

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií



Webová aplikace jako samostatné rozhraní informačního systému pro mobilní zařízení

Diplomová práce

Information system as an individual interface of web application for the mobile appliance

Diploma work

Studijní program: N2612 – Elektrotechnika a informatika
Studijní obor: Informační technologie
Autor: Bc. Petr Hendrych
Vedoucí práce: Ing. Roman Špánek, Ph. D.
Ústav mechatroniky a technické informatiky
Konzultant: Ing. Pavel Tyl
Ústav mechatroniky a technické informatiky

V Liberci 3. 1. 2010

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

Ústav mechatroniky a technické informatiky

Akademický rok: 2009/2010

ZADÁNÍ DIPLOMOVÉ PRÁCE

Jméno a příjmení: **Petr Hendrych**

studijní program: N 2612 – Elektrotechnika a informatika

obor: 1802T007 - Informační technologie

Vedoucí katedry Vám ve smyslu zákona o vysokých školách č.111/1998 Sb. určuje tuto diplomovou práci:

Název tématu:

Webová aplikace jako samostatné rozhraní informačního systému pro mobilní zařízení

Zásady pro vypracování:

1. Student se seznámí s metodami a technologiemi pro vytváření multiplatformních aplikací a aplikací pro mobilní zařízení.
2. Student nastuduje dostupné hardwarové a softwarové prostředky pro práci s webovými aplikacemi, které nabízí současné mobilní telefony a PDA.
3. Student provede analýzu problematiky CRM
4. Na základě provedené analýzy student navrhne a implementuje aplikaci pro mobilní zařízení umožňující přístup ke vzdáleně uloženým datům a metodám.

Rozsah grafických prací: dle potřeby dokumentace

Rozsah průvodní zprávy: cca 40 až 50 stran

Seznam odborné literatury:

SOAP

[1] Rob Allen, Nick Lo, Steven Brown: Zend Framework in Action, September 2007
Softbound , ISBN: 1933988320

[2] Steve Graham, Doug Davis, Simeon Simeonov, Glen Daniels, Peter Brittenham,
Yuichi Nakamura, Paul Fremantle, Dieter Koenig, Claudia Zentner: Building Web
Services with Java: Making Sense of XML, SOAP, WSDL, and UDD, Sams; 2 edition
(July 8, 2004), ISBN-13: 978-0672326417.

[3] Paul Golding: Next Generation Wireless Applications: Creating Mobile
Applications in a Web 2.0 and Mobile 2.0 World, Wiley; 2 edition (June 10, 2008),
ISBN-13: 978-0470725061

Vedoucí diplomové práce: Ing. Roman Špánek

Konzultant: Ing. Pavel Tyl

Zadání diplomové práce:

14. 10. 2009

Termín odevzdání diplomové práce:

21. 5. 2010

L.S.

.....
Doc. Ing. Petr Tůma, CSc.
CSc.

Vedoucí ústavu

.....
Prof. Ing. Václav Kopecký,

Děkan

V Liberci dne 16. 10. 2009

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé DP a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce.

Datum:

Podpis:

Poděkování

Tímto bych chtěl poděkovat všem, kteří mi pomáhali při vzniku této práce a tím přispěli k co nejlepšímu výsledku. Poděkovat bych chtěl hlavně velice ochotným a vstřícným pracovníkům firmy Digital Solutions, zejména pak Ing. Ondřejovi Říhovi a Janu Koppovi za skvělou spolupráci, ochotu, čas a vytvoření perfektních podmínek pro tvorbu aplikace. Poděkování patří také samozřejmě vedoucímu práce Ing. Romanu Špánkovi, Ph. D. za rady, věcné připomínky a čas, který mi věnoval.

Poděkování patří i mým rodičům za poskytnuté zázemí a trpělivost.

Abstrakt

Hlavním cílem práce bylo vytvořit informační systém přístupný mobilním telefonům. Práce se zabývá mobilními zařízeními jako platformou pro přístup k informačnímu systému a hodnotí možné cesty vzniku a implementace takového systému jako multiplatformní aplikace. Na základě provedených analýz vznikl informační systém Mstormm v podobě mobilní verze webové aplikace určený pro realitní makléře, který je stručně představen v poslední části práce. Zde jsou popsány jeho ovládací prvky, celková orientace v prostředí, možnosti jeho dalšího vývoje a problémy, na které jsem během vývoje narazil. V neposlední řadě se práce zabývá také problematikou CRM (Customer relationship management).

Klíčová slova: multipatformní, webové aplikace, Zend Framework, mobilní zařízení, Mstormm.

Abstract

The main aim of this diploma thesis consists of a creation of an information system being available through mobile phones. This thesis deals with mobile devices as a platform for an access to information systems and it evaluates possible ways of formation and implementation of this system as a multiplatform application. On the basis of analysis an informational system Mstormm was created. This system has the form of web application which is designed for real estate agents. In the last part of this thesis there is a brief introduction to the application, its controls, overall orientation in the environment, possibilities of further development and problems, which were solved during the development process of this application. Finally, this thesis also deals with the issue of CRM (Computer relationship management).

Key words: multiplatform, web application, Zend Framework, mobile devices, Mstormm.

Obsah

Prohlášení	4
Poděkování.....	5
Abstrakt	6
Obsah	7
Úvod	8
1. Metody a technologie pro vytváření multiplatformních aplikací.....	9
1.1. Co je multiplatformní aplikace	9
1.2. Základní typy aplikací.....	10
1.3. Webová aplikace vs. GUI řešení.....	10
2. Hardwarové a softwarové prostředky pro práci s webovými aplikacemi na současných mobilních telefonech.....	14
2.1. Úvod do problematiky	14
2.2. Přehled mobilních telefonů	15
2.3. Přehled mobilních prohlížečů	15
2.4. Internet v mobilním telefonu	19
2.5. Analýza problematiky CRM.....	22
3. Informační systém Mstormm.....	26
3.1. Cíle aplikace.....	26
3.2. Použité technologie a jejich aplikace.....	26
3.3. Seznámení s prostředím aplikace.....	31
3.4. Vzniklé problémy a jejich řešení	38
3.5. Testování aplikace	45
3.6. Budoucí vývoj.....	48
Závěr	49
Seznam použité literatury a zdrojů informací	50
Seznam příloh.....	53

Úvod

Cílem práce bylo vytvořit přehledný a uživatelsky přátelský informační systém, který umožní realitním makléřům přístup k pro ně důležitým informacím pomocí jejich mobilních telefonů kdekoliv v terénu při přímém kontaktu se zákazníkem. Šlo mi tedy především o vytvoření multiplatformního informačního systému, který bude dobře využitelný na co nejširším spektru mobilních telefonů a tím i pohodlně, rychle a flexibilně dostupný maximálnímu počtu uživatelů. Chtěl jsem se seznámit s moderními metodami tvorby webových aplikací a zjistit jejich možnosti pro tvorbu tohoto typu, která vyžaduje jisté optimalizace oproti klasickým webovým aplikacím určeným pro PC. Dále jsem chtěl získat znalosti spojené s tvorbou a návrhem databází, vytvářením SQL dotazů pro vyhledávání požadovaných dat a také přehledným zpracováním a zobrazováním těchto dat znamenajících jejich transformaci v plnohodnotné informace, které jsou pro informační systém nezbytné. Posledním velkým motivačním aspektem byla možnost zkusit si realizovat praktickou aplikaci pro reálného zákazníka, která bude skutečně nasazena do provozu a využívána řadou uživatelů. Toto s sebou samozřejmě neslo nutnost aktivně komunikovat jak s odborníky z vývojového týmu počítačové verze aplikace, kteří mi poskytovali potřebné zázemí, nástroje a informace nutné pro růst aplikace, tak s finálním zákazníkem v budoucnu využívajícím vzniklý systém, aby vše bylo dle jeho představ. Zkušenosti nabitě během tohoto projektu jsou pro mé budoucí uplatnění v praxi, kde je kolektivní spolupráce základním předpokladem úspěchu, nedocenitelné.

1. Metody a technologie pro vytváření multiplatformních aplikací

1.1. Co je multiplatformní aplikace

Multiplatformní mobilní software je termín, který je užíván ve spojení s aplikacemi či operačními systémy, které mohou běžet na více mobilních platformách. Pod pojmem platforma se obecně míní kombinace hardware a software, která je nutná pro spuštění softwarového produktu. To znamená, že vzniklý program může například běžet jak na operačním systému Symbian, který využívají především mobilní telefony finské společnosti Nokia, tak na operačním systému OS X pro Apple iPhone.

Příklady nejznámějších mobilních operačních systémů:

- Windows Mobile
- Symbian
- Android
- OS X

Software, který je veden jako multiplatformní, je buď:

- dodáván pro každou platformu zvlášť
- nese si s sebou kód pro více platforem
- je k dispozici v mezikódu, který je dle příslušného operačního systému a hardwarové platformy interpretován (např. aplikace napsané v Javě)

Každá uváděná forma má své výhody a nevýhody. Například software uložený v mezikódu (např. Java či .NET) ke své funkčnosti potřebuje runtime, který je nutné nainstalovat. Takový kód je pak po spuštění překládán do instrukcí pro danou platformu. Běh je tedy často pomalejší než v ostatních uvedených variantách.

[\[1\]](#)

1.2. Základní typy aplikací

Jako vývojář jsem musel nejprve rozhodnout, o jaký typ aplikace se bude jednat. Na výběr jsem měl dvě základní varianty. Buď bude aplikace tvořená jako webová, která bude následně přístupná na všech mobilních platformách s webovým prohlížečem, nebo jako desktopová GUI¹ aplikace instalovaná na příslušné zařízení. Webové aplikace jsou takové, které běží na serveru a k uživateli se přenáší pouze uživatelské rozhraní přes protokol HTTP(S) v popisném jazyku (X)HTML. V některých případech se část logiky též přesouvá na webový prohlížeč a to pomocí programovacího jazyku JavaScript. GUI aplikace oproti tomu běží přímo na pracovní stanici klienta. Obě varianty mají samozřejmě své klady i zápory a před konečným rozhodnutím jsem musel zvážit několik aspektů rozebraných v další části této kapitoly.

[\[2\]](#)

1.3. Webová aplikace vs. GUI řešení

1.3.1. Nutnost instalace

Zde je výhodnější webová aplikace. Stačí mít na dané platformě nainstalován kvalitní webový prohlížeč a aplikace je dostupná. Instalovat se nemusí nic. GUI aplikace se naopak instalovat musí přímo na pracovní stanici. To může být v případě mobilních telefonů vzhledem k velkému množství jejich různých typů lišících se jak výkonem, tak použitým operačním systémem problém.

[\[2\]](#)

1.3.2. Bezpečnost dat

Toto je problém složitější a nelze jednoznačně určit lepší výběr. Bezpečnost dat lze charakterizovat jako zachování důvěrnosti, integrity a dostupnosti informací.

- **Důvěrnost** je zajištění přístupu informace jen pro ty, kteří jsou k tomu oprávněni.
- **Integrita** dat je zabezpečení kompletnosti informací a metod jejich zpracování.
- **Dostupnost** zajišťuje, že informace jsou přístupné uživatelům v době, kdy je potřebují.

¹ Grafické uživatelské rozhraní

Bezpečnostní politiku lze pak rozdělit na několik oblastí:

Provozní bezpečnost

Cílem provozní bezpečnosti je zajistit správný a bezpečný provoz prostředků pro zpracování informací a minimalizovat riziko selhání systému.

Komunikační bezpečnost

Cílem komunikační bezpečnosti je ochrana informací v průběhu výměny dat.

- Šifrování různých toků (user - WWW, WWW - DB, user - DB...)
- Autentizace - ověření totožnosti (čipové karty, certifikáty)
- Autorizace - ověření práv pro vykonání určité činnosti
- Elektronický podpis
- Firewall

Bezpečnost dat

Cílem bezpečnosti dat je zabezpečení informací proti jejich zneužití.

- Zabezpečení dokumentů
- Zabezpečení dat

Zajištění bezpečnosti dat závisí hodně na typu aplikace. Pokud jde o firemní aplikaci pracující s velkým množstvím s časem se měnících dat, pak by tato data měla být na serveru a to také většinou u všech typů aplikací jsou, tedy i u GUI aplikací.

