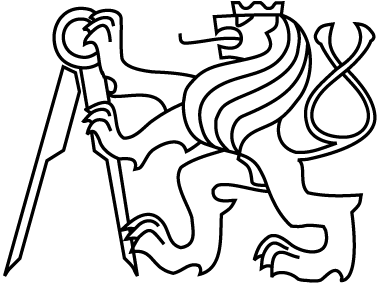
České vysoké učení technické v Praze  
Fakulta elektrotechnická  
Katedra počítačů



Bakalářská práce

Serverová část systému pro evidenci dopravních přestupků a pohybu policistů

*Pavel Brož*

Vedoucí práce: Ing. Martin Komárek

Studijní program: Softwarové technologie a management

Obor: Softwarové inženýrství

13. února 2012

Poděkování

Zde můžete napsat své poděkování, pokud chcete a máte komu děkovat.

Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a použil jsem pouze podklady uvedené v přiloženém seznamu.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č.121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne XX.XX.2012 Pavel Brož

Abstract

Abstrakt

Úvod

## Popis problému

Kdykoliv v dnešní době dojde k přestupku při parkování, policista musí na papír na místě vyplňovat parkovací lístek. Přitom si musí pamatovat nejrůznější čísla paragrafů zákona, zjišťovat adresu, kde se událost odehrála apod. Papírové materiály se navíc mohou kdykoliv ztratit a jsou nepraktické na přenášení a zvyšují administrativu.

Východiskem z této situace je vytvoření elektronického systému využívající moderní mobilní telefony, který by vyplňování parkovacího lístku urychlil a zjednodušil. Takovýto systém však není samostatně provozuschopný a proto je potřeba k aplikaci na mobilním zařízení přidat i další podpůrné systémy jako je aplikační server, databáze údajů a klientský systém na osobní počítač pro základní správu.

## Cíle bakalářské práce

Bakalářskou práci je možné rozdělit do čtyř samostatných částí, které pak budou spolupracovat a tvořit tak funkční systém. První částí je síťová vrstva mobilního klienta, která bude mít za úkol zprostředkovávat komunikaci se serverem. Druhá část práce se zabývá serverovou aplikací, která bude zpracovávat data z mobilního klienta a ukládat je do databáze a také bude mobilnímu klientovy poskytovat další služby. Třetí částí je aplikace pro osobní počítač na policejní služebně, skrze který bude možné zajišťovat správu mobilních klientů, prohlížet data z databáze a měnit PIN policistů. Čtvrtou částí, je databáze pro ukládání dat z mobilních zařízení a databáze pro zajištění fungování systému.

## Způsob vývoje

Vývoj systému Smart-Fine bude probíhat iterativně a inkrementálně. Délka iterací se bude řídit dynamicky s běžnou délkou iterace 3-4 týdny. Na konci každé iterace budou určeny cíle pro další iteraci. Pokud nebudou všechny stanovené cíle dokončeny v dané iteraci, bude jejich řešení přesunuto do další iterace.

V rámci vývoje budou vytvořeny dokumenty Project Overview Statement, analýza, návrh a testování, do kterých bude umístěn popis vývoje systému a do dokumentu bakalářské práce budou zapisovány změny, které byly v rámci iterace do příslušných dokumentů zaneseny.

Vývoj všech částí systému, kterých se týká tato bakalářská práce tj. síťová vrstva mobilního klienta, aplikační server a klientská aplikace na PC bude probíhat v jazyce Java. Jako vývojové prostředí pro síťovou vrstvu bude použito Eclipse IDE a pro ostatní části prostředí NetBeans IDE.

Iterace 1

## Vytvoření prototypu mobilního klienta

První iterace proběhla v rámci semestrálního projektu a předmětu řízení SW projektů a to jako týmový projekt. V této fázi byla hlavně zpracovávána část mobilního klienta a také byly stanoveny obecné požadavky na celý systém. Výsledkem iterace byly dokumenty Project Overview Statement, analýza a návrh a také i byl vytvořen prototyp mobilní aplikace, který byl podroben uživatelskému testování.

