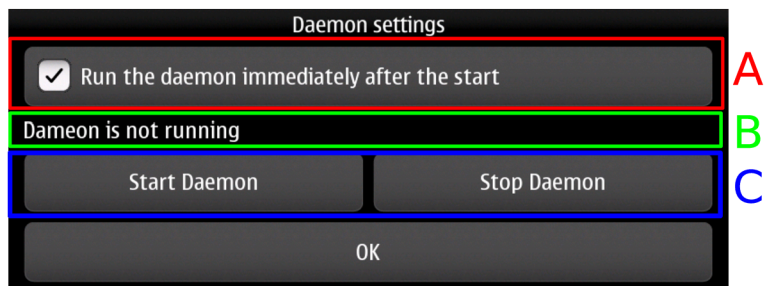
Poslední částí grafického uživatelského rozhraní, která pracuje s databází, je možnost přidávat, upravovat a odstraňovat upozornění. Upozornění je tvořeno na základě pozice a sady akcí neboli šablonou. Návrh vzhledu rozhraní je zobrazen na obrázku 4.6. Obrázek popisující návrh je rozdělen na více částí, které jsou označeny a zároveň jednoznačně identifikovány. Oblast *A* je viditelná pouze v případě úpravy upozornění, kde lze vybrat to upozornění, které bude podléhat úpravě na základě názvu. Písmeno *B* a jeho hraniční oblast vymezuje část, kam je nutno zadat povinný název upozornění. Následující dvě velice podobné oblasti jsou *C* a *D*, jež slouží pro výběr pozice, respektive šablony.

Na základě posledních jmenovaných informací je definováno upozornění. V případě odstranění záznamu pozice či šablony, které jsou použité pro upozornění, bude záznam s daným upozorněním odstraněn.



Obrázek 4.6: Návrh grafické části pro práci s upozorněními

Velice důležitou částí je nastavení chování aplikace běžící v pozadí, jejíž popis je dále v této kapitole. Dialog nastavení není spouštěn tlačítkem, ale je součástí menu, které je základním prvkem grafického uživatelského rozhraní.



Obrázek 4.7: Návrh grafické části pro nastavení aplikace běžící v pozadí

Nastavení je v uživatelském grafickém rozhraní intuitivní, přesto je jeho návrh zobrazen na obrázku 4.7. První část označená písmenem *A* a oblastí stejné barvy ukazuje, zda se aplikace běžící v pozadí má spouštět ihned po startu operačního systému Maemo. Následující dvě tlačítka jsou označena oblastí *C*. Ovládají aktuální činnost aplikace běžící v pozadí, jako je spuštění a vypnutí v reálném čase. Poslední oblast *B* označuje indikátor stavu aplikace běžící v pozadí.

Celkové spojení a komunikace jednotlivých tříd, jež byly vytvořeny pro grafickou část aplikace, lze vidět na UML diagramu tříd, zobrazeném na obrázku v příloze A. Jedná se

pouze o grafickou část aplikace, která je samostatná a oddělená od části aplikace běžící v pozadí, jejíž popis následuje v této kapitole.

4.2 Aplikace běžící v pozadí

Velice důležitá část aplikace je samostatně fungující konzolová aplikace, která běží samostatně v pozadí. Nevyrušuje a neodvádí pozornost uživatele grafickým uživatelským rozhraním. Konzolová aplikace je implicitně po instalaci nastavena na automatické spouštění ihned po startu operačního systému. Konfiguraci lze změnit v aplikaci s grafickým uživatelským rozhraním, kde lze rovněž ovládat spuštění nebo vypnutí konzolové aplikace.

Činnost aplikace lze rozdělit do tří částí. Získání pozice, komunikace s databází obsahující upozornění a vykonání upozornění při dosažení cílové pozice, které jsou na sobě nezávislé. Při běhu konzolové aplikace není omezena možnost úpravy databáze pomocí aplikace s grafickým uživatelským rozhraním. GPS souřadnice je dosahováno využitím GPS modulu, který je součástí mobilního zařízení. Při změně pozice je zároveň provedena kontrola tabulky databáze obsahující upozornění, zda na danou pozici, která je aktuálně získána, je naplánováno upozornění. Při kontrole, zda je naplánována akce na danou pozici, je zohledňována přesnost cílového bodu a zároveň oblast tolerance.

Informace o uložení GPS souřadnic, přesnosti pozic či oblasti tolerance jsou popsány dále v této kapitole. Při dosažení cílové pozice, s ohledem na přesnost a oblast tolerance, je vydáno upozornění, které bylo na danou pozici naplánováno. Většinou se jedná o sadu akcí, které jsou uloženy v šabloně akcí.

Šablona akcí určuje jaké akce a v jakém pořadí budou vykonány. Popis uložení dat v databázi, způsob řazení a určování akcí dle šablon, je popsán dále v této kapitole.

Je-li nastaveno upozornění na několik přilehlých pozic, které se svými oblastmi tolerance překrývají, jsou vyvolána upozornění na všechny pozice, kterých bylo tímto dosaženo. Vždy je nejprve vykonána sada upozornění na jednu pozici a až poté co je dokončena poslední akce aktuálně vyvolané sady upozornění, pokračuje se upozorněním na pozici další. Počet takto se překrývajících souřadnic není omezen.

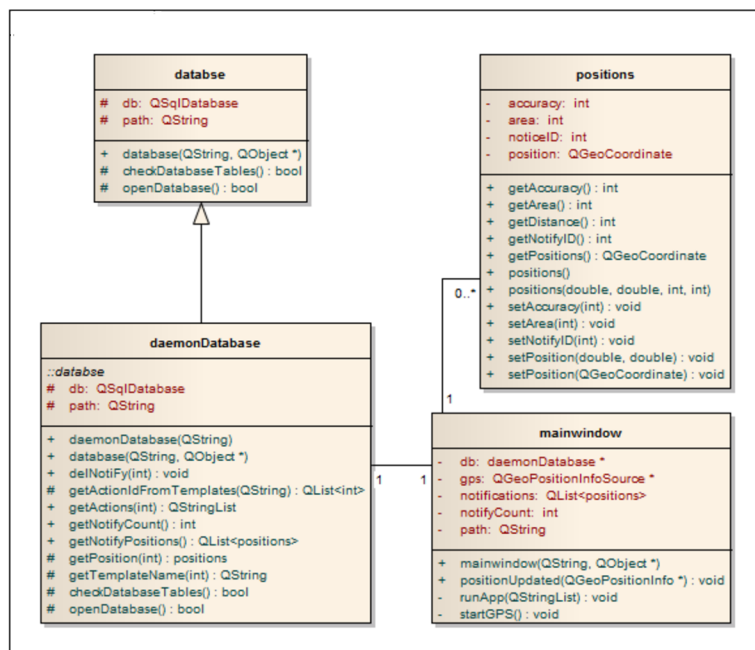
Pro úplnost a vysvětlení komunikace a spojení jednotlivých tříd navržených pro aplikaci běžící v pozadí, je vytvořen UML diagram tříd, který je zobrazen na obrázku 4.8.