[\[2\]](#)[\[3\]](#)

1.3.3. Spolehlivost

Pokud aplikace běží jako webová na serveru a uživatel se na ní může připojit kdykoliv z jakékoliv pracovní stanice s připojením k internetu, je webová aplikace výhodnější. Pokud ovšem uživatel má občas potíže s její dostupností skrze internet, není aplikace na serveru výhodou, ale naopak. U nasazení webových aplikací je výhodou i snadnější zálohování dat.

U spolehlivosti z hlediska kvality programového kódu je na tom asi lépe GUI aplikace. Webové aplikace dnes totiž často využívají Ajax (Asynchronous JavaScript and XML), který se obtížně ladí a udržuje. Zároveň u něj není stoprocentně zaručeno, že bude u všech uživatelů plně funkční. Jednak jde o jeho interpretaci webovým prohlížečem, jednak je problémem, že sám uživatel může mít JavaScript v prohlížeči zakázán a proto není dobré se na něj stoprocentně spoléhat.

[\[2\]](#)

1.3.4. Ovladatelnost a vzhled

Závažným problémem u webových aplikací je zajištění rozumného vzhledu napříč všemi webovými prohlížeči, na kterých by mohl případný uživatel aplikaci spouštět. Ne všechny mobilní internetové prohlížeče interpretují popisný jazyk (X)HTML stejně a ne všechny mobilní prohlížeče mají stejné schopnosti zpracování JavaScriptu a kaskádových stylů. Proto například stránky vypadající dobře v prohlížeči OperaMini už nemusejí vypadat stejně elegantně v prohlížeči Safari Mobile a naprosto perfektně fungující stránky v prohlížeči Nokia Mini Map Browser nemusejí korektně fungovat v prohlížeči Opera Mobile. Proto je nutné vznikající aplikaci neustále testovat na nejpopulárnějších prohlížečích a její vzhled optimalizovat pro jejich prostředí.

[\[2\]](#)

1.3.5. Multiplatformnost

Možnost nasadit aplikaci na různé platformy má více rozměrů. Je nutné určit, jaké bude cílové zařízení. Může jít o osobní počítač, mobilní telefon, elektronický diář apod. V našem případě jde tedy o mobilní telefony. Zde je potom otázkou vlastní internetový prohlížeč. Pro všechny důležité mobilní platformy je k dispozici minimálně jeden webový prohlížeč, který dostatečně ctí standardy a umožňuje tak ovládat a zobrazovat webovou aplikaci, proto můžeme považovat řešení v podobě webové aplikace za multiplatformní.

U desktopových aplikací to tak snadné není. V moderní informatice se tradičně distribuují desktopové aplikace koncovým uživatelům jako binární obrazy, které jsou uloženy jako spustitelné soubory ve speciálním binárním formátu. Takové spustitelné soubory ovšem podporuje pouze operační systém a architektura, pro kterou byly vytvořeny. To znamená, že takové programy nemůžeme považovat za multipatformní.

Když strojový čas platforem zlevnil, začaly se objevovat jazyky, které nejsou zaměřené na vytvoření maximálně efektivního kódu, ale které si kladou za primární cíl přehlednější a pohodlnější zápis algoritmů a zmenšení problémů při přenosu programů mezi různými architekturami zařízení. Díky tomu vznikly interpretované jazyky. Jejich zdrojový kód není překládán přímo do strojového kódu, ale je interpretován pomocí speciálního programu - interpretu. Interpret tedy umožňuje přímo vykonávat zápis jiného programu v jeho zdrojovém kódu a umožňuje tak programování kódu, který je snadno přenositelný mezi různými platformami.

V současnosti je nepopulárnějším nástrojem pro vývoj multiplatformních aplikací pro mobilní telefony jedna ze základních platform Javy - Java ME, dříve označovaná jako Java 2 Micro Edition nebo J2ME. Jedná se o sbírku API² určenou pro vývoj software pro malá zařízení a zařízení s omezenými prostředky. V současné době většina výrobců mobilních telefonů implementuje J2ME ve svých výrobcích.

[\[2\]](#)[\[4\]](#)[\[5\]](#)[\[6\]](#)

1.3.6. Konečný výběr

Při závěrečném rozhodnutí nejvíc záleží na samotném typu tvořené aplikace. Je třeba se zamyslet nad různými aspekty a podle toho zvolit její nejvhodnější typ. V úvahu je třeba brát, pro koho bude cílová aplikace určena, na jakých platformách bude spouštěna, jak bude rozsáhlá a komplikovaná z hlediska ovládání. V případě, že je aplikace malá, nenáročná na ovládání, je důležitá její multiplatformnost a uživatele nesmí odradit nutná instalace od jejího užívání, je výhodnější webová aplikace.

Na základě zvážení všech pro a proti, jednání se zákazníkem a odborným konzultantem z firmy Digital Solutions bylo rozhodnuto, že bude řešená jako mobilní verze webové aplikace a bude tedy dostupná makléřům realitní kanceláře skrze webový prohlížeč jejich mobilních telefonů.

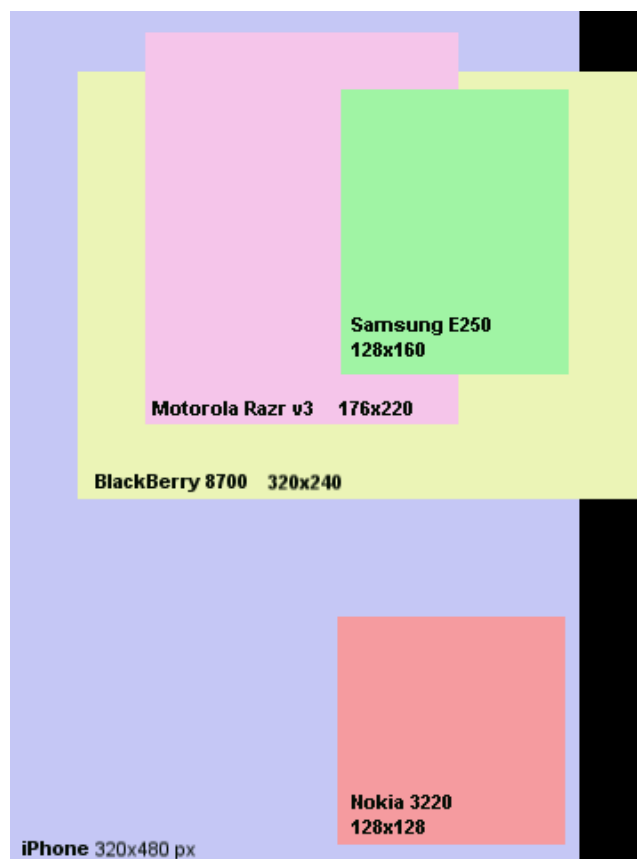
K tomuto rozhodnutí vedly především tyto konkrétní faktory. Přestože existují v dnešní době kvalitní a silné nástroje pro vývoj multiplatformních desktopových GUI aplikací pro mobilní telefony, stále zde neodpadá problém distribuce a instalace vzniklého produktu na konečnou pracovní stanici. To by mohla být v případě využívání vzniklého systému velká překážka. Pro zákazníka bylo mnohem pohodlnějším řešením, aby aplikace fungovala na mobilních prohlížečích defaultně předinstalovaných do zařízení přímo výrobcem bez nutnosti stahování a instalování externích souborů. Dalším významným plus pro webovou aplikaci bylo, že toto řešení, jak se v průběhu analyzování problému ukázalo, bude univerzální napříč všemi operačními systémy mobilních zařízení a bude tak mít menší nároky na technickou podporu uživatelů.

² API označuje v informatice rozhraní pro programování aplikací.

2. Hardwarové a softwarové prostředky pro práci s webovými aplikacemi na současných mobilních telefonech

2.1. Úvod do problematiky

Na světě je okolo 3.3 mld. kusů mobilních telefonů a většina z nich je schopna zobrazit webové stránky. Tvorba mobilních stránek ovšem naráží na velký problém s kompatibilitou. Není snadné udělat mobilní stránky tak, aby se správně zobrazily na všech typech telefonů. Mezi lidmi jsou totiž přístroje různých kvalit a cen. Například rozlišení displeje bývá přístroj od přístroje velmi rozdílné. Na obrázku je porovnána velikost obrazovek pěti mobilních telefonů.



Obrázek 1. Porovnání obrazovek mobilních telefonů

Přitom velikost displeje je jen malá část celého problému nekompatibility. Existují další rozdíly v přítomnosti a nepřítomnosti různých kláves, existence řady mobilních internetových prohlížečů, jejich rozdílných schopností, atd.

Pro efektivní splnění daného zadání a požadavků kladených na aplikaci bylo tedy nezbytné zjistit, na jakých platformách a mobilních prohlížečích bude aplikace pravděpodobně nejčastěji používána, aby mohla být co nejlépe optimalizována na působení těchto dvou faktorů. Proto je další část této kapitoly věnována přehledu nejprodávanějších telefonů na českém trhu, pro ně nepoužívanějším mobilním prohlížečům a jejich možnostem.

[7]

2.2. Přehled mobilních telefonů

Žebříčku nejprodávanějších telefonů za rok 2009 na českém trhu kraluje finská společnost Nokia s odhadovaným podílem 45 procent. Zájem je především o levné dotykové telefony. Tyto mobilní telefony nejčastěji používají operační systém Symbian s defaultně přeinstalovaným prohlížečem Nokia Mini Map Browser. Na druhém místě je společnost Sony Ericsson s odhadovaným podílem 30 procent. Telefony značky Sony Ericsson využívají nejčastěji webový prohlížeč NetFront. Posledním ze třetice nejvýznamnějších výrobců mobilních telefonů pro český trh je Samsung se svým patnáctiprocentním podílem. Telefony společnosti Samsung opět často využívají prohlížeč NetFront, ty nejnovější pak nový prohlížeč Dolhin. Velkým hitem poslední doby je telefon iPhone od společnosti Apple s operačním systémem označovaným jako OS X, který využívá mobilní verzi prohlížeče Safari.

[8]

2.3. Přehled mobilních prohlížečů

Stále více uživatelů používá mobilní telefon pro přístup k internetu. K tomu lze využít buď prohlížeč, který je integrovaný v telefonu od výrobce, nebo si jej doinstalovat. Zásadní průlom udělala japonská Opera Mini. S rostoucími úhlopříčkami displejů a s rozšířením displejů dotykových rostly i požadavky na mobilní prohlížeče. Dalším mezníkem byl Apple iPhone. Různých předinstalovaných i doinstalovatelných prohlížečů pro mobilní telefony je hodně, ale zdaleka ne všechny jsou stejně kvalitní. V současné jsou na trhu nejvíce rozšířené tyto prohlížeče: Opera Mini, Opera Mobile, Safari Mobile, Nokia Mini Map Browser, NetFront a Internet Explorer Mobile.

[9][10]

2.3.1. Opera Mini

Opera Mini funguje na principu předzpracovávání dat proxy serverem³ – do telefonu se tedy stahuje předem zpracovaná podoba stránek, která je vhodná pro zobrazení na displeji mobilního telefonu. Výhodou je, že objem stahovaných dat je menší než v případě otevírání celých stránek, což znamená, že se nám stránka zobrazí rychleji a ušetříme za přenesená data. Datovou náročnost máme také šanci sami ovlivnit nastavením umožňujícím zakázání stahování obrázků nebo jejich stahování v nižší kvalitě. Opera Mini funguje ve dvou zobrazeních. První je klasické a to druhé mobilní. V mobilním zobrazení se celá stránka zobrazí v jednom sloupci, tak abychom nemuseli skrolovat do stran. V klasickém zobrazení se stránka na displeji zobrazí podle jejího původního formátování. Pokud je zobrazená stránka o hodně větší než rozměr displeje mobilního telefonu, je daná stránka zobrazená na displeji celá ve své miniaturní podobě a uživatel si může zoomem přiblížit požadovanou část stránky. Pro Operu Mini stačí mobilní telefon s podporou Javy. Výhodou je, že je zdarma a že nabízí podporu češtiny.

[\[9\]](#)[\[10\]](#)

2.3.2. Safari Mobile

Výhody a nevýhody mobilního Safari jsou do vysoké míry spojené s koncepcí přístrojů iPhone a iPod. Těží především z dotykového displeje, díky čemuž je na iPhonu prohlížení stránek velice snadné. Safari Mobile je dobře vyladěné a dobře využívá kapacit sítí mobilních dat - načítání stránek sice trvá déle než na Opeře Mini, ale vzhledem k tomu, o kolik víc při tom Safari stahuje dat, se dá považovat za nejrychlejší prohlížeč. Safari mobile patří mezi vyspělejší prohlížeče pro mobilní telefony. Tento prohlížeč má ale i své nevýhody. Chybí zde například zpracování flashe, správce hesel nebo prohledávání textu. Počet oken, které dokáže otevřít najednou, je omezen na 8, přestože hardware přístroje je natolik výkonný, že by s vyšším neměl mít problém.