V dokumentu  Project Overview Statement byla jasně definována řešená problematika, cíle a přínosy, kterých má být tímto systémem dosaženo, shromážděny byly jak obecné, tak funkční požadavky na systém a to zvláště pro první iteraci. Již v této době byly však nalezeny požadavky, které byly natolik rozsáhlé, že bylo nerealistické jejich splnění a proto byly přidány s tím, že budou řešeny v dalších iteracích tj. jako součást bakalářské práce. Spolu s požadavky byla určena i kritéria úspěchu pro první iteraci. Vytipována byla také rizika a překážky, které mohly během vývoje nastat a spolu s tím byly vymyšleny způsoby jak těmto problémům předcházet a jak je řešit.

Dokument Project Overview Statement

Analýza systému byla z velké části zpracována v programu Enterprise Architect od firmy Sparx Systems. Soupis požadavků z Project Overview Statement byl převeden do diagramu požadavků. Z funkčních požadavků byly odvozeny případy užití a byl vytvořen diagram mapování funkčních požadavků na případy užití. Poté byly k jednotlivým případům užití sepsány scénáře. Součástí analýzy bylo také vytvoření doménového modelu a popsání významu atributů jednotlivých tříd.

Dokument analýza

Návrh byl podobně jako analýza zpracováván v programu Enterprise Architect. Nejprve byly z doménového modelu izolovány možné návrhové třídy a následně byly vytvořeny jejich vzájemné závislosti, atributy a metody. Všechny třídy, atributy a metody byly okomentovány tak, aby tyto popisky mohly být využity při generování kódu jako komentáře JavaDoc. Společně s třídami byly navrženy i balíčky, do kterých byly třídy následně umístěny. Při konstrukci návrhových tříd bylo identifikováno několik známých problémů, na jejichž řešení byly použity návrhové vzory DAO (Data Access Object) a Singleton.

Dokument návrh

Implementace mobilního klienta probíhala v aplikaci Eclipse. Pro zrychlení práce byla základní struktura kódu (rozhraní, třídy a příslušné metody a atributy) vygenerována z programu Enterprise Architect za použití návrhu vytvořeného předtím. Cílem implementace pak bylo doplnit výkonný kód do metod.

Souběžně s vývojem mobilního klienta (business logiky) vznikal i prototyp uživatelského prostředí. Nejprve byl vytvořen hrubý návrh v programu Balsamiq Mockup, který byl následně otestován heuristickou evaluací. V další fázi byl již vytvořen prototyp běžící na mobilním telefonu. Tento prototyp již využíval tříd business logiky a byl plně funkční s ohledem na určená kritéria úspěchu.

Iterace 2

## Sběr požadavků a návrh zpráv síťové komunikace

### Cíle iterace

Cíle druhé iterace je možné rozdělit na dvě základní části a to: sběr požadavků na serverovou aplikaci a aplikaci PC klienta a navržení zpráv pro síťovou komunikaci.

### Průběh iterace

V rámci iterace proběhlo nejprve hledání požadavků a posléze byl vytvořen návrh zpráv pro síťovou komunikaci mobilního klienta se serverem a PC klienta se serverem.

Hledání požadavků pro server a PC klienta proběhlo bez potíží, nalezené požadavky byly shromážděny pomocí aplikace Enterprise Architect.

Při návrhu zpráv pro síťovou komunikaci došlo k problému, že nebyl znám typ zabezpečení dat, průběh přihlašování k systému (serveru) a průběh používání mobilního klienta během služby.

Aby bylo možné navrhnout síťové zprávy bylo nejprve nutné zajistit bezpečnost dat při přenosu. Pro šifrování zpráv bylo zvoleno řešení za použití protokolu TLS, který je, co se týče zabezpečení dat, na vysoké úrovni a zároveň s tím, je jeho implementace přítomna v systému Android, takže je možné ji okamžitě použít. K zabezpečení dat bylo zvažováno i použití protokolu SSL, ale vzhledem k tomu, že TLS je novější a je stejně jako SSL podporován nativně v systému Android, byla zvolena varianta s protokolem TLS. Pro zabezpečení dat při přenosu mezi PC klientem a serverem, je použit protokol TLS také.