4.3 Data

Aplikace pracuje s daty, která je nutno uchovat i po vypnutí aplikace, nebo případně po vypnutí celého mobilního zařízení. Nabízí se možnost uchovat data ve formátu XML nebo v databázovém systému SQLite [10]. XML soubor s daty lze upravit v jakémkoli textovém editoru. Možnost úpravy je někdy vítaná, ale součástí aplikace je grafické uživatelské rozhraní, sloužící pro úpravu dat, se kterými se pracuje.

Grafické uživatelské rozhraní plně nahrazuje manuální úpravu souborů a zároveň zabraňuje uživateli vnést do datové části chyby nebo nesmyslné údaje (například nevyplněný název pozice). Vzhledem k důvodu možnosti zanést chybu do XML souboru obsahující data, jsem zvolil pro ukládání dat z aplikace SQLite databázi.

SQLite je relační databázový systém, popisovaný knihovnou jazyka C, poskytující rozhraní SQL jazyka bez nutnosti konfigurace. Databáze podléhá téměř plnému znění standardu SQL-92. Komunikace probíhá na základě SQL dotazů, které jsou známé i u ostatních relačních databázových systémů, a jsou definovány ve standardu SQL-92. Databáze



Obrázek 4.8: UML diagram tříd aplikace běžící na pozadí

je uložena v samostatném souboru a není součástí žádného většího systému založeném na principu klient-server.

Výhodou tohoto přístupu je to, že není zapotřebí spouštět samostatný proces, který běží v pozadí a čeká na uživatelské dotazy požadující data v databázi. Pro implementaci databázi není vyžadována konfigurace nebo administrace systému. Použití dané databáze se doporučuje pro menší množství zpracovávaných dat nebo pro použití na mobilních zařízeních.

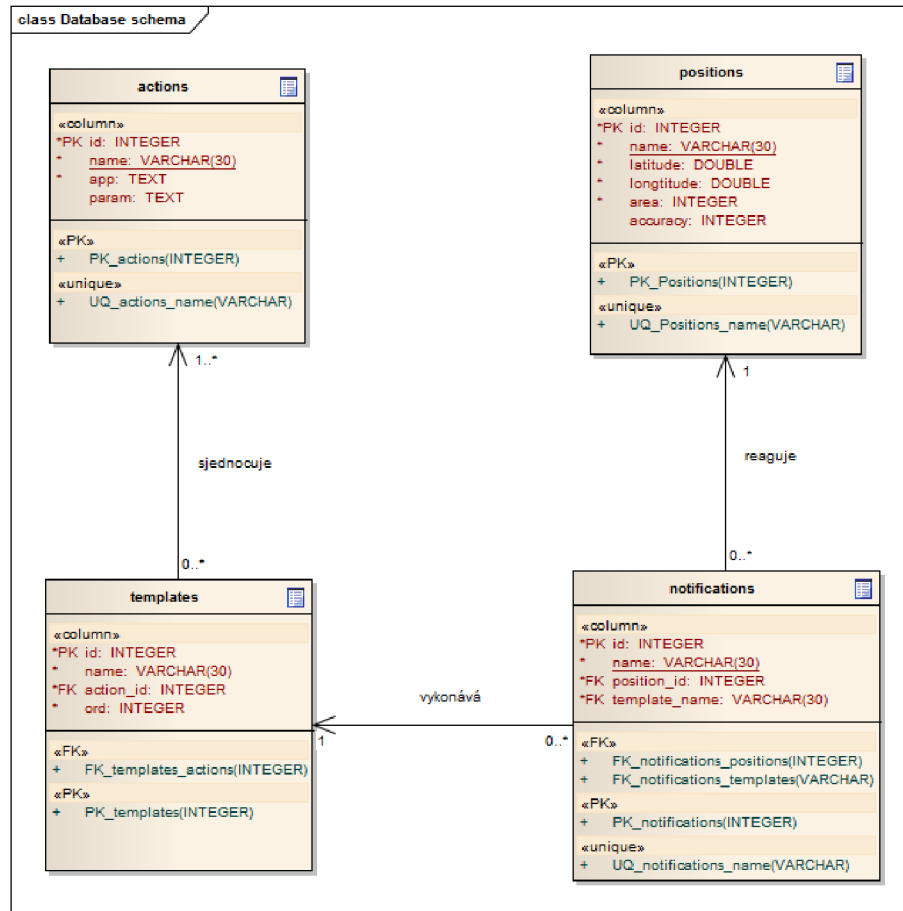
Funkčnost vybrané databáze je založena na procesu, který se spouští až v případě potřeby. To tedy znamená, že pokud se s databází nepracuje, neběží v systému žádný zbytečný proces. Pro většinu běžných operací nad databází je rychlejší, než je varianta systému klient-server.

Data jsou uložena v jednom jediném souboru, který je ovládán příkazy jazyka SQL, za pomoci daného relačního databázového systému. Data je pro přehlednost nutné rozdělit na několik samostatných částí (logických celků), kam se budou ukládat pouze data stejného zaměření. Význam jednotlivých datových typů a operací je definován normou SQL-92. Základní popis operací a datových typů vysvětluje literatura [10].

Konzistentní stav databázových dat je zajištěn dodatečnými úpravami záznamů při operacích odstranění či úpravě záznamu. Popis jednotlivých tabulek a jejich závislostí je uveden v každé podkapitole věnující se dané tabulce. Popis tabulek zároveň obsahuje vysvětlení některých dat, která na první pohled nejsou zřejmá, zejména vysvětlení proč jsou daná data ukládána.

Úplné znění tabulek v databázi a jejich propojení popisuje ER diagram neboli datový model jazyka UML, který je zobrazen formou obrázku 4.9. ER diagram zobrazuje jednotlivé tabulky databáze včetně detailů a realizace vztahů mezi tabulkami.

Následuje popis jednotlivých tabulek, kde je definován význam jednotlivých atributů. Součástí popisu tabulek je vysvětlení případného omezení pro daný typ atributu a závislostí



Obrázek 4.9: ER Diagram navržené databáze

mezi tabulkami. Závislost mezi tabulkami je využita zejména při žádosti o záznam z jiné tabulky. V případě úprav či odstranění záznamu, na který je odkazováno z jiné tabulky, by nastal stav nekonzistentní databáze.

Nekonzistentní stav databáze způsobuje nedefinované chování celé aplikace. Ochranou proti nekonzistentnímu stavu databáze je dodatečná kontrola všech tabulek databáze, kde lze očekávat vzájemnou závislost jednotlivých tabulek.