[\[9\]](#)[\[10\]](#)

³ Proxy server funguje jako prostředník mezi klientem a cílovým počítačem (serverem), překládá klientské požadavky a vůči cílovému počítači vystupuje sám jako klient.

2.3.3. Opera Mobile

Prohlížeč nahrazuje Internet Explorer pro Windows Mobile. V současné době je nejvíce rozšířená její verze 9.5. Opera Mobile si vede výborně v testech webových standardů Acid a ve speciálním testu přímo pro mobilní prohlížeče od společnosti W3C. Opera Mobile je také poměrně úsporná, rychlá a nabízí podporu Flash Lite 3.0. Maximálně lze mít v jednom okamžiku otevřeno pět oken. Prohlížeč běží v fullscreen režimu a tak je celý displej vyhrazen pouze internetové stránce. Další výhodou, která zpohodlní prohlížení webu je takzvaná „gumovost“, kterou nabízí zmíněné Safari na Apple iPhone. Pokud jsme vlastníky dotykových displejů lze tahy prstů stránkou libovolně posouvat a rolovat. Funkci zoomu nabídla již Opera Mini. Plnohodnotná Opera jde ještě dále a tak dvojklikem na displeji dosáhneme optimálního přiblížení zobrazené stránky, jeho opakováním pak prohlížeč nabídne opět výchozí zobrazení.

[9][10]

2.3.4. Nokia Mini Map Browser (Symbian S60 browser)

Nokia Mini Map Browser je v každé symbianové Nokii. Tradiční symbianový internetový prohlížeč je dnes velmi kvalitní a užitečný nástroj, který využívá stejný vykreslovací engine – WebKit⁴ – jako Safari na Apple iPhonu. Největší problém u prohlížeče Nokia představuje dlouhé načítání stránek. Prohlížeč Nokia naopak boduje funkcemi. Dokáže stahovat soubory, má použitelnou RSS čtečku a podporuje Flash. Zvládne najednou až 20 otevřených oken.

[9][10]

2.3.5. NetFront

Prohlížeč NetFront je produktem společnosti ACCESS. Nesází na moderní vzhled jako Opera Mobile. Možností nastavení vykreslování je ale hodně. Podařená je volba Virtual Canvas. Zde si můžeme zvolit rozlišení, na které má být optimalizována prohlížená stránka. Zajímavá je funkce automatického posouvání stránky nebo možnost prohlížení stránek off-line. Nechybí mu podpora RSS kanálů, vlastní správce hesel, blokování pop-up oken a podpora technologií, jako je např. Macromedia Flash Player.

[9]

⁴ WebKit je knihovna pro vykreslování HTML

2.3.6. Internet Explorer Mobile

Mobilní Internet Explorer byl po dlouhou dobu jakýmsi standardem mezi prohlížeči na platformě Windows Mobile. Prohlížeč má bohužel problém s formátováním a podpora moderních funkcí zde také chybí. Hlavním problémem Mobile Internet Exploreru je zastaralé uživatelské prostředí. Toto všechno se ale mění od vydání jeho nové verze Internet Explorer Mobile 6.

Vykreslovací jádro této verze špatně Acid testy, ale s běžnými weby si poradí velmi dobře. Prohlížeč je díky integraci do systému svižný, hlavně jeho start je velice rychlý. Vykreslení stránky je srovnatelné s ostatními prohlížeči, které nepoužívají proxy server s kompresí dat. Po instalaci Flash Lite 3.1 si Internet Explorer Mobile 6 dobře poradí také s flashem. Dále nový Internet Explorer obsahuje rozšířenou podporu skriptů a technologie AJAX.

[9][11]

2.3.7. Vyvozené závěry

Přestože všech šest výše zmíněných prohlížečů je velice rozšířených, byla aplikace během vývoje testována a primárně optimalizována pro Operu Mini a to především z důvodu její dostupnosti. Přestože se Opera Mini ukázala jako velice populární prohlížeč u nás, ne každý majitel mobilního telefonu ji má ve svém přístroji a ne každý je ochotný a schopný si ji do mobilu doinstalovat. Proto bylo potřeba věnovat pozornost i dalším mobilním prohlížečům. Vytvořit stránky dokonale optimalizované pro všechny mobilní prohlížeče by ovšem bylo technicky náročné. Toto řešení by nejspíše zahrnovalo rozpoznávání jednotlivých prohlížečů a vytvoření pro každý samostatný layout stránek. Jako kompromisní řešení bylo se zákazníkem dohodnuto, že bude dále věnována pozornost prohlížečům postaveným na vykreslovacím jádru WebKit. Na tomto jádru je založena řada mobilních prohlížečů, mezi které patří i Nokia Mini Map Browser a prohlížeč Safari Mobile. Mezi další příklady prohlížečů založených na tomto jádru patří i Google Chrome nebo Iris Browser.

2.4. Internet v mobilním telefonu

2.4.1. Úvod

Přístup na internet přes mobilní telefon nebo stahování elektronické pošty do něj je v Česku stále populárnější. Nyní tyto služby podle údajů mobilních operátorů využívá kolem 2,2 milionu Čechů. Rychlý přístup na internet je jednou z hlavních oblastí, do které operátoři investují. V České republice se nyní můžeme připojit nejčastěji celkem pěti různými technologiemi v rámci mobilní sítě, jedná se o GPRS, EDGE, UMTS, HSDPA a CDMA.

Mobilní operátor	Poskytované datové technologie
T-Mobile (www.t-mobile.cz)	CSD, GPRS, EDGE, UMTS
Telefónica O2 (www.cz.o2.com)	CSD, HSCSD, GPRS, EDGE, CDMA 1xEV-DO, UMTS, HSDPA
Vodafone (www.vodafone.cz)	CSD, GPRS, EDGE, UMTS
U:fon (http://www.ufon.cz/)	CDMA 1xEV-DO

Tabulka 1. Přehled datových technologií

[12]

2.4.2. GPRS

Technologie GPRS (General Packet Radio Service) je realizována přenosem signálu v síti GSM (Global System for Mobile communications) určenou pro mobilní telefony v Evropě na frekvencích 900/1800 MHz. Nejstarší typ přenosu dat, který po svém spuštění odstartoval malou revoluci. Do té doby se bylo možné připojit k internetu pouze pomocí vytáčeného spojení, kdy se platilo za čas strávený on-line, na rozdíl od GPRS, u něhož se platí pouze za stažená data. Největší výhodou GPRS je jeho rozšíření – je všude tam, kam sahá mobilní síť a až na drobné výjimky ho podporují všechny mobilní telefony a modemy. Velkým nedostatkem je naopak rychlost, která se teoreticky pohybuje kolem 80 Kb/s pro download a 40 Kb/s pro upload, ovšem ve skutečnosti se reálná rychlost pohybuje kolem 50 Kb/s pro download a ještě méně pro upload.

Mobilní operátor	Míra pokrytí území GPRS
T-Mobile	98 %
Telefónica O2	99 %
Vodafone	94 %

Tabulka 2. Míra pokrytí GPRS

[\[12\]](#)[\[13\]](#)

2.4.3. EDGE

Jedná se o jakousi nadstavbu, jež pracuje na podobném principu jako GPRS. Přenášená data se ale jinak modulují, čímž bývá dosaženo větších přenosových rychlostí, které mohou teoreticky dosahovat až 237 Kb/s. EDGE není automaticky v místě signálu operátora, jako je tomu u GPRS. V současné době je EDGE přítomné na většině BTS⁵ operátora T-Mobile a Vodafone. Pokud není dostupné, bude telefon pracovat s GPRS.

Mobilní operátor	Míra pokrytí území EDGE
T-Mobile	92 %
Telefónica O2	16%
Vodafone	86 %

Tabulka 3. Míra pokrytí EDGE

[\[12\]](#)[\[13\]](#)

2.4.4. UMTS/ HSDPA

Sítě 3. generace, od kterých se hodně slibovalo, se nakonec zatím příliš nerozšířily. Oproti předchozím technologiím GPRS a EDGE fungujícím na síti GSM, která byla primárně vytvořena pro hlasové přenosy, byla síť UMTS projektována plně pro využití hlasových i datových přenosů, což ji řadí do skupiny technologií generace 3. Síť UMTS funguje na frekvenci 1885-2200 MHz. Teoretická maximální přenosová rychlost je

⁵ **Systém základnových stanic** je část sítě GSM, která je zodpovědná za přenos a příjem radiových signálů z mobilního telefonu. Systém základnových stanic provádí překódování hovorových kanálů, přidělování radiových kanálů mobilním telefonům a mnoho dalších úkolů patřících k radiové síti.

2 Mb/s. Síť UMTS byla v České republice spuštěna v roce 2005. Od té doby byly vydány nové specifikace, které vedly ke zvýšení přenosových rychlostí. Jedna z technologií se označuje jako HSDPA (High-speed Downlink Packet Access) a umožňuje zvýšit přenosovou rychlost až na 14,4 Mb/s, resp. 48 Mb/s v sestupném směru. HSDPA je založeno na několika inovacích architektury sítě, díky nimž se dosahuje nižšího zpoždění, rychlejších reakcí na změnu kvality kanálu a zpracování H-ARQ, tedy Hybrid automatic repeat request, hybridního automatického požadavku na opakování přenosu. Pokrytí je stejné jako u technologie UMTS.

Mobilní operátor	Míra pokrytí území UMTS
T-Mobile	34 %
Telefónica O2	4 %

Tabulka 4. Míra pokrytí UMTS

[\[12\]](#)[\[13\]](#)

2.4.5. CMDA

Základním principem funkce CDMA je umožnění současné komunikace více uživatelů v rámci jednoho frekvenčního pásma. Tyto mobilní technologie jsou realizovány v síti zvané CDMA2000, která funguje na frekvenci 450-2100 MHz. V síti CDMA2000 jsou v současnosti používány dvě technologie pro datový přenos - jsou jimi CDMA2000 1xEV-DO a CDMA2000 1xRTT. Technologie 1xEV-DO slouží pouze k datovému přenosu a umožňuje dosahovat přenosové rychlosti až 2.4 Mb/s, technologie 1xRTT umožňuje realizovat přenos hlasový i datový, ten ovšem pouze při maximální dosahované rychlosti 307 kb/s. V porovnání s technologiemi GPRS či EDGE dosahuje tento typ připojení výrazně lepších výsledků, co se týče přenosové rychlosti ale i latence, která se pohybuje kolem 300 ms.

Mobilní operátor	Míra pokrytí území CMDA
Telefónica O2	90 %
U:fon	57 %

Tabulka 5. Míra pokrytí CMDA

[\[12\]](#)

2.5. Analýza problematiky CRM

2.5.1. Customer relationship management

Účelem tvořeného informačního systému byla podpora vztahu realitních makléřů se zákazníky. Jde tedy o systém podpory CRM - Customer relationship management. CRM je databázovou technologií podporovaný proces shromažďování, zpracování a využití informací o zákaznících firmy. Je tedy firmami využíván především k poznání, pochopení a předvídání potřeb, přání a nákupních zvyklostí zákazníků. Podporuje také oboustrannou komunikaci mezi firmou a jejími zákazníky. Jako CRM v přeneseném smyslu též označujeme softwarové, hardwarové a personální vybavení firmy, které je výkonem těchto funkcí pověřeno.

[\[14\]](#)

2.5.2. Filozofie CRM

Podstatou CRM by měl být čistý proces, jehož výchozím bodem by měla být hodnota pro zákazníka. Pořadí otázek je následující: Jakou hodnotu můžeme vyprodukovat pro naše zákazníky? Jaké postupy a produkty potřebujeme k tomu, abychom toho dosáhli? A jakých kompetencí je zapotřebí k tomu, abychom uspěli? Každá společnost by se tedy měla zabývat těmito problémy:

1. Tvorba marketingové strategie k udržení stávajících zákazníků a získání nových
2. Porozumění zákazníkům
3. Identifikace klíčových procesů
4. Zvyšování spokojenosti zákazníků při zlepšování klíčových procesů

Řízení vztahů se zákazníky je strategie, která se orientuje na vybudování a podporu dlouhotrvajících vztahů se zákazníky. Není to tedy jen technologie, ale změna filosofie společnosti tak, aby důraz byl kladen na zákazníka a interakce s ním. Přínos CRM je zřejmý: zefektivnění procesů a poskytnutí obchodníkům, marketingu a vedení společnosti lepší, podrobnější informace o zákaznících. CRM pomáhá firmám vytvořit více profitabilní vztah se zákazníkem a snížit operativní náklady. Na nedodržování této strategie havaruje většina implementací CRM.