Protože protokol TLS poskytuje spolehlivé a šifrované spojení se serverem není nutné do síťových zpráv přidávat další kontrolní mechanizmy (např. CRC). O kontrolu dat před změnou během přenosu se starají protokoly nižších vrstev.

Aby bylo možné navrhnout vhodný způsob přihlašování bylo nejprve nutné zjistit, jak a kdy se mobilní klient přihlašuje na server. Z toho důvodu byl v aplikaci Enterprise Architect vytvořen diagram aktivit, zachycující používání mobilního zařízení během služby a diagram aktivit zachycující způsob připojení a přihlášení na server. Tyto diagramy pak pomohly k vytvoření vhodného způsobu přihlášení.

Následně byly navrženy i další síťové zprávy a to jak pro komunikaci mobilní klient - server, tak PC klient - server.

### Zhodnocení iterace

Během 2. iterace došlo ke splnění všech vytyčených cílů ovšem s několika nedokončenými částmi. Hledání požadavků s menší prioritou bude pokračovat i v dalších iteracích. Během návrhu síťových zpráv také nebyl znám formát, v jakém budou uložena geolokační data a tudíž nebylo možné zprávu pro přenos těchto dat mezi mobilním zařízením a serverem ale i PC klientem a serverem vytvořit.

Iterace 3

## Analýza serveru a PC klienta

### Cíle iterace

Cílem třetí iterace je provést analýzu aplikace serveru a aplikace PC klienta. Součástí analýzy bude vytvoření případů užití, jejich mapování na funkční požadavky a navrhnutí doménového modelu.

### Průběh iterace

Pro modelování případů užití byla použita aplikace Enterprise Architect.

Jako první byly izolovány role aktérů, kteří v případech užití vystupují. Většina aktérů je jednoduchých, až na role Policista a Nadřízený policista, kde bylo použito dědičnosti. Důvodem bylo, že aktér Nadřízený policista má zároveň vlastnosti role Policista, ale k tomu navíc přidává další, rozšiřující vlastnosti, které mu umožňují používat funkcionalitu PC klienta, kterou má k dispozici pouze policista, který velí na služebně (nadřízený policista).

Poté co byly známi aktéři, byly vytvořeny případy užití nejprve pro PC klienta a poté pro server. Jako výchozí body pro hledání případů užití posloužily funkční požadavky nalezené v předchozí iteraci. Případy užití v modelu PC klienta ukazují interakci mezi aktéry Policista a Nadřízený policista, tedy skutečnými lidmi a aplikací PC klienta. Naproti tomu případy užití v modelu serveru, ukazují hlavně interakci mezi počítačovými systémy a to mezi serverem a mobilním klientem a pak také mezi serverem a PC klientem. Několik málo případů užití, pak také bylo nalezeno pro interakci mezi administrátorem (aktér Administrátor) a serverem.

Ke všem případům užití byly vytvořeny scénáře užití za pomoci strukturovaných textových záznamů, které nabízí aplikace Enterprise Architect. Zároveň byly i všechny případy užití doplněny komentáři, vysvětlujícími, co daný případ užití popisuje, případně jaký je jeho kontext.

Pro kontrolu, zda jsou všechny případy užití, vytvořeny na základě nějakého funkčního požadavku, byla během jejich vytváření zaznamenávána závislost konkrétního případu užití na požadavku, který realizuje. Následně tak mohlo být za pomoci funkce Relationship Matrix aplikace Enterprise Architect kontrolováno, zda všechny případy užití, mají odpovídající funkční požadavek a zároveň, jestli jsou všechny funkční požadavky, realizovány nějakým případem užití. Pro dokument analýzy, pak byl sestaven i diagram mapování případů užití na požadavky.

### Zhodnocení iterace

Seznam použitých zkratek