Pozice

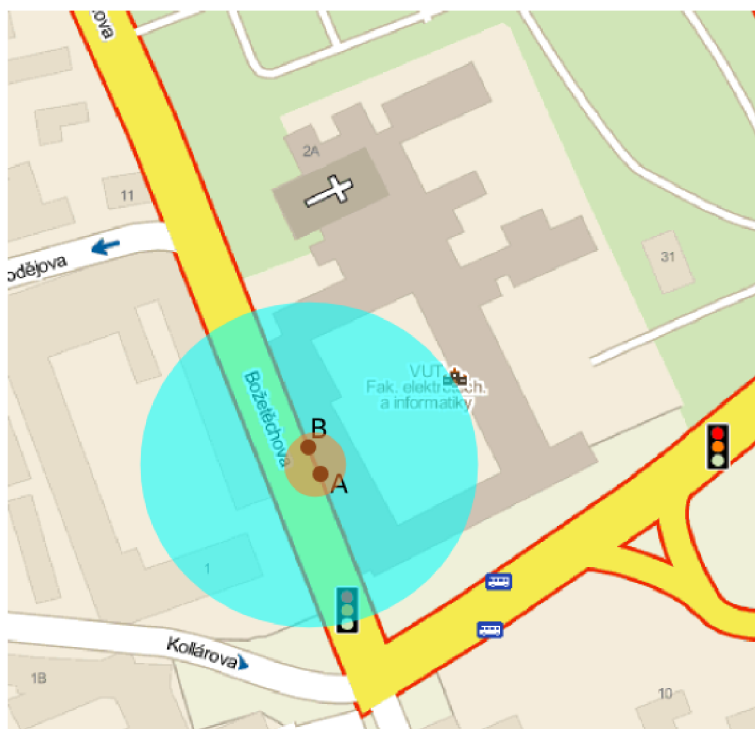
Tabulka **positions** obsahuje záznamy o pozicích, které jsou později využity pro tvorbu upozornění. Záznam pozice obsahuje základní identifikaci daného bodu. To jsou zeměpisná šířka a délka, a rozšířené informace specifické pro aplikaci. Specifické informace souvisejí s polohou, ale nejsou nutné pro zjištění pozice, nýbrž její zjištění ulehčují a dovolují určitou toleranci.

Při ukládání pozice, na níž se mobilní zařízení aktuálně nachází, je uživatel informován, zda je vhodné použít aktuální pozici v závislosti na přesném určení dané pozice. Informace o přesnosti aktuální pozice je uložena do databáze společně s polohou a tvoří okolo přesné pozice oblast, u níž lze tolerovat, že se jedná o stejný bod. Pokud je tento krok vynechán nelze očekávat, že vždy na stejném místě bude vyvoláno upozornění, pokud bude

nastaveno pro tuto pozici. Důvodem je nepřesnost zařízení, jež je způsobena použitím dané technologie, nebo polohou, kde je špatný signál z orbitálních družic. Při ukládání pozice je možné, že nastane stav, kdy ukládaná souřadnice, jež je zobrazena v grafickém uživatelském rozhraní, není přesnou pozicí daného bodu. Při uložení aktuální pozice včetně přesnosti lze předpokládat, že kruhová oblast s poloměrem přesnosti dané pozice bude obsahovat přesnou souřadnici bodu, na níž ji mobilní zařízení ukládá.

Dalším údajem pro rychlejší určení pozice je velikost oblasti, která značí kruhovou oblast od přesné souřadnice bodu. Za tu lze považovat dosažení přesné pozice. Jedná se o stejný údaj jako je přesnost pozice s tím, že tento údaj uživatel ovlivňuje sám podle vlastního uvážení. Tento údaj je velice často využíván pro předčasné upozornění na danou pozici před jejím dosažením.

Pojmy tolerance oblasti a přesnost pozice jsou popsány obrázkem 4.10, kde jsou tyto pojmy zobrazeny na reálném příkladu. Přesná souřadnice je situována na vchod fakulty informačních technologií v Brně, identifikována bodem *A*, kde přesnost dané pozice byla 11 m a mobilní zařízení ukazuje pozici mírně posunutou - bod *B*. Na obrázku lze vidět kruhovou oblast v oranžové barvě, která definuje přesnost dané pozice a modrou kruhovou oblast definující toleranci oblasti o velikosti 50 m.



Obrázek 4.10: Popis informací při získávání pozice

Pozice je jednoznačně identifikována zeměpisnou šířkou a délkou, jež jsou uloženy v databázi jako reálná čísla. Záznam, obsahující pozici, je nutné identifikovat jednoznačným jménem, tak poslouží pro rychlou identifikaci v grafickém uživatelském rozhraní.

Vnitřně je záznam pozice v databázi definován jednoznačným identifikátorem, který není uživateli zpřístupněn. Je využit pro získání konkrétní pozice z jiné tabulky.

Identifikační číslo záznamu bylo doplněno pro rychlejší a efektivnější práci se záznamy v databázi. Je možné určit jméno pozice jednoznačným identifikátorem, ale porovnávání

řetězců je pomalejší a méně efektivní, než je tomu u čísel. Lepší volbou je vytvořit primární klíč z číselného záznamu.

Pozice je využívána k tvorbě upozornění. Upozornění je realizováno na základě údajů o pozici. Pokud je vytvořeno upozornění na danou pozici (záznam v databázi) a následně je pozice upravována či odstraněna, dochází k nekonzistentnímu stavu databáze. Upozornění je vytvořeno pro pozici, která vlastní jiné identifikační číslo nebo pozice není v databázi uložena. Pro odstranění tohoto nechtěného jevu, je při úpravách či odstraňování záznamů s pozicemi, spuštěna kontrola závislosti. V případě, že je nalezeno upozornění na pozici, která byla upravena, je v tabulce `notifiactions` upravena identifikace dané pozice na správnou hodnotu identifikující záznam s pozicí. Pokud je záznam s pozicí odstraněn, je odstraněno i upozornění na tuto pozici, které by v tomto případě nebylo nikdy vyvoláno.

Akce

Pro potřeby reakce při upozornění na určitou pozici je nutné mít možnost přidávat, upravovat, odstraňovat a celkově regulovat samotné akce pro tvorbu jednotlivých upozornění. Upozornění jsou ukládána do databáze jako všechna data, s využitím grafického uživatelského rozhraní. Akce je definována jako určitá činnost, která je vykonána na žádost, v tomto případě je žádostí upozornění. Musí být definován program, který provádí požadovanou činnost. Součástí aplikace jsou dva programy, které jsou popsány v kapitole 4.4 a jsou implicitně uloženy v databázi pro využití jejich činnosti jako akce pro upozornění.

Aplikace umožňuje použít software třetích stran, který není součástí aplikace. Možnost přidávat aplikace třetích stran je nesporná výhoda, která umožňuje individuální přizpůsobení upozornění každému uživateli dle jeho uvážení a softwarové výbavy.