[\[14\]](#)[\[15\]](#)

2.5.3. Vývoj CRM

Systémy CRM se používají pro specifické činnosti již delší dobu, ale z důvodu například nových obchodních modelů, internetu, či elektronické komunikaci, se kompletně mění pohled na tuto oblast. V dnešní době je třeba změnit orientaci z produktů na zákazníky. Klíčovým termínem marketingu první poloviny minulého století bylo „4P“: product (výrobek), price (cena), place (umístění) a promotion (propagace). Firmy zaměstnávaly týmy prodejců, kteří se snažili produkt protlačit na trh, ať už byl zájem trhu o něj jakýkoliv. V okamžiku, kdy se objevily první marketingové průzkumy, výrobci a prodejci se začínali spíše orientovat na to, co si trh žádá. Moderním termínem je tedy spíše „4C“: customer total cost (náklady), customer value (hodnota), convenience (komfort) a communication (komunikace).

CRM se stává klíčovou záležitostí pro organizace všech velikostí. Všechny firmy mají totiž své zákazníky a bez ohledu na svou velikost, oblast podnikání, produkty nebo služby potřebují všechny podniky efektivně zpravovat vztahy se svými zákazníky. Udržení dobrého zákazníka je stejně důležité jako získání nového. V minulých letech bylo ale nasazení těchto systémů výsadou pouze velkých společností, které měli prostředky na nákladnou implementaci a provozování těchto aplikací. V současné době jsou řešení CRM cenově dostupné i středním či malým firmám.

[\[14\]](#)[\[16\]](#)

2.5.4. CRM jako produkt a jeho aplikace

S názvem CRM se setkáváme v oblasti IT již několik let. Díky vysokému počtu dodavatelů těchto řešení je možné vybírat z široké nabídky CRM aplikací. Role informačních technologií v cílovém CRM konceptu je definována především jako podpora a automatizace celého CRM procesu, který standardně začíná získáním znalostí o klientech (data warehouse), pokračuje detailní analýzou jejich potřeb a vzorů chování, tj. přeměnou dat na informace (business intelligence & analytical CRM) a následně umožňuje využití těchto informací k efektivním a personalizovaným interakcím s klienty (operational CRM) všemi distribučními a komunikačními kanály (collaborative CRM).

[\[14\]](#)

2.5.5. Problémy při zavádění CRM

Mnoho společností sice má CRM, ale pod tímto pojmem využívá pouze částechy skutečné databáze a leckdy ani nemají tušení, jak CRM využít co nejefektivněji a nejlevněji. Zásadní otázka je právě zavádění do praxe. Problém v řadě případů spočíval v tom, že došlo k základnímu nepochopení podstaty CRM. Zaměstnanci nebyli schopni změnit své zavedené pracovní postupy, jediný rozdíl spočíval v tom, že aplikovali novou podporu informačních technologií (IT). Vedení firem nebylo do věci zapojeno, pouze pověřilo oddělení informačních technologií, aby se zabývalo problematikou CRM.

[\[15\]](#)

2.5.6. Strategie zavádění CRM

Na začátku výběru CRM není rozumné a ani efektivní zakoupit CRM vyvinuté pro segment velkých korporací a čekat, že díky omezení několika funkcí aplikace naplní potřeby dané firmy. Je také přežitkem domnívat se, že existuje nějaké univerzální CRM, které by splnilo potřeby všech menších a středních organizací. Naopak. Stále více se setkáváme s tím, že víc než podle velikosti se tyto systémy dělí podle odvětví, pro které jsou určeny.

První věcí, které by se měli zájemci vyhnout, je nepotřebná komplexnost a složitost řešení. U CRM je jednou z nejdůležitějších věcí hladká integrace do stávajících podnikových systémů. Pro co možná nejjednodušší nasazení CRM je vhodné volit takové, které dodavatel na základě detailních znalostí daného odvětví vyvinul specificky právě pro něj. Čím blíže bude řešení CRM konkrétnímu oboru, tím snadnější bude jeho nasazení a spuštění.

Výběr vhodného CRM informačního systému pro daný subjekt je individuální proces závislý na mnoha objektivních i subjektivních faktorech. Obecně lze však doporučit jak postupovat. Při výběru vhodného CRM informačního systému je potřeba si položit několik důležitých otázek.

1. Jaké má subjekt zkušenosti s problematikou CRM a příslušnými CRM IS?

Zavádění CRM systému bez propracované CRM informační strategie je nejen časovou, ale především finanční ztrátou. Jestliže subjekt nemá žádné či velmi slabé zkušenosti s problematikou CRM, lze jednoznačně doporučit využít služeb externího poradce. Spolupráce s ním může probíhat na poradenské či komplexně dodavatelské úrovni.

2. Jaký je k dispozici rozpočet?

Pokud firma nemá dostatek finančních prostředků pro nákup, implementaci, následný provoz a údržbu rozsáhlého CRM řešení, je pochopitelně vhodné použít alespoň jednodušší variantu CRM informačního systému (především primárně moduly pro operativní činnosti řízení vztahů se zákazníky). Subjekt tímto získá provozní zkušenosti s CRM a vzhledem k tomu, že většina CRM informačních systémů je stavěna modulárně, má možnost následně systém rozšiřovat o další CRM funkcionality.

3. Jak velká je cílová skupina?

Z hlediska požadavků na výkonnost CRM IS je důležité znát, v jakém řádu (stovky, tisíce, desetitisíce, atd.) se bude pohybovat počet oslovovaných zákazníků firmy. Obecně platí pravidlo, že čím větší počet potenciálních zákazníků, tím rozsáhlejší a sofistikovanější CRM IS musí být. Rostou také požadavky na systémovou integraci s dalšími podnikovými aplikacemi a hierarchii přístupových práv k datovým zdrojům na jednotlivých úrovních řízení.

4. Existuje na trhu CRM informační systém pro konkrétní firemní odvětví?

Vzhledem k tomu, že existuje celá řada oborů (bankovníctví, telekomunikace, farmacie atd.), kde má využívání CRM dlouholetou tradici, je přínosné si při hledání optimálního CRM IS položit také tuto otázku. Díky tomuto faktu jsou totiž na trhu k dispozici specializované CRM informační systémy, či alespoň jejich moduly. Je vhodné proto zvážit, zda existující systémy nebudou mít pro firmu kvalitativně vyšší přidanou hodnotu.

5. Automatizační činnosti by měly probíhat na úrovni operativní či analytické?

V tomto bodě musí management firmy rozhodnout, zda je pro podnik dostačující systém disponující funkcionalitou pro management času, správu osob a kontaktů (operativní CRM), či zda je důležitá také analytická část systému – potřeba roste především s množstvím potenciálních zákazníků (analytické CRM).

6. Jaká bude vazba CRM IS na další firemní informační systémy?

Posledním krokem rozhodovacího procesu je otázka včasné a promyšlené systémové integrace CRM informačního systému na další podnikové systémy. Včasná analýza této problematiky zamezí celé řadě následných integračních problémů. Čím více je CRM informační systém otevřený, tím je vhodnější pro jakoukoliv budoucí konfiguraci propojení firemních informačních systémů.

[\[16\]](#)[\[17\]](#)

3. Informační systém Mstormm

3.1. Cíle aplikace

Cílem práce bylo vytvořit přehledný informační systém pro realitní makléře, který jim bude schopen poskytnout kdykoliv a kdekoliv pro ně důležité informace prostřednictvím mobilních telefonů. Konkrétně bylo nutné makléřům zprostředkovat informace týkající se přehledů nabídek a poptávek nemovitostí a dalších bližších parametrech těchto záznamů, jako je například popis, lokalita, jméno makléře starajícího se o danou zakázku, atd. Dále se jednalo o různé upomínky a vzkazy makléřům týkající se jednotlivých zakázek. Tyto informace jsou velice důležité pro maximální efektivitu práce makléřů a jejich profesionální jednání s klienty.

Základním požadavkem na systém tedy bylo, aby rychle, snadno a přehledně zpřístupnil požadovaná data. K splnění tohoto cíle tedy bylo nutné do informačního systému začlenit vhodné filtry pro výběr pouze tížených dat. Výsledky tohoto snažení jsou představeny v kapitole zabývající se seznámením se samotným informačním systémem, který byl pojmenován Mstormm.

3.2. Použité technologie a jejich aplikace

3.2.1. Úvod

Pro vytvoření webových stránek není potřeba kupovat žádné speciální programy. Na vytvoření jednoduché webové stránky stačí obyčejný textový editor pro zápis popisného jazyka (X)HTML, avšak to k vytvoření plnohodnotných interaktivních stránek, které jsou schopné dynamicky reagovat na požadavky od klienta, samozřejmě nestačí. Pro vytvoření takové webové aplikace musí vývojář zapojit další nástroje.

3.2.2. Apache

Pro možnost rozumného vývoje webové aplikace je dobré mít na svém PC nainstalovaný nějaký softwarový webový server, na kterém může vývojář testovat vznikající webové stránky skrze internetový prohlížeč na adrese <http://127.0.0.1/> (localhost) před jejich odesláním na server určený pro zpřístupnění aplikace veřejnosti. Mezi takové softwarové webové servery patří i Apache HTTP Server. K výhodám Apache patří zejména dostupnost pro všechny hlavní platformy (Windows, Linux, atd.) a také fakt, že Apache je k dispozici zdarma.

3.2.3. PHP a Zend Framework

Samotná aplikace byla nakonec převážně tvořena v objektové nadstavbě PHP nazvané Zend Framework často označované jako ZF. PHP je skriptovací programovací jazyk, který se stal velmi oblíbený především díky jednoduchosti použití. PHP se nejčastěji začleňuje přímo do struktury jazyka (X)HTML, jeho skripty jsou prováděny na straně serveru a k uživateli je přenášén pouze výsledek jejich činnosti. PHP jazyk je nezávislý na platformě, skripty tedy fungují bez větších úprav na mnoha různých operačních systémech. PHP také podporuje přístup k většině databázových systémů (např. MySQL, Oracle, PostgreSQL).

Platforma ZF byla vybrána jako nejlepší volba společně s konzultantem z firmy Digital Solutions Ing. Ondřejem Říhou a to především z důvodu vývoje mateřského informačního systému určeného pro osobní počítače právě v tomto webovém aplikačním frameworku, který se ukázal jako velice silný nástroj pro tvorbu webových aplikací. Dalším důvodem pak byla snadnější případná budoucí správa a údržba předaného systému samotnými programátory z firmy Digital Solutions, kteří mají se Zend Frameworkem již bohaté zkušenosti.

Mezi hlavní výhody tohoto frameworku patří zjednodušování a sjednocování zpracování úloh, které se v PHP často řeší. Frameworky se snaží poskytnout programátorovi nový efektnější způsob řešení a také mu pomoci ušetřit mnoho kódu. ZF užívá modulární architektury, která umožňuje vývojářům použít jen ty komponenty, které potřebují. Částečné závislosti mezi komponentami však existují. ZF v sobě zahrnuje komponenty pro MVC aplikace, autorizaci a autentizaci, implementuje různé druhy cache, filtrů a validátorů pro uživatelská data, jazykové komponenty a mnoho dalších.

Zend Framework je postaven na návrhovém vzoru MVC neboli model, view (pohled), controller. Je to tedy objektová struktura, jež odděluje data, jejich zpracování a následné zobrazení. Model jsou data, která se vypisují na stránce. Úkolem pohledu je prezentovat model ve vhodné formě klientovi. Můžeme tedy mít jeden model a několik různých pohledů. Jedem pohled budeme například používat pro mobilní telefony a jiný pro klienty, kteří používají osobní počítače. Pohled je tedy nejčastěji (X)HTML stránka. Controller je pak část návrhového vzoru MVC, která se stará o sestavení modelu a výběr pohledu pro požadavek klienta. Typickými operacemi patřícími do popisu controlleru jsou:

- na základě URL určit, která část kódu bude obsluhovat daný požadavek
- přístup k datové vrstvě modelu a provádění vytvoření modelu
- předání modelu příslušnému pohledu a jeho vygenerování

Výhodou tohoto návrhového vzoru je fakt, že se jednotlivé části chovají jako zapouzdřené komponenty. To znamená že:

- pohled neví o tom, odkud a jakým způsobem byl vygenerován model
- controller se nestará o to, jakým způsobem budou data prezentována v pohledu, pouze se postará o jeho výběr
- model ani pohled nemusí mít informace o tom, jaký požadavek klient odeslal

[\[18\]](#)[\[19\]](#)

3.2.4. MySQL

Aplikace pracuje se vzdáleně uloženými daty v databázi, která byla vytvořena v databázovém systému MySQL. Komunikace s databází probíhá pomocí jazyka SQL. Pro svou snadnou implementovatelnost (lze jej instalovat na Linux, MS Windows, ale i další operační systémy), výkon a především díky tomu, že se jedná o volně šiřitelný software, má vysoký podíl na v současné době používaných databázích.

[\[20\]](#)

3.2.5. JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, který se nejvíce používá jako interpretovaný programovací jazyk pro WWW stránky, často vkládaný přímo do (X)HTML kódu stránky. JavaScript se používá pro oživení stránek a jsou jím obvykle ovládány interaktivní prvky GUI. Program psaný v JavaScriptu se spouští až po načtení celé WWW stránky a to na straně klienta (je interpretován pomocí webového prohlížeče). Z toho plynou určité komplikace. Jednotlivé implementace JavaScriptu se v různých prohlížečích mírně liší, a proto není podmínkou, že skript, který pracoval bezchybně v jednom prohlížeči, bude plnit svoji funkci i v jiném. Dále také existují prohlížeče, které JavaScript nepodporují nebo uživatelé, kteří si JavaScript zakazují, proto není dobré se na něj plně spoléhat. Přesto aplikace JavaScript využívá a to v podobě frameworku jQuery.

[\[21\]](#)

3.2.6. jQuery

jQuery je JavaScriptový framework, který umožňuje snadno vyhledávat elementy DOMu⁶, modifikovat je, i vytvářet nové. Stejně tak je jednoduché vytváření nových HTML elementů. Stačí zadat HTML kód a jQuery sám vytvoří patřičnou strukturu DOMu. Framework jQuery ale nabízí i další možnosti usnadnění práce - umí pracovat s událostmi, nabízí pokročilé funkce pro práci s poli, nechybí ani podpora technologie Ajax a animací.

Tento framework jsem si vybral především z těchto důvodů:

- je malý

Zkomprimovaná verze má pouze 20 kB (nezkomprimovaná pak 61 kB). Ke klientu se navíc přenáší pouze jednou a při dalších požadavcích se bere z cache prohlížeče.

- podpora všech prohlížečů
- velká komunita a aktivní vývoj nových verzí
- bohaté zdroje o jQuery na internetu

V aplikaci využívám tuto knihovnu v několika případech. Prvním případem je dynamické vložení položek do HTML formulářových objektů *select*, tedy modifikace elementů DOM. Druhým případem je právě podpora práce s technologií Ajax, kterou aplikace také používá. Přesnější informace jsou popsány v následující podkapitole věnované přímo této technologii.

Dále je výhod této knihovny využito při kontrole či případném automatickém doplnění nebo upravení zadaných hodnot do (X)HTML formulářových prvků *text* ještě před jejich samotným odesláním na server. Kontrola probíhá jako reakce na událost *change* daného prvku. Tedy po předání řízení prvku jinému. Samozřejmě, že není možné se na JavaScript plně spoléhat kvůli výše zmíněným faktům a plná validace tedy probíhá ještě jednou na straně serveru po odeslání dat. Přesto je zde tato předvalidace použita, aby uživatel zbytečně neodesílal na server nesmyslná data a nezdržoval tak sebe i server. Více pozornosti je tomuto tématu věnováno v kapitole představující samotnou aplikaci.

[22]

⁶ **DOM (Document Object Model)** je objektově orientovaná reprezentace XML nebo HTML dokumentu. DOM je rozhraní pro programování aplikací umožňující přístup či modifikaci obsahu, struktury, nebo stylu dokumentu, či jeho částí.

3.2.7. Ajax

Ajax neboli Asynchronous JavaScript and XML je obecné označení pro technologii vývoje interaktivních webových aplikací, které mění obsah svých stránek bez nutnosti jejich opětovného načítání. Díky této technologii je možné z webového prohlížeče komunikovat s webovým serverem a vyměňovat si s ním data, která se mohou ihned uložit na serveru, nebo naopak zobrazit u klienta. Mezi výhody patří odstranění nutnosti opětovného načtení a překreslení celé stránky při každé operaci, které jsou nutné u klasického modelu WWW stránek. Práce s Ajaxem má ovšem i své nevýhody. Kromě potíží při ladění aplikace Ajax například ke své funkčnosti nezbytně potřebuje (zapnutou) podporu JavaScriptu.

V aplikaci je tato technologie použita u filtrů pro zobrazení pouze požadovaných dat podle určitých parametrů, kdy bylo zapotřebí do (X)HTML formulářového objektu *select* dynamicky načítat položky podle zvoleného parametru v „*selectu*“ jiném. Tyto položky jsou vráceny ze serveru jako výsledek dotazu, který je na server odeslán právě díky technologii Ajax. Zároveň byla tato technologie použita k tomu, aby byl uživatel stále informován o počtu vyhledaných záznamů při odeslání požadavku na server s aktuálním nastavením filtrů. To znamená, že po jakékoliv změně v nastavení filtrů je uživateli systému okamžitě vrácena aktualizovaná hodnota počtu záznamů, které by byly vyhledány, aniž by bylo nutné znovu vykreslovat celý obsah stránky.

3.2.8. CSS

Pro hezké stylování stránek je důležité využívat Kaskádové styly (CSS). CSS je jazyk pro popis způsobu zobrazení stránek napsaných v jazycích (X)HTML nebo XML. Hlavním smyslem tohoto jazyka je umožnit návrhářům oddělit vzhled dokumentu od jeho struktury a obsahu.

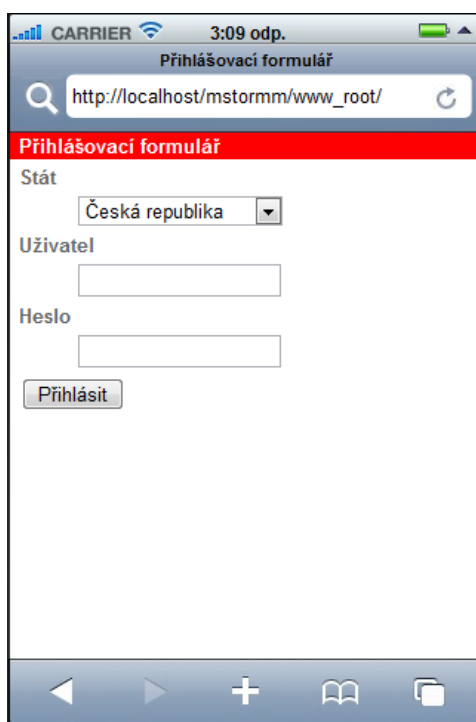
3.2.9. Emulátory mobilních zařízení

Pro snadnější vývoj aplikace, který vyžadoval testování vzhledu i funkčnosti, bylo také zapotřebí používat emulátory mobilních prohlížečů na PC. Ne ale pro každý mobilní prohlížeč je zdarma k dispozici emulátor, a pokud ano, ne vždy je snadné ho zprovoznit. Během vývoje jsem používal simulátor mobilního prohlížeče Opera Mini 4 pomocí nástroje MicroEmulator a simulátor mobilního telefonu Apple iPhone nazvaný MobiOne pro testování prohlížeče Safari Mobile.

MicroEmulator představuje Java simulátor mobilních telefonů, pomocí něhož testujeme aplikace tvořené v J2ME. Tento emulátor je na internetu k dispozici zdarma a bez nutnosti jakýchkoliv registrací. Tento nástroj nám také dovoluje simulovat velikost displeje telefonu nastavitelnou dle vlastní potřeby v pixelech. Dalším krokem k začátku testování stránek pomocí MicroEmulatoru bylo stažení samotné Opery Mini. Na jejich mateřských stránkách lze snadno a zdarma stáhnout potřebný instalační soubor a ten otevřít ve zmiňovaném emulátoru. Druhým použitým nástrojem byla aplikace MobiOne. Tato aplikace je opět k sehnání zdarma a bez nutnosti registrací.

3.3. Seznámení s prostředím aplikace

V této části práce bych čtenářům rád představil vzniklý informační systém a základní orientaci v jeho prostředí. Po načtení úvodní stránky je uživateli zobrazen přihlašovací formulář se vstupními poli pro zadání jména a hesla. Systém je přístupný pouze makléřům společnosti Mmreality, kteří se tedy musí prokázat svým uživatelským jménem a zvoleným heslem. Během komunikace je klientské jméno i heslo šifrováno pomocí symetrické blokové šifry splňující standard AES⁷.



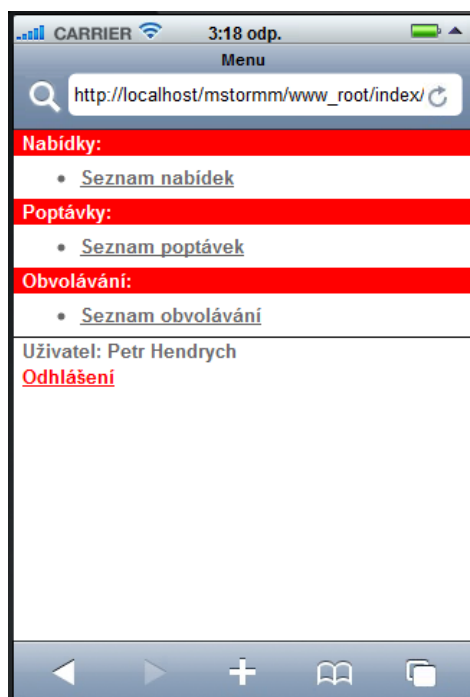
Obrázek 4. Safari Mobile 4.0 – Přihlašovací formulář

⁷ AES (Advanced Encryption Standard) je standard amerického úřadu pro standardizaci

V případě selhání ověření uživatele je ze serveru vrácen chybový kód, podle kterého je následně zobrazeno chybové hlášení. Možnosti jsou následující:

```
$errorCodes = array(
    "AE00"=>'Neznámá chyba v aplikaci',
    "AE01"=>'Nevyplněné jméno',
    "AE02"=>'Nevyplněné heslo',
    "AE03"=>'Uživatel nenalezen',
    "AE04"=>'Účet byl smazán',
    "AE05"=>'Účet je neaktivní',
    "AE06"=>'Platnost hesla vypršela',
    "AE07"=>'Není vyplněn KK makléře',
    "AE08"=>'Nevalidní email',
    "AE09"=>'Nevalidní telefonní číslo',
    "AE10"=>'Nevyplněno EIN',
    "AE11"=>'Nevyplněno číslo bankovního účtu',
    "AE12"=>'Nevyplněna adresa');
```

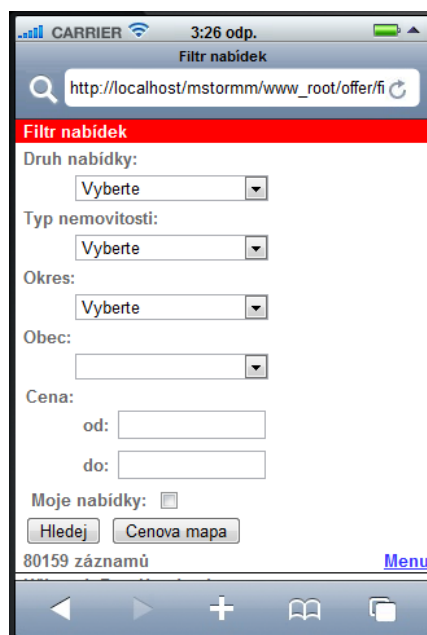
Po úspěšném přihlášení je do *session*⁸ uloženo jméno, příjmení a ID makléře. Na displeji telefonů je následně zobrazeno základní menu, kde si uživatel může zvolit jeden ze tří typů informací. Jde o položky „*Seznam nabídek*“, „*Seznam poptávek*“ a „*Seznam obvolávání*“. Dále je na všech stránkách v informační části zobrazeno jméno a příjmení přihlášeného makléře a odkaz na odhlášení ze systému, který provede vymazání *session* a uživatele přesměruje zpět na přihlašovací formulář.