Název akce je povinným údajem zlepšující přehled a možnosti orientace v seznamu akcí v grafickém uživatelském rozhraní. Název musí být jedinečný v celé databázi, a je pro uživatele jednoznačným identifikátorem.

Následující povinný údaj je program, který definuje činnost akce, přesněji řečeno cesta k programu. Aplikace nerozlišuje, zda se jedná o program běžící v konzolovém prostředí nebo program s grafickým uživatelským rozhráním.

Některé aplikace, většinou konzolového typu, vyžadují zadání parametrů při spuštění, na jejichž základě je ovlivněno chování celé aplikace. Pro tento případ je vytvořen poslední nepovinný údaj, jež slouží pro účely parametrů. V případě, že aplikace vyžaduje více parametrů, lze je uvést s použitím běžného oddělovače (mezera).

Vnitřně jsou záznamy jednotlivých akcí jednoznačně definovány číselným identifikátorem, který je využit při odkazování si na konkrétní akci z jiné tabulky a před uživatelem je skryt.

Tabulka `actions` patří mezi tabulky, které jsou závislé na jiné tabulce. V tomto případě se jedná o tabulku `templates` (tabulku šablon). Tabulka šablon jež shlukuje jednotlivé akce a řadící je do seznamu, ze kterého jsou následně spouštěny při vyvolání upozornění, je popsána v následující části této kapitoly.

Problém nekonzistentního stavu databáze nastává při úpravě či odstranění záznamu akce, na který se odkazuje tato tabulka. Konzistentní stav databáze je zachován tak, že při úpravě je provedena kontrola tabulky šablon, zda obsahuje akci, která byla upravena. Pokud je nalezena akce, která byla upravena, je zaměněno identifikační číslo, které již není klasifikováno jako správné. Pokud byl záznam dané akce odstraněn a v tabulce šablon je použit, je z tabulky šablon vymazán.

Kontrola akcí je již dokončena, ale nastává problém s tabulkou šablon, která musí vždy

obsahovat alespoň jednu akci. Pokud je ze šablony akcí odstraňována akce a tudíž je šablona prázdná, je šablona zrušena. Reakcí na odstranění šablony je hledání závislosti v tabulce **notifications**. Řešení závislosti mezi tabulkami **notifications** a **templates** je popsána v následující části této kapitoly.

Šablony

Pod pojmem šablona lze očekávat strukturu obsahující seznam akcí. Akce, které byly popsány v předchozí části, jsou uspořádávány či shlukovány do šablon. Tyto šablony jsou řazeny za sebe tak, jak budou následně spouštěny v případě vykonání upozornění. Šablony představují tabulku v databázi pojmenovanou **templates**. Uspořádání, úpravu, přidávání a odstraňování jednotlivých akcí ze šablony lze provádět pouze v grafickém uživatelském rozhraní.

Tabulka **templates** vyžaduje od uživatele pojmenování, které musí být jedinečné v celé databázi a slouží pro jednoznačné určení sady akcí při tvorbě upozornění. Počet akcí, které jednotlivé šablony mohou obsahovat, není shora omezen. Je ale nutné dodržet podmínku přítomnosti alespoň jedné akce v každé šabloně. Omezení minimálního počtu akcí je kontrolováno v grafickém uživatelském rozhraní. Uživatel je na tuto skutečnost upozorněn při vytváření nové šablony nebo při ukládání šablony, která neobsahuje žádnou akci. Součástí udržování seznamu akcí je jejich samotné řazení do pořadí, ve kterém budou vykonávány při dosažení cílové oblasti.

Řazení akcí je intuitivním způsobem zpracováno v grafickém uživatelském rozhraní. Zde je zobrazen seznam akcí, jimiž šablona disponuje. Seznam je zobrazen v pořadí, ve jakém budou jednotlivé akce vykonávány. Je situován vertikálním směrem, kde nejvyšší akce je vykonána jako první a nejnižší jako poslední. Akce lze mezi sebou zaměňovat, odstraňovat a zároveň přidávat akce nové.

Uživatel jednoznačně identifikuje šablonu podle názvu, který je pro danou šablonu jednoznačný. Vnitřně v databázi je šablona rozdělena na několik záznamů, kde jednotlivé záznamy mají svůj jednoznačný identifikátor ve formě číselného údaje. Údaj, který spojuje záznamy do jedné šablony je právě název, který zadal uživatel v grafickém uživatelském rozhraní. Součástí jednotlivých záznamů je pořadí jednotlivých akcí a samotný odkaz na akci. Odkaz na akci je realizován jednoznačným identifikátorem akce z tabulky **actions**.

Upozornění

V textu byla již naznačena určitá závislost mezi tabulkami **templates** a **actions**. Závislost je tvořena již u tabulky **actions**, která byla popsána v předchozí části textu. Zároveň byl představen způsob chování v případě odstranění akce, která je součástí šablony. Zde je závislost ještě rozšířena na tabulku **notifications**. Při vytváření upozornění jsou povinné dva základní údaje: pozice, na kterou se bude reagovat a samotná šablona akcí obsahující činnost vykonávanou v případě dosažení cílové oblasti. Pokud při úpravě šablony dochází ke změně názvu je nutné pozměnit tento název i v tabulce **notifications**, která se podle názvu odvolává na šablonu akcí.

Lze provádět úpravy v šabloně, které nemění název. V tomto případě není potřeba kontrolovat upozornění. Při úplném odstranění šablony (nikoli akce z šablony) jsou následně odstraněna všechna upozornění, kde se vyskytuje odstraňovaná šablona.

Databáze, nacházející se v nekonzistentním stavu, by způsobila neočekávané chování aplikace. Toto chování by se projevilo při vynechání kontroly při úpravách, které mají vliv a zasahují do jiných tabulek.

4.4 Ostatní části

Součástí aplikace jsou podpůrné programy. Tyto podpůrné programy slouží k propojení s aplikací. Jedná se o programy `vibration` a `sound`. Programy vykonávají svou činnost v závislosti na svých parametrech, s jejichž pomocí jsou spouštěny. Přítomnost zmíněných programů slouží pro připojení k aplikaci jako akce, které reagují na upozornění. Nelze se spolehnout, že každý uživatel mobilního zařízení Nokia N900 disponuje stejným software a z tohoto důvodu byly vytvořeny dané programy.

Samotná činnost programů vyplývá z jejich názvů, ale je i přesto upřesněna alespoň základní činností. Program `vibration` spouští vibrace na mobilním zařízení a v závislosti na parametrech je dán typ vibrace a její délka. Druhým programem je `sound`, který umožňuje přehrávat hudební soubory v různých formátech, slouží tedy jako hlasové upozornění při dosažení cílové oblasti.