Obrázek 5. Safari Mobile 4.0 – Menu

Po výběru jedné z možností je uživateli zobrazena stránka s filtry pro omezení množství vyhledaných dat. Další snímek zobrazuje možnosti filtrů pro výpis nabídek.

⁸ **Session** v informatice představuje permanentní síťové spojení mezi klientem a serverem.



Obrázek 6. Safari Mobile 4.0 – Filtr nabídek

Zde má uživatel možnost filtrovat záznamy podle následujících parametrů: „*Druh nabídky*“, „*Typ nemovitosti*“, „*Okres*“, „*Obec*“, „*Cena od*“, „*Cena do*“ a „*Moje nabídky*“. Typ nemovitosti nabízí možnosti „*Byty*“, „*Domy*“, „*Pozemky*“ a „*Komerční objekty*“ nebo potomky těchto možností. Položky v první skupině výběrových možností jsou brány jako nadřazené rodičovské elementy dalších podnabídek. Pokud tedy uživatel vybere možnost typu rodič (např. „*Byty*“) a odešle formulář, je mu na následující stránce nabídnut seznam pro možnost výběru jednoho, více nebo všech potomků. Následující obrázek zobrazuje nabídku při volbě typu „*Byty*“.



Obrázek 7. Safari Mobile 4.0 – Byty

Filtr „Okres“ umožňuje filtrovat nabídku v rámci určitého okresu České republiky. S výběrem okresu dojde zároveň i k načtení obcí do filtru „Obec“ spadajících pod tento okres. Díky tomu může uživatel dále zúžit výběr na nabídky pro konkrétní obec. Dále můžeme filtrovat nabídky podle určité cenové hranice a to použitím filtru „Cena od“, „Cena do“ nebo jejich kombinací. Posledním možným omezením výběru záznamů je položka „Moje nabídky“, která zúží množinu pouze na nabídky, které spadají pod přihlášeného makléře.

Všechny filtry se dají libovolně kombinovat a při jejich jakékoliv změně, je uživateli vrácena informace o počtu vyhledaných nabídek s daným nastavením. To bylo velice důležité z hlediska pohodlné ovladatelnosti, aby uživatelé nemuseli zbytečně listovat množstvím stránek s vyhledanými záznamy, než naleznou požadovanou položku. Bylo totiž nutné si uvědomit, že uživatelům budou zobrazeny vyhledané záznamy na displeji mobilního telefonu, jehož velikost neumožňuje zobrazení většího počtu vyfiltrovaných položek najednou a vyhledané záznamy tedy bude nutné pravděpodobně rozdělit do více stránek. Kromě těchto popsanych filtrů obsahuje stránka potvrzovací tlačítko pro odeslání formuláře a odkaz na návrat do hlavního menu.

Po odeslání formuláře je uživateli zobrazen buď výše zobrazený filtr pro výběr potomků rodičovské nemovitosti, nebo rovnou výsledek filtrování v případě, že uživatel volil nemovitost typu potomek nebo tento filtr vůbec neaplikoval.



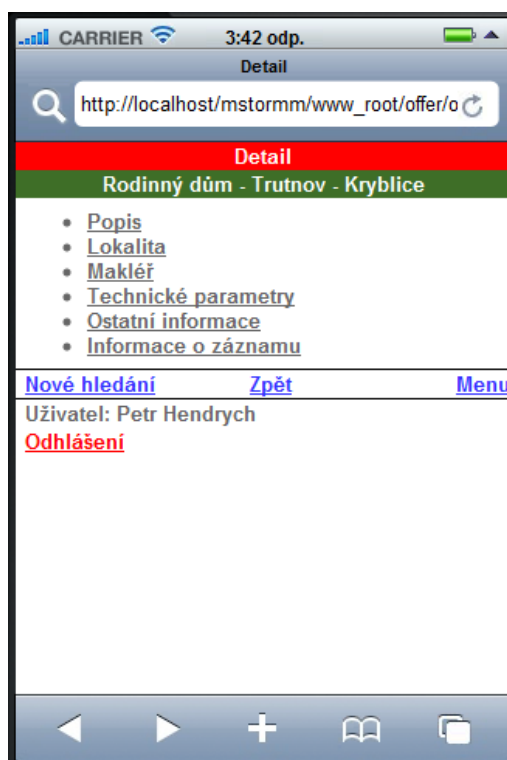
The screenshot shows a mobile browser interface with a status bar at the top displaying 'CARRIER', signal strength, Wi-Fi, time '3:39 odp.', and battery level. The page title is 'Nabídky'. Below the title is a search bar with the URL 'http://localhost/mstormm/www_root/offer/fi'. The main content is a table of real estate offers with the following data:

Typ nemov.	Tech. inf.	m²	Cena
Rodinný dům	-- --	300	3915000.00
Rekreační objekt	-- --	66	1150000.00
Byt 2+1	-- P OV	64	500000.00
Rodinný dům	-- --	548	2900000.00
Výrobní objekt	-- --	22661	4500000.00
Stavební parcela	-- --	2891	700000.00
Byt 2+1	-- C OV	6311	890000.00
Stavební parcela	-- --	3800	600.00
Rodinný dům	-- --	167	4900000.00
Rodinný dům	-- --	1550	2900000.00

Below the table, there is a pagination control showing '<<|< 1-10 ze 80159 >|>>'. At the bottom of the page, there are links for 'Nové hledání' and 'Menu', and a user login section showing 'Uživatel: Petr Hendrych' and a red 'Odhlášení' link.

Obrázek 8. Safari Mobile 4.0 – Nabídky

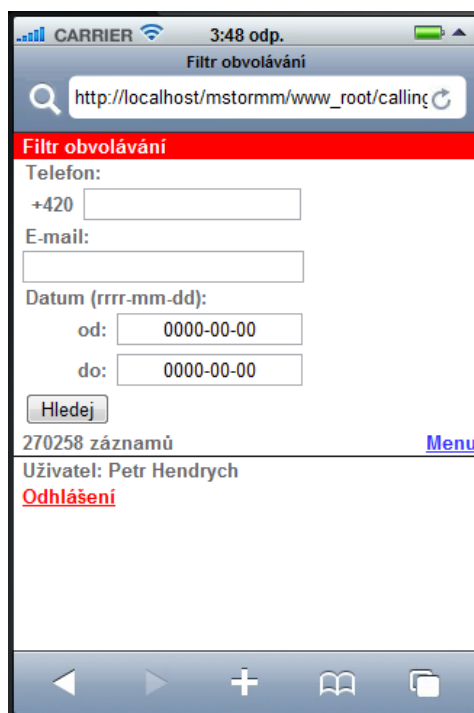
Vybrané záznamy jsou uživateli zobrazeny v tabulce, kde je vidět typ nemovitosti, její technické parametry, rozloha a cena. Typ nemovitosti u každého záznamu funguje jako odkaz na další stránku, kde jsou makléři nabídnuty detailnější informace o daném záznamu. Kromě zobrazených nabídek obsahuje stránka informaci o pořadí aktuálně zobrazených záznamů a jejich celkovém počtu. Dále umožňuje listovat mezi jednotlivými položkami. Uživateli je také nabídnuta možnost vrátit se zpět na stránku s výběrovými filtry pomocí odkazu „*Nové hledání*“ nebo do základního menu pomocí odkazu „*Menu*“.



Obrázek 9. Safari Mobile 4.0 – Detail záznamu

Nabídnuté informace nemusí být u každého záznamu stejné, proto je toto menu vygenerováno vždy dynamicky v závislosti na informacích, které jsou o dané položce v databázi k dispozici. Uživatel zde opět nalezne odkazy pro návrat na výběrové filtry a do základního menu. Dále je zde přidán odkaz „*Zpět*“, který uživatele vrátí na stránku s výpisem jednotlivých nabídek.

Způsob filtrování, výpisu a ovládání stránek s poptávkami se nijak neliší od způsobů představených u nabídek, proto se jimi nebudu dále zabývat. Poslední volbou v úvodním menu je „*Seznam obvolávání*“. Pod touto položkou nalezne uživatel filtry pro vyhledávání položek týkajících se obvolávání, tedy seznam klientů, které je potřeba z nějakých důvodů kontaktovat.



Obrázek 10. Safari Mobile 4.0 – Filtr obvolávání

Zde má uživatel možnost filtrovat záznamy podle následujících parametrů: „Telefon“, „Email“, „Datum od“ a „Datum do“. U filtrů „Telefon“ a „Email“ se jedná o kontaktní údaje na zákazníka, kdy makléři stačí zadat jakoukoliv část řetězce, kterou tyto údaje obsahují. Například pokud zadá do vstupního pole „Email“ text „petr“, jsou mu po odeslání požadavku vráceny všechny záznamy s emailem obsahujícím tento řetězec. Každá položka ze seznamu obvolávání má v databázi údaj o tom, v jakém časovém rozmezí by se měl zákazník kontaktovat. K omezení počtu vyhledaných položek pomocí tohoto parametru slouží filtr „Datum od“ a „Datum do“. Uživatel opět může používat libovolné kombinace všech filtrů a po změně jakéhokoliv z nich je informován o počtu navrácených položek. Kromě filtrů je i zde samozřejmě potvrzovací tlačítko pro odeslání formuláře a odkaz umožňující návrat do hlavního menu.

Obvolávání		
Telefon/email	Výsledek	Maklér
721480955 Marketadohn@seznam.cz	Vyřizuje se	Veronika Šlancarová
607870733	Volný inzerát	Veronika Šlancarová
721101944	Volný inzerát	Veronika Šlancarová
720309681	Volný inzerát	Veronika Šlancarová
721682093	Volný inzerát	Veronika Šlancarová
774602475	Volný inzerát	Veronika Šlancarová
608809447	Volný inzerát	Veronika Šlancarová
725809644	Volný inzerát	Veronika Šlancarová
602731071	Volný inzerát	Veronika Šlancarová

Obrázek 11. Safari Mobile 4.0 – Obvolávání

Po odeslání dotazu jsou makléři jednotlivé záznamy opět vypsány do tabulky, kde uživatel vidí kontaktní údaj na zákazníka, stavu pohledávky a jméno makléře, který záznam naposledy editoval. V této tabulce funguje jako odkaz „*Telefon/Email*“, pod kterým uživatel nalezne doplňující text k záznamu.

Text		
<p>N Byt 3+1 Hustopeče Cena: 1670000Kč 14.04.2010 <Zpět> - Prodám kompletně zrekonstruovaný byt 3+1 na výborném místě, ul.Na Sídlišti v Hustopečích, 5 min. od centra, hned vedle školka,dětské hřiště s pískovištěm, škola, atd. Nová zděná koupelna + wc,nová kuch.linka, podlahy, dlažby, dle dohody nový nábytek. Cena: 1.670 000,- Kč. Při rychlém jednání sleva!!!</p>		
Nové hledání	Zpět	Menu
<p>Uživatel: Petr Hendrych Odhlášení</p>		

Obrázek 12. Safari Mobile 4.0 – Text

3.4. Vzniklé problémy a jejich řešení

3.4.1. Různá velikost displejů mobilních telefonů

Velikosti displejů mobilních telefonů se citelně liší, což se ukázalo být velkým problémem při návrhu vzhledu stránek. Jako autor webových stránek nevím, jaký mobilní telefon bude makléř používat. Bylo potřeba zajistit, aby stránky vypadaly stejně úhledně na velkých i malých displejích, aby se nezměnilo rozložení formulářových prvků a hlavně, aby stránky byly stále dobře ovladatelné.

Nápadem na řešení tohoto problému bylo udělat stránky omezeně flexibilní tak, aby upravovaly šířku podle šířky prohlížeče, ale aby nad určitou šířku nešly. To umožňuje vlastnost kaskádových stylů *max-width*. Bohužel mobilní prohlížeče většinou tuto vlastnost nepodporují. Stránky byly tedy utvořeny tak, aby byly plně flexibilní s plovoucí šířkou, kdy se vždy zobrazí přes celou šířku prohlížeče. U tohoto řešení se stránky stávají nepřehledné pouze při zobrazení na PC z důvodu přílišného roztažení. Komplikace by zde nastaly, kdyby měly stránky obsahovat pevně široké prvky (reklamy, obrázky, složitější grafiku). Stránky byly ale od začátku koncipovány tak, aby byly jednoduché a žádný z těchto prvků obsahovat neměly.

Elastičnosti stránek je pak dosaženo tím, že šířka těl tabulek, do kterých jsou zobrazovány prvky stránek, je vždy nastavena na 100% šířky prohlížeče. Velikost okrajů a odsazení písma je nastavena v jednotkách *em*. *Em* je jednotka udávající velikost podle šířky velkého písmena „M“ a je dána operačním systémem. Jednotka *em* mění svou velikost v závislosti na aktuální velikosti písma. Aktuální velikost písma může být ovlivněna nadřazeným nastavením stylu (*font-size*), formátovacími značkami (*<big>* apod.) a také volbou uživatele. Většina prohlížečů umožňuje nastavit velikost zobrazovaného písma (zobrazit / velikost textu). To je i základní význam těchto jednotek - umožňují design, který se při změně tohoto nastavení nezhroutí.

Problémem při zobrazování stránek byl Apple iPhone, kde se stránky zobrazovaly ve své zmenšenině bez ohledu na nastavení velikosti písma a dalších parametrů. Pro odstranění tohoto problému bylo potřeba nastavit takzvaný *viewport*. *Viewport* určuje, jak bude stránka při zobrazení přiblížena či oddálena a jak na ní bude zalomen text. Nastavení „*viewportu*“ se provádí vložením příslušného meta-tagu do hlavičky stránek.

```
<meta name = "viewport" content = "width = device-width" />.
```

[23][24]

3.4.2. Návrh SQL dotazů

Součástí práce nebylo navrhnout a vytvořit databázi, ta byla již hotová, místo toho mě čekal náročný úkol se v této databázi zorientovat a pochopit vzájemné vazby mezi jednotlivými tabulkami. Jedná se totiž o databázi pro rozsáhlý informační systém obsahující na desítky provázaných tabulek (mně osobně jich bylo zpřístupněno kolem sedmdesáti a to jsem pro svoji práci nepotřeboval veškeré entity, které databáze obsahuje). Některé z tabulek pak obsahují až tisíce záznamů. Pro práci s tak velikou databází bylo potřeba dokonale ovládnout syntaxi jazyka MySQL, kdy aplikace pro vyhledání požadovaných dat používá rozsáhlé dotazy obsahující například vnořený *select* (tzv. *subselect*), větvení dotazu pomocí příkazu *if*, spojování mnoha tabulek a vytváření tak složitých relací v jednom dotazu přes cizí klíče tabulek.

3.4.3. Ajax – synchronní vs. asynchronní přístup

Jako jeden ze způsobů využití technologie Ajax jsem uvedl vrácení počtu záznamů pro aktuální nastavení filtrů vyhledávání. Při využívání této funkce jsem si všiml nestandardního chování, které se projevovalo vrácením stejného počtu záznamů i při změně nastavení filtru „Okres“, které mělo dva důsledky. Prvním byla změna položek ve filtru vypisujícím obce spadající pod daný okres, druhým pak byla změna informace o počtu vrácených záznamů. Při pročitání zdrojového kódu a zvažování různých možností příčiny této nepříjemnosti se ukázalo, že příčinou vzniku této anomálie může být jediné nesynchronní vykonávání příkazů. Po provedené analýze vlastností Ajaxu jsem zjistil, že může pracovat ve dvou režimech a to v režimu synchronního a asynchronního vykonávání příkazů. Ajax je totiž defaultně nastaven na režim asynchronní, což znamená, že se mohou vykonávat další příkazy dříve, než jsou vrácena data ze serveru. To mělo za následek, že SQL dotaz na vrácení počtu vyhledaných položek se odeslal dříve, než byly načteny položky do filtru umožňující výběr obce, změněna jeho hodnota a tedy i hodnota parametru pro vrácení počtu záznamů. Problém vyřešilo nastavení Ajaxu do režimu synchronního. Synchronní komunikace způsobí pozastavení běhu programu až do dokončení přenosu dat. Tím bylo zajištěno, že se nejprve načtou nové položky do filtru, změní se jeho hodnota a až následně bude odeslán SQL dotaz. V tomto případě použití zkratky AJAX není správné, korektní by bylo SJAX: *Synchronous JavaScript and XML*.

[25]

3.4.4. JavaScript

Použití JavaScriptu se ukázalo v některých prohlížečích jako problémové. Zde se jednalo například o funkčnost Ajaxu. Ten však při finálním testování nepracoval jen ve zlomku z důležitých testovaných prohlížečů. Navíc by šla jeho funkce jen velice těžko obejít jinou cestou při zachování stejně jednoduché ovladatelnosti aplikace, proto tento nedostatek nebyl řešen.

Další překážkou, na kterou jsem narazil, byl odkaz zpět, který byl řešen pomocí JavaScriptu.

```
<a class="link" href="previous.html" onClick="history.back();return false;">Zpět</a>
```

Tento příkaz nebyl funkční v prohlížeči Safari v situaci, kdy předchozí stránka využívala metodu POST k odeslání dat z formuláře a při jejím obnovení bylo nutné opětovné odeslání dat touto metodou, proto byl příkaz v těchto případech nahrazen přesměrováním na požadovanou stránku pomocí PHP.

Například:

```
<?php echo"<a class=\"link\" href=\"\".$this->url(array('controller' => 'demand', 'action' => 'filterdemand', 'page' => $this->page, 'id' => null)).\"\">Zpět</a>"; ?>
```

Příslušná data pak byla uložena v *session*, odkud je stačilo načíst.

3.4.5. JavaScript – datum

V aplikaci jsem se snažil využít i jQuery plug-in s názvem Datepicker, což je snadno implementovatelná „*javascriptová*“ komponenta pro volbu data pomocí kalendáře. Tato komponenta se ovšem ukázala jako příliš náročná pro mobilní webové prohlížeče, ve kterých nefungovala korektně, a proto byla nahrazena jednodušším řešením. V současné době musí uživatel zadat datum do příslušného pole ručně. Tvar zadání data je RRRR-MM-DD. Jeho zadání je však uživateli usnadněno předvyplněním polí hodnotou 0000-00-00, tak aby uživatel nemusel zadávat znak „-“ a číselné zadání si pouze upravil do správného tvaru. Na událost *change* pole pro zadání data je JavaScriptem zkontrolován jeho správný tvar a případně je upraven. Pokud je po změně pole druhá číslice udávající století stále nula, je první číslice udávající tisíciletí automaticky změněna na 2 (předpoklad roku 20XX). Další dvě číslice jsou ponechány.

Pokud je ale druhá číslice udávající století větší než nula, jsou první tři číslice udávající rok změněny na 199 (předpoklad roku 199X). Poslední číslice je ponechána. U měsíce je pak kontrolováno dodržení rozsahu 1 až 12. Pokud je překročena spodní hranice, je číslo měsíce změněno na 01. Pokud je naopak překročena horní hranice, je měsíc upraven na 12. Den je pak kontrolován opět dodržením rozsahu 1 – X (X záleží na zadání měsíce) a v případě jeho nedodržení následuje úprava v závislosti na překročené hranici. Pokud uživatel zanechá pole pro zadání data beze změny, nejsou tyto filtry brány při vyhledávání v potaz.

3.4.6. Zajištění bezpečnosti dat

Každý systém může být napaden na mnoha úrovních: od operačního systému serveru až po prohlížeč uživatele. Já jsem řešil problémy, které jsou způsobeny interakcí webové aplikace a prohlížeče. Na této úrovni bylo potřeba zajistit zabezpečení aplikace proti několika typům útoků. Bezpečnost webových aplikací je mnohdy podceňovaná, přitom tyto aplikace mohou sdílet velmi důvěrná data. Proto představím nejznámější typy útoků a možností jejich obrany.

3.4.7. Řešené typy útoků

Krádež session

Protože je HTTP bezstavový protokol, ale často je potřeba alespoň určitou formu stavu přenášet, používá se mechanismus *session*. *Session* dává webovému serveru možnost uložit si libovolné (většinou však ne příliš obsáhlé) informace o uživateli, kteří k němu přistupují, a to o každém zvlášť. Uživateli je při prvním přístupu vytvořeno *session* a náhodně vygenerována jeho jednoznačná identifikace (pojmenovaná SESSIONID, SESSID, SID apod.). Jednoznačná identifikace uživatele je pak buď uložena v HTTP *cookie*⁹ nebo přenášena jako parametr URL. V dnešní době se spíše používá první metoda. Podle takto uloženého HTTP *cookie* pak server najde potřebné informace, například o přihlášeném uživateli, jeho úrovni přístupu a podobně. Podstatné je, že samotná data se již nepřenášejí. Mechanismus *session* je však možné zneužít. Stačí získat identifikátor *session* a následně můžeme předstírat korektně

⁹ Cookies jsou **malé textové soubory**, představují mechanismus pro ukládání informací na počítači klienta.

přihlášeného uživatele. Zabezpečení *session* funguje na tom principu, že přístup k proměnným konkrétního uživatele je možný přes jeho identifikátor v podobě dostatečně dlouhého, náhodně generovaného, a tedy neuhodnutelného klíče. Druhé specifikum je to, že *session* má poměrně krátkou dobu platnosti (řádově několik desítek minut). Samotné použití *session* ovšem automaticky nezaručuje, že se k datům pomocí nich uložených dostane pouze uživatel, jemuž jsou určeny. Webové aplikace – pokud *session* využívají – by současně s ním měly zavést konkrétní principy, které riziko zneužití *session* minimalizují.

Ochranu *session* jsem řešil následovně. *Session* jsou ukládány do samostatné složky, která není přístupná jiným PHP aplikacím a uživatelům. Název *session* je volen tak, aby byl unikátní v rámci všech PHP aplikací sdílejících stejnou doménu. Dále je použito nastavení *use_only_cookies = on*. To zaručí, že se budou identifikátory *session* ukládat vždy jako *cookies*. Nevýhoda tohoto přístupu spočívá v tom, že prohlížeč musí mít povoleno přijímat *cookies*, jinak nebudou *session* fungovat. Výhoda je ta, že to je mnohem bezpečnější než posílat identifikátory *session* v URL. Dalším typem útoku je *Session Fixation*. Jedná se o útok, kdy útočník klientovi nastaví nějakou hodnotu *Session ID* (podstrčením odkazu na webu, v e-mailu nebo např. přímou editací uživatelských *cookies*) a jakmile se uživatel přihlásí, tak tuto *Session ID* použije pro sebe. Obrana proti tomuto typu útoku je ve frameworku Zend poměrně jednoduchá – stačí před prováděním citlivých operací, jako je přihlašování, zavolat funkci *Zend_Session::regenerateId()*, která způsobí změnu *Session ID*, takže původní ID útočník nebude moci využít.

[\[26\]](#)[\[27\]](#)

Injection a SQL Injection

První typ útoku (injection) využívá špatného ošetření vstupu od uživatele při sestavování dotazů a parametrů do jiných systémů. Nejčastějším typem tohoto útoku je SQL Injection – špatná obsluha parametrů při tvorbě SQL dotazů.

Ukázka špatného použití (proměnná *param* obsahuje text ze vstupu):

```
String query = "select * from tabulka where field = $param";
```

Pokud však uživatel do *param* umístí např. hodnotu `'' or 1=1 or field = ''`, může tak přechíst všechna data v celé databázi, spojením totiž vznikne dotaz:

```
select * from tabulka where field = '' or 1=1 or field = ''
```

Případně díky „*subselectům*“ či „*joinům*“ lze přechít i data z jiných tabulek. Tento problém má naštěstí snadné řešení – sestavovat SQL dotaz pomocí jednoduchého skládání řetězců. V případě použití Zend frameworku existuje více možností, jak lze sestavit SQL dotaz bezpečně. Já osobně jsem při vytváření dotazů volil využití takzvané otazníkové notace a následné sestavení dotazu pomocí příkazu *quoteInto*, kdy je následně vstupní proměnná ošetřena tak, aby jejím vložením nemohl vzniknout útok typu SQL Injection.