Při instalaci jsou programy automaticky vloženy do databáze, která je také při instalaci vytvořena. Následně jsou vytvořeny ukázkové šablony obsahující dané akce. V případě přechodu na vyšší verzi již není v plánu upravovat databázi, kde je hlavním důvodem ztráta uživatelem definovaných dat a vztahů mezi nimi.

Poslední částí této práce je vytvoření balíčku, který slouží k distribuci aplikace. Balíček byl vytvořen podle pravidel systému Linux a zejména distribuce Debian. Pravidla pro tvorbu balíčků podléhají přísným pravidlům, jež je nutné dodržet jinak není balíček přijat operačním systémem k instalaci. Součástí balíčku bylo vytvoření skriptu pro ovládní aplikace běžící v pozadí. Skript umožňuje spouštět aplikaci běžící v pozadí ihned po startu operačního systému neboli po zapnutí mobilního zařízení.

Kapitola 5

Dosažené výsledky

Tato kapitola se zabývá dosaženými výsledky, kde v první části je hodnocena přesnost získávané pozice. Zkoumání horizontální přesnosti bylo prováděno pro možnosti lepšího nastavení pevně daných konstant používaných při ukládání pozic. Samotné zjišťování přesnosti probíhalo v různých podmínkách.

Další část kapitoly je věnována zkoumání použitelnosti uživatelského rozhraní, které je hodnoceno na základě odezvy komunity uživatelů. Na samotný závěr kapitoly jsou shrnuty možnosti dalšího rozšíření aplikace. Návrhy rozšíření jsou z velké části odezvou vývojářů, kteří tuto aplikaci zkoušejí.

Aplikace byla vytvořena a zkompletována do balíčku. Balíček byl odeslán do testovací laboratoře společnosti Nokia. Pokud testování proběhne v pořádku, pak aplikace bude možno stáhnout z oficiálního extra repozitáře operačního systému Maemo.

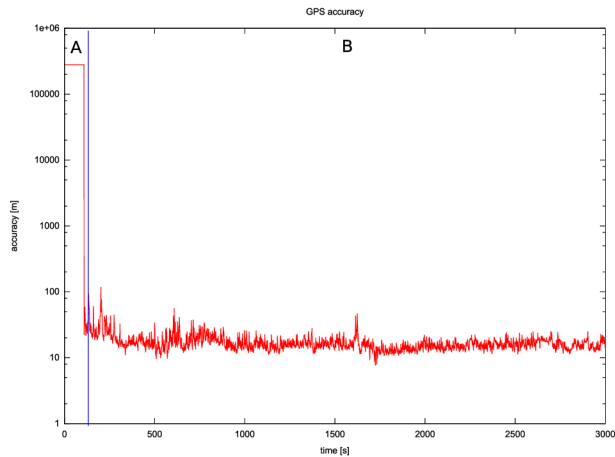
Bohužel v rámci časové náročnosti testování jednotlivých aplikací, které jsou do jmenované společnosti odesílány ve velkém množství, nelze zaručit dostupnost balíčku s aplikací GEOReminder v době psaní této práce. Obvykle přesun z testovací laboratoře do prvního veřejného repozitáře, který není doporučen pro běžné uživatele, zabere zhruba měsíc. Dalším krokem je přesun do klasického repozitáře doporučeného pro všechny uživatele. Přesun mezi samotnými repozitáři již netrvá takovou dobu, jako je tomu mezi testovacím centrem a prvním repozitářem.

5.1 Přesnost získání pozice

Při práci s GPS modulem daného zařízení se vyskytlo několik známých problémů s přesností systému získání pozice. Při ukládání aktuální pozice lze definovat tzv. oblast tolerance, která je popsána v kapitole 4. Zkoumání horizontální přesnosti by mělo upřesnit, jak danou oblast tolerance nastavit.

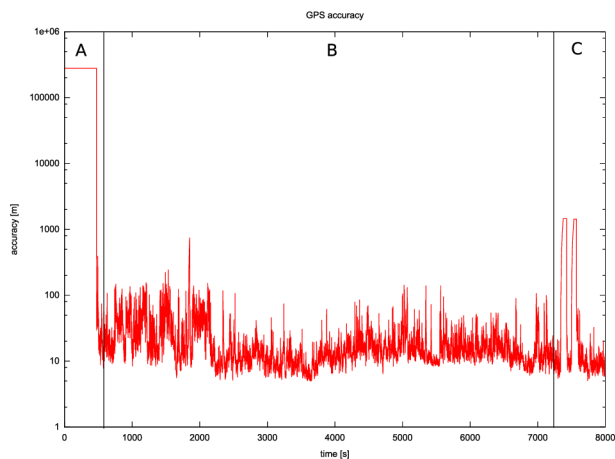
Bylo vytvořeno několik záznamů horizontální přesnosti systému GPS. Záznam probíhal na zařízení Nokia N900. Obrázek 5.1 popisuje záznam horizontální pozice v jediném statickém bodě. Místo pro vykonání měření bylo v zastavěné a obydlené oblasti. Na obrázku lze identifikovat část, kde zařízení nemá zjištěnou pozici, která je označena oblastí *A*. Po získání pozice je už jen měněna horizontální přesnost, tak jak ukazuje oblast *B*, která je závislá na síle signálu a počtu dostupných orbitálních družic.

Další měření, zobrazené na obrázku 5.2, probíhalo za jízdy automobilem po dálnici D1, kde lze opět identifikovat oblast, označenou písmenem *A*, kde zařízení nemá k dispozici aktuální pozici. Na obrázku lze vidět mnohem větší nepřesnost aktuální GPS pozice, která



Obrázek 5.1: Horizontální přesnost GPS systému u statického bodu

je velice nestálá a mění se v závislosti na dostupnosti signálu z orbitálních stanic tak, jak je na obrázku vyznačeno písmenem *B*. Oblast označená *C* je naměřena v zalesněné části dálnice D1, kde lze vidět nárůst horizontální nepřesnosti.

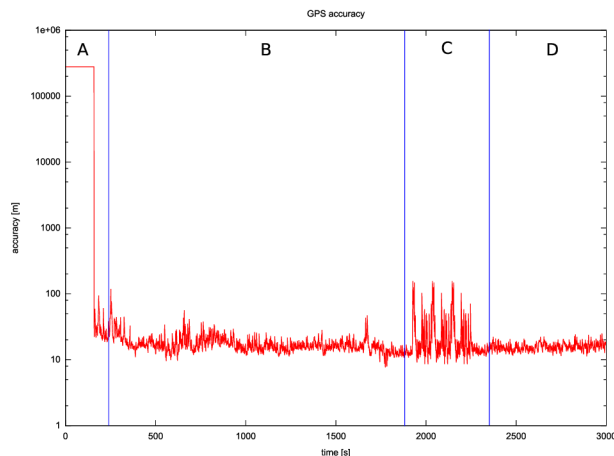


Obrázek 5.2: Horizontální přesnost GPS systému při jízdě automobilem

Poslední měření probíhalo při chůzi po Praze. Grafické zobrazení je vidět na obrázku 5.3, kde lze jako u předešlých obrázků identifikovat oblast, označenou *A*, kdy zařízení je bez aktuální pozice. Oblast *B* je zaznamenala získanou polohou v méně zastavené části Prahy. Při dosažení obydlené oblasti lze pozorovat zhoršení přesnosti tak, jak je označeno na obrázku písmenem *C*. Opětovné získání přesnosti je patrné v oblasti *D*, kde zařízení bylo na velkém náměstí a tedy přístup k signálu z orbitálních stanic byl silnější.