Ukázka:

```
if ($estate_id>0){  
    $condition = ' AND d_id IN (SELECT det_d_id FROM  
mm_demands_estate_types_cz where'. $db->quoteInto(' det_et_id = ?',  
$estate_id).')';  
    $string = $string.$condition;  
}
```

[\[28\]](#)[\[29\]](#)

Cross Site Scripting (XSS)

Hojně používaná metoda útočníků, kdy pomocí JavaScriptu dokáží ovládnout celý systém webové aplikace. Tato metoda těží z nedbalosti programátorů, kteří nedokonale zabezpečí vstupy a výstupy uživatele a dovolí mu vytvořit kód JavaScriptu a ten navíc ještě na svých stránkách spustit. Útočník tak může ovládat aplikaci a vytvářet podvržené stránky nic netušícím uživatelům. Co je však ještě horší, je možné tímto způsobem zjistit *SESSION ID* (jedinečný identifikátor uživatele) a následně se za tohoto uživatele vydávat. Je nutné si uvědomit, že veškeré vstupy uživatele mohou být potencionálně nebezpečné. Vývojář proto musí veškeré vstupy uživatele kontrolovat a filtrovat na základě toho, co chce od uživatele získat. Pokud od uživatele žádáme jméno a heslo pro přihlášení, určitě mu nedovolíme, aby do políček zadal nějaký „*javascriptový*“ kód. Pokud toto udělá, musí aplikace tento kód odstranit a nedovolit jeho spuštění. Toho můžeme docílit například tím, že od zadaných hodnot odstraníme veškeré HTML značky (tím odstraníme i HTML značky pro uvozování „*javascriptových*“ kódu `<SCRIPT></SCRIPT>` a znemožníme spuštění kódu).

Vstupy od uživatelů kontrolují na dvou úrovních. První úroveň je kontrola zadání pouze povolených znaků pomocí JavaScriptu hned po té, co uživatel zadá data do textového pole. Kontrola je provedena jako reakce na událost *change* příslušného formulářového prvku a povolené znaky jsou závislé na typu vstupního pole. Pokud se jedná o vstupy určené k filtrování položek podle telefonního čísla nebo ceny, tedy

o vstupy kde si uživatel vystačí s množinou znaků omezenou jen na čísla, jsou všechny ostatní znaky z těchto formulářových prvků odstraněny. Pokud se jedná o vstup určený pro zadání emailu, kde je spektrum povolených znaků mnohem širší, jsou z tohoto pole odstraněny pouze znaky, které uvozují, nebo ukončují HTML tagy. Jde tedy o znaky “<” ”/” ”>”, u kterých bych nepředpokládal, že budou použity v žádné reálné emailové adrese. Posledním typem jsou formulářová pole určená pro zadání data. Formát data je jasně vyobrazen vedle vstupního pole, kdy je po uživateli vyžadován tvar RRRR-MM-DD. V tomto případě jsou pak ze vstupu odstraněny všechny znaky kromě čísel a znaku “-”. K odstraňování nepovolených znaků je používáno regulárních výrazů, které značně usnadňují tuto práci.

Kontrola pomocí JavaScriptu ovšem není dostačující, protože může být v řadě zařízení nefunkční, případně zakázána samotným uživatelem. Z tohoto důvodu byla data validována ještě jednou a to na straně serveru. U vstupů omezených pouze na čísla je použit filtr *Digits*, který propustí pouze číselné hodnoty. Na vstup pro zadání emailu či data jsou použity filtry *StripTags* (odstranění HTML tagů) a *StringTrim* (odstranění mezer na začátku a konci řetězce).

[\[28\]](#)[\[30\]](#)[\[31\]](#)

3.4.8. Šifrování přenášených dat

Šifrování je nepostradatelnou součástí zabezpečení webových aplikací. Jednou z nejznámějších metod zabezpečeného přenosu dat pomocí je SSL (Secure Sockets Layer). SSL je vrstva/protokol zabezpečující data na přechodu mezi aplikační a transportní vrstvou (protokolem TCP/IP). Zde lze zajistit šifrování přenášených dat a autentizaci serveru pomocí digitálních certifikátů. Oproti klasickému protokolu HTTP se bude zabezpečený lišit jen o písmeno “s” tedy HTTPS. Přes tento zabezpečený protokol jsou přenášena data i v případě informačního systému Mstormm. Pro použití SSL je třeba mít na straně serveru nainstalovanou podporu SSL a také jej musí podporovat prohlížeč na straně uživatele. SSL spojení funguje na principu šifry, kdy každá z komunikujících stran má dvojici šifrovacích klíčů (veřejný a soukromý). Veřejný klíč je možné zveřejnit. Pokud tímto klíčem kdokoliv zašifruje nějakou zprávu, je zajištěno, že ji bude moci rozšifrovat jen majitel použitého veřejného klíče svým soukromým klíčem. Při autentizaci ověřujeme pravost klienta, s nímž komunikujeme.

[\[32\]](#)

3.4.9. Vyhodnocení zabezpečení aplikace

Přestože existují další typy útoků, jako například Cross Site Request Forgery nebo Clickjacking, které nebyly řešeny, je zabezpečení aplikace dostatečné. Nejnebezpečnější typy útoků, které potenciálně hrozily, byly ošetřeny a další nebylo z povahy samotné aplikace třeba řešit.

3.5. Testování aplikace

Vzniklá aplikace byla následně otestována na všech šesti výše zmíněných a v dnešní době nejrozšířenějších mobilních prohlížečích, pro které jsem provedl několik testů. Během testování jsem se zaměřil především na tyto parametry:

- zobrazení
- rychlost komunikace (všechny naměřené časy byly zaokrouhleny na celé sekundy)
- funkčnost aplikace (problém JavaScriptu)

Aby se zamezilo vlivům kolísání signálu mobilních sítí, bylo provedeno více měření a v testu uvádím nejčastěji vyskytující se hodnoty. Všechny testy byly prováděny v lokalitě kolejí Harcov na připojení 3G (UMTS/ HSDPA).

3.5.1. Přehled výsledků

Prohlížeč	Typ telefonu	OS	Vzhled	Funkčnost	Čas1	Čas2	Čas3
Opera Mini 4.0	Nokia 5800	Symbian 9.4	Dodržení všech nastavení	100%	5	7	5
Opera Mini 5.0	Sony Ericsson C510	Sony Ericsson Java Platform 8	Dodržení všech nastavení	100%	5	5	4
Opera Mobile 9.5	HTC Touch Diamond	Windows Mobile 6.1	Nedodržení některých nastavení**	Nefunkční ajax	6	4*	4
Opera Mobile 10	Samsung Omia 2 I800	Windows Mobile 6.5	Dodržení všech nastavení	100%	6	6	4
Nokia Mini Map Browser 7.1	Nokia 5800	Symbian 9.4	Dodržení všech nastavení	100%	6	7	5
Safari 4	iPhone	OS X	Dodržení všech nastavení	100%	7	10	6
NetFront 3.3	Sony Ericsson C510	Sony Ericsson Java Platform 8	Dodržení všech nastavení	Nefunkční ajax	6	6*	5
Internet Explorer Mobile 6	Samsung Omnia 2 I800	Windows Mobile 6.5	Dodržení všech nastavení	100%	6	8	6

Tabulka 6. Přehled výsledků testování aplikace Mstormm

Čas1 - přihlášení do systému (ověření uživatele)

Čas2 - vykreslení formuláře s filtry

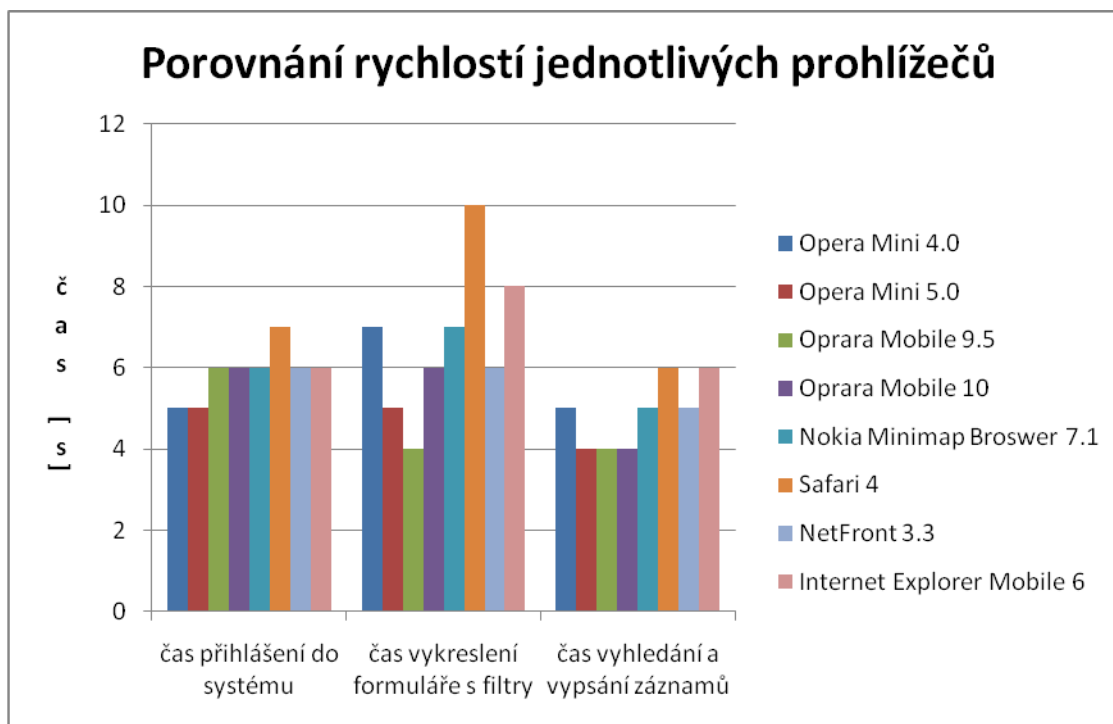
Čas3 - vyhledání a vypsání záznamů

** irelevantní čas z důvodu nefunkčního Ajaxu (neodesílá se dotaz pro informaci o počtu vyhledaných záznamů)*

*** ignorování některých CSS (nastavení velikosti prvků select a nastavení vzhledu tabulky)*

3.5.2. Vyhodnocení

Všechny testy byly provedeny na reálných telefonech, aby měření odpovídala skutečnosti. V některých případech však byly fotografie telefonů nahrazeny „PrintScreenem“ emulátorů prohlížečů z důvodu lepší kvality vzniklého snímku. Na jednotlivých snímcích je vidět, jak různé prohlížeče interpretují HTML tagy, kdy se jejich zobrazení téměř vždy mírně liší. Rychlosti komunikace jsou pak velice podobné. Nejdéle zobrazuje stránky prohlížeč Safari Mobile. V testu dopadla nejhůře Opera Mobile 9.5, která nedodržela nastavení kaskádových stylů (CSS). Navíc v ní nebyl funkční JavaScript, který je pro aplikaci velice důležitý. JavaScriptové kódy se ukázaly nefunkční i v prohlížeči NetFront 3.3, přestože výrobci uvádí v případě obou prohlížečů jeho podporu. Na všech ostatních prohlížečích je aplikace stoprocentně funkční a přehledně zobrazená. Nicméně testování u nativních prohlížečů je velice komplikované, protože jejich funkce se nemusí lišit jen v závislosti na jejich verzi, ale také v závislosti na typu zařízení a verzi operačního systému. V případě nekorektní funkčnosti aplikace bych proto jednoznačně doporučil nainstalovat na daný přístroj zmiňovanou Operu Mini 4 nebo 5, která se v testu osvědčila na 100%. Dále je zobrazen graf porovnávající rychlosti komunikace jednotlivých prohlížečů.



Graf 1. Porovnání rychlostí mobilních prohlížečů

3.6. Budoucí vývoj

Aplikace má stále některé nedostatky, které bych v co nejbližším časovém horizontu rád odstranil. Jedná se především o problém nefunkčního JavaScriptu a špatné interpretace nastavení kaskadových stylů při zobrazení stránky u některých prohlížečů. Aplikace by si také jistě zasloužila řadu praktických rozšíření, například možnost jednotlivé záznamy pomocí systému Mstormm vkládat nebo editovat přímo pomocí telefonu. Toto je ovšem otázka dohody se zákazníkem. V nejbližší době je v plánu aplikaci zpřístupnit i realitním makléřům na Slovensku a oživit tak možnost výběru „Slovenská republika“ v přihlašovací formuláři, která v současné době není funkční. Dalším plánovaným rozšířením je funkce zobrazení takzvané „Cenové mapy“, která by mohla vypadat následovně:

Lokalita:	Beroun
Nejlevnější byt	1 280 000,-Kč
Nejdražší byt	3 500 000,-Kč
Průměrná cena za byt	2 024 33,-Kč
Průměrná cena Kč/m ²	29 783,-Kč

Statistika by byla převzata z měsíčních uzávěrek a možnost jejího zobrazení by nejspíše byla doplněna rovnou k filtrům v záložce „Seznam nabídek“.

Závěr

Během práce na tomto projektu jsem se seznámil s