Výsledky měření přesnosti je vhodné zohlednit při tvorbě upozornění, jak lze nastavit oblast tolerance, která by neměla být vždy minimální. Měření přesnosti probíhalo v odlišných podmínkách, ale výsledky jsou očekávané. Je možno konstatovat, že horizontální přesnost je závislá na dostupnosti a síle signálu z orbitálních stanic a zároveň pohyb mobilního zařízení.

Při zadávání oblasti tolerance by měl být zohledněn způsob přibližování se k cílové ob-



Obrázek 5.3: Horizontální přesnost GPS systému při chůzi

lasti, kdy je nutné zahrnout různě velkou horizontální přesnost aktuální pozice, v závislosti na rychlosti pohybujícího se zařízení.

5.2 Uživatelské rozhraní

Shrnutí uživatelského rozhraní je vhodné konzultovat nebo také diskutovat s uživateli stejného mobilního zařízení, kdy lze získat odezvu potřeb a předností jednotlivých uživatelů. Po shrnutí je možné vytvořit představu, zda je vhodné vyvíjet aplikace pro mobilní platformu na základě jednoduchého rozhraní, bez dodaného návodu na použití, či vytvořit komplexní rozhraní a dodat patřičný manuál.

Aplikace byla po vytvoření instalačního balíčku umístěna na různé internetové stránky zabývající se vývojem softwaru na mobilní zařízení. Zpětná vazba byla zaznamenána v menší míře tak, jak bylo očekáváno. Důvodem je počet uživatelů pracujících s mobilním zařízením Nokia N900.

I přes nižší účast uživatelů je patrné, že velice přínosná na aplikaci je možnost individuálně tvořit upozornění a šablony akcí, kde není předem zadáno omezení na konkrétní akce. Možnost zařadit aplikace třetích stran má největší kladnou odezvu. Odezvu jsem, při psaní této práce, obdržel od pěti uživatelů.

Kladné hodnocení přináší navržené grafické rozhraní aplikace, kde je zachována jednoduchost. Uživatelé oceňují právě jednoduchost a intuitivnost samotného ovládání. Zejména jednoduché použití tlačítek, provádí cílenou funkci.

Grafické uživatelské rozhraní je hodnoceno velice dobře i když se vyskytla výtka k menu, které obsahuje nastavení aplikace běžící v pozadí. Není nijak zvláště zvýrazněno či označeno, že se jedná o menu. Tento problém je známý již od prvotního návrhu grafického rozhraní samotného operačního systému Maemo. Nabízí se možnost v příštím vývoji a případných úpravách odstranit menu a nahradit jej klasickým tlačítkem tak, jak jsou spouštěny ostatní události aplikace.

Aplikace byla spouštěna na mobilním zařízení, ke kterému byl vývoj směřován. Po zveřejnění projektu na internet, byla odezva od uživatelů, jež aplikaci nainstalovali na stejné mobilní zařízení, kladná. Všechny funkce aplikace pracovaly správně. Někteří uživatelé požadovali zdrojové kódy samotné aplikace, které by použili po úpravě na jiném

mobilním telefonu.

Po shrnutí výsledků hodnocení od pěti uživatelů mohu konstatovat, že většina uživatelů upřednostňuje zkoumání uživatelského rozhraní bez dodaného manuálu. Podle většinového kladného hodnocení je možné usuzovat, že aplikace má dobře navržené grafické rozhraní, ve kterém se lze po krátké době orientovat. Při diskuzi, zda by nebylo vhodnější rozhraní udělat jako komplexní a nabídnout více možností nastavení, jsem dostal ve většině případů záporné odpovědi.

Současně jsem se zpětnou vazbou dostal návrhy na další rozšíření, jež jsou popsána dále v této kapitole.

5.3 Možné rozšíření

Vývoj aplikace pro účel této práce byl ukončen, ale nabízí se možnost pokračování. Aplikace je navržena pro určitou modularitu. Do aplikace se může vložit mnoho drobností, které ji vylepší. Protože zpětná vazba není tak frekventovaná a její obsah je většinou stejný, je nápadů na rozšíření méně. Přesto jsou zde velice zajímavé návrhy, jež by své uplatnění našly.

Velice zajímavým návrhem pro vývoj této aplikace je možnost tvorby upozornění s opakováním. Upozornění by bylo vykonáno při dosažení cílové pozice a při dostatečné vzdálenosti od této pozice by bylo opět automaticky naplánováno. Při tomto rozšíření aplikace je zde další možnost vylepšení. Možnost uložit počet těchto opakovaných automatických upozornění, kdy by se při každém vykonání upozornění snížil tento počet až k nule a upozornění by se již neplánovala.

Další modifikace automatického opakování je možnost opakovat upozornění v určitém časovém úseku, ať už v rámci minut, hodin, dnů či měsíců. Princip je velice podobný předešlému nápadu s tím, že by se neodečítal počet upozornění, ale probíhala by kontrola kalendářního data, zda se má naplánovat další upozornění.

Kontrola data by se měla provést i před samotným vykonáním upozornění. Důvodem je dosažení cílové pozice až po překročení daného časového intervalu platnosti. V tomto případě by mělo být automaticky upozornění odstraněno bez vykonání jakékoli činnosti.

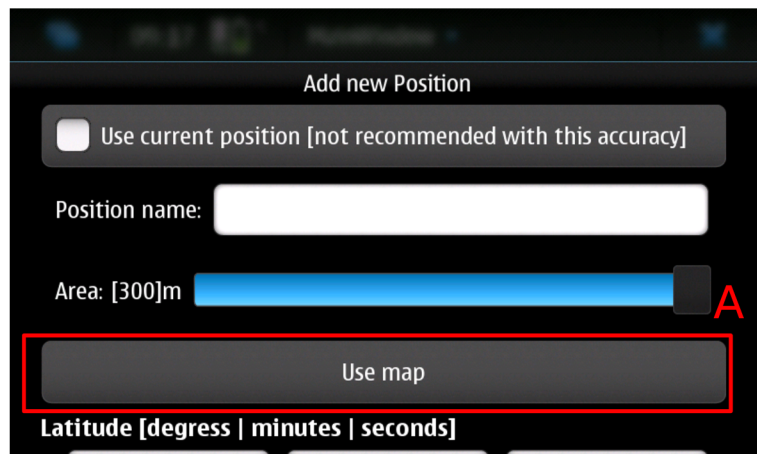
Při ukládání aktuální pozice mobilního zařízení v grafickém uživatelském rozhraní není možnost tuto pozici zobrazit na mapě. Rozšířením ukládání pozice o zobrazení na mapě, na níž by bylo umožněno opravit aktuální pozici, se nabízí jako další možnost úpravy aplikace. Realizace tohoto rozšíření by byla možná na základě přidání tlačítka do dialogu, který umožňuje manipulovat s pozicemi.

Dialog, umožňující přidávání a úpravu pozice, je ukázán na obrázku 5.4, který je rozšířen o tlačítko zobrazující mapu, na obrázku označeno písmenem *A*.

Návrh, jak by mohlo vypadat zobrazení mapy, je ukázán na obrázku 5.5. Bodem *A* je označena pozice, kterou mobilní zařízení považuje za aktuální. Uživateli je umožněno, s využitím dotykového displeje, nalezenou pozici (bod *A*) přemístit na aktuální pozici, která odpovídá skutečné poloze mobilního zařízení.

Aktuálně je aplikace navržena pro získání pozice na základě více druhů přijímání pozice a to v závislosti na dostupnosti jednotlivých služeb. Není doposud implementována možnost propojit jednotlivá zařízení mezi sebou, kde by bylo možné upozornění, že se v určité blízkosti objevilo stejné mobilní zařízení. Nabízí se možnost importovat a exportovat uložené pozice, šablony, akce nebo celá upozornění mezi sebou.

V aplikaci není implementována schopnost reagovat na různé podněty, jako je například pozice mobilního zařízení. To může být v poloze displejem dolů nebo nahoru, anténou



Obrázek 5.4: Návrh dialogu pro manipulaci s pozicemi



Obrázek 5.5: Návrhu dialogu s mapou pro korekci pozice

nahoru nebo dolů nebo postavené na jedné z bočnic. Implementace pro určité pozice by do budoucna mohla zajišťovat zamezení vykonání reakce v dané pozici.

Příkladem může být důležité jednání na lodi, kdy není vhodné vykonávat upozornění o případných oblastech, jež by mobilní zařízení dosáhlo. V daném případě by bylo vhodné mobilní zařízení otočit displejem dolů a upozornění by bylo ignorováno.

Dalších možností využití aplikace lze dosáhnout s využitím jiných aplikací a jejich vzájemným propojením. Příkladem může být prozvonění či automatické zavolání někomu ze seznamu kontaktů v případě dosažení cílové oblasti. Častější a praktičtější použitím se možná jeví zaslání krátké sms zprávy či zprávy některému klientovi využívajícího síťovou komunikaci, například Jabber nebo ICQ.

V případě používání aplikace v dosahu bezdrátové sítě Wi-Fi, lze využít internet k získání aktuální teploty v dané oblasti a vykonat předem nastavené upozornění. Upozornění by bylo vhodné směřovat na dobu, kdy jde uživatel do školy či do práce, aby věděl, jak se má obléci. Další způsob upozornění by nebyl podle času, ale právě podle teploty, kdy by bylo vykonáno upozornění po dosažení dané teploty.

Nasazení aplikace do reálného života bude možné pro běžné uživatele ihned po zařazení do repozitáře společnosti Nokia. Aplikaci lze využít v mnoha oborech, kde lze specifikovat aplikaci tvorbou samotných akcí, které na ní nejsou závislé.

S využitím textového editoru lze aplikaci přizpůsobit pro nákupní seznam, kdy upozornění bude spuštěno při dosažení oblasti s nákupním centrem.

Další možností přizpůsobení je například na docházku v práci. Změna profilu mobilního telefonu automaticky při příchodu do školy nebo do práce a automatické navrácení původního profilu při odchodu.

Kapitola 6

Závěr

Cílem této práce bylo vytvořit aplikaci pro mobilní zařízení Nokia N900, která by umožňovala spouštění a tvorbu různých upozornění v závislosti na pozici. Tato pozice je ukládána manuálně vyplněním zeměpisné šířky a délky nebo automaticky na základě aktuální pozice GPS zařízení. Pro větší variabilitu lze jako upozornění zvolit spuštění libovolné aplikace nebo několika programů. Příkladem takové aplikace může být například přehrávání zvuku, vibrace telefonu nebo textové oznámení.

Návrh aplikace by nemohl proběhnout bez získání znalostí o vývoji aplikací pro mobilní platformu a specifické hardwarové a softwarové prostředky mobilního zařízení Nokia N900, které je popsáno v kapitole 2. Současně bylo nutné nastudovat z doporučené literatury problematiku získávání pozice, funkčnost systému získávání pozice. Podrobnosti k tomuto tématu byly popsány v kapitole 3.

Implementace zahrnovala návrh grafického uživatelského rozhraní, za pomoci Nokia Qt SDK s využitím Qt Mobility API, a tvorbu aplikace běžící v pozadí. Obě části byly navrženy a úspěšně implementovány tak, jak je popsáno v kapitole 4. Tato kapitola obsahuje zároveň popis jednotlivých datových struktur, které jsou v aplikaci využity.

Aplikace byla vytvořena a zkompletována do balíčku. Balíček byl odeslán do testovací laboratoře společnosti Nokia. Vytvořená aplikace je přiložena na externím mediu – CD, které je součástí této práce. Pokud testování proběhne v pořádku, pak aplikaci bude možné stáhnout z oficiálního extra repozitáře operačního systému Maemo.

Na základě odezvy uživatelů po zveřejnění této aplikace bylo vytvořeno určité shrnutí zpětné vazby, které zároveň přineslo impulsy pro další pokračování vývoje aplikace. S uživateli bylo diskutováno řešení a případná vylepšení aplikace, kdy některá jsou přímo součástí aplikace a některá jsou popsána jako možnosti dalšího vývoje. Podrobnější informace lze nalézt v kapitole 5, která popisuje dosažené výsledky této práce.

Součástí tvorby aplikace bylo vytvoření krátkého demonstračního videa, sloužícího pro základní pochopení ovládání aplikace přes grafické uživatelské rozhraní. Na videu lze shlédnout průběh uložení aktuální pozice mobilního zařízení, tvorbu upozornění na danou pozici a samotnou reakci na dosažení oblasti tolerance uložené pozice. Video je uloženo na nosiči CD viz příloha B.

Všechny kladené cíle se podařilo splnit.

Literatura

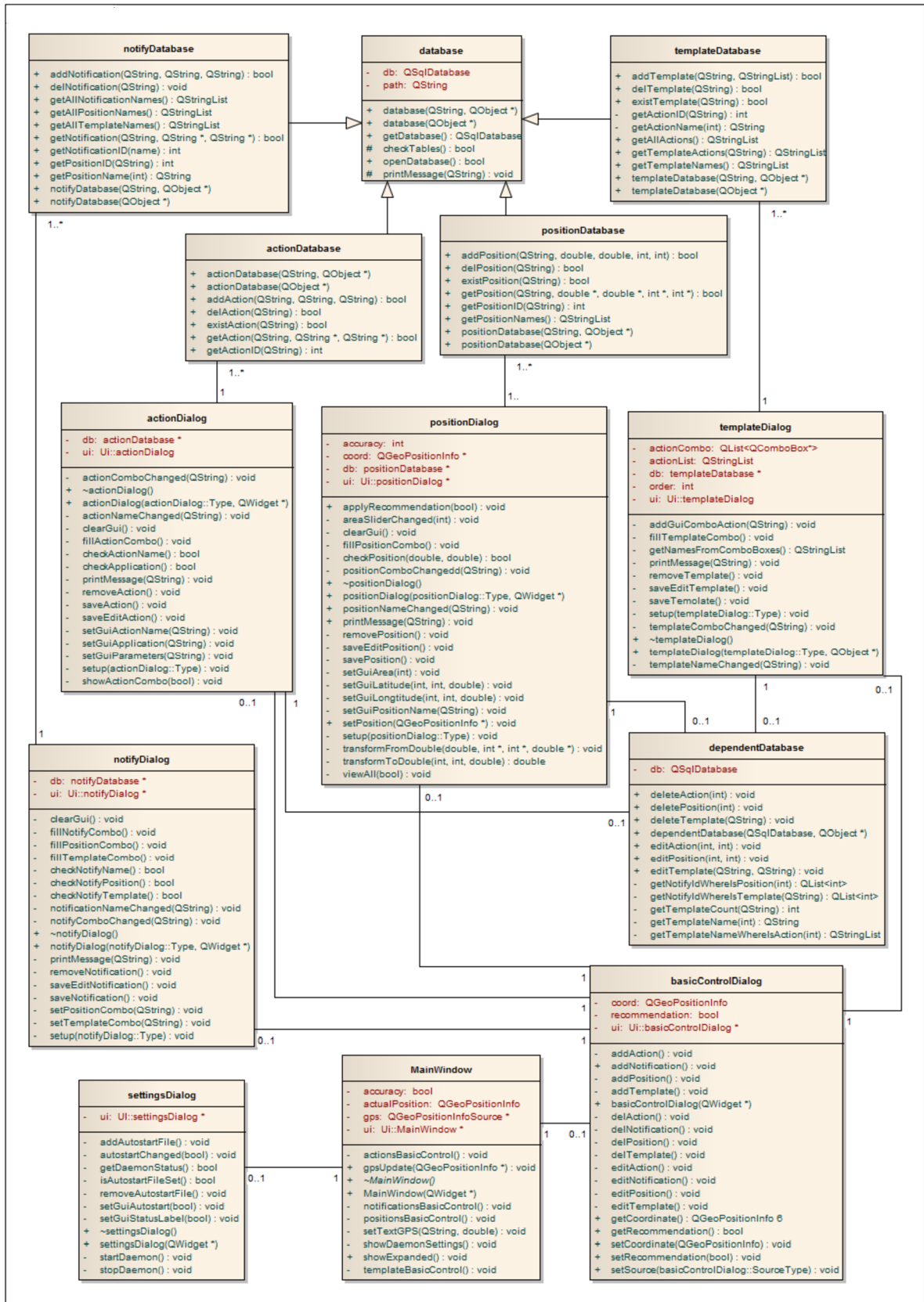
- [1] Blanchette, J.: *C++ GUI Programming with Qt 4*. Prentice Hall, 2008, ISBN 978-0-13-235416-5, 752 s.
- [2] Bosch, J.: Software Product Families in Nokia. In *SPLC*, 2005, s. 2–6.
- [3] Hojgr, R.; Stankovič, J.: *GPS: Praktická uživatelská příručka*. Computer Press, 2007, ISBN 978-80-251-1734-7, 256 s.
- [4] kolektiv autorů: *Global Positioning Systems, Inertial Navigation, and Integration*. John Wiley & Sons, 2002, ISBN 978-04-713-5032-3.
- [5] kolektiv autorů: *Understanding GPS: principles and applications*. Artech House, 2006, ISBN 1-58053-894-0.
- [6] kolektiv autorů: *Linux: Dokumentační projekt*. Computer Press, 2007, ISBN 978-80-251-1525-1.
- [7] kolektiv autorů: *Building embedded Linux systems*. O'Reilly Media, 2008, ISBN 978-0-596-52968-0.
- [8] Molkentin, D.: *The Book of Qt 4: The Art of Building Qt Applications*. Open Source Press GmbH, 2006, ISBN 978-3-937514-12-3.
- [9] Negus, C.: *Linux Bible 2010 Edition*. Wiley Publishing, 2010, ISBN 978-0-470-48505-7.
- [10] Owens, M.: *The Definitive Guide to SQLite*. Apress, 2006, ISBN 978-1-59059-673-9.
- [11] Prata, S.: *Mistrovství v C++*. Computer Press, 2007, ISBN 978-80-251-1749-1.
- [12] Rankin, K.: Nokia N900. *Linux Journal*, ročník 2010, č. 193, 2010: str. 11, ISSN 1075-3583.
- [13] Thelin, J.: *Foundation of Qt Development*. Apress, 2007, ISBN 978-1-59059-831-3.
- [14] Tuček, J.: *GIS - Geografické informační systémy: Principy a praxe*. Computer Press, 1998, ISBN 80-7226-0691-X, 438 s.
- [15] WWW stránky: Oficiální stránky produktu Nokia N900. [cit. 2011-11-04], [online]. URL <<http://www.nokia.cz/produkty/telefony/nokia-n900>>
- [16] WWW stránky: Oficiální stránky operačního systému Maemo. [rev. 2011-01-04], [online]. URL <<http://maemo.nokia.com/maemo>>

- [17] WWW stránky: Maemo SDK: User Guide. [rev. 2011-04-11], [online].
URL <<http://maemo-sdk.garage.maemo.org/user-guide.html>>

Příloha A

UML diagram tříd

UML diagram tříd znázorňuje celkové spojení a komunikace jednotlivých tříd, které byly vytvořené pro grafickou část aplikace.



Obrázek A.1: UML diagram tříd grafické části aplikace

Příloha B

Obsah CD

Datový nosič CD obsahuje:

- Text bakalářské práce ve formátu PDF.
- Text bakalářské práce ve formátu TEX.
- Obrázky použité při tvorbě textu.
- Demonstrační video.
- Zdrojové kódy aplikace s grafický uživatelským rozhraním
- Zdrojové kódy aplikace běžící v pozadí
- Skript ovládající aplikaci běžící v pozadí
- Instalační balíček aplikace pro mobilní zařízení Nokia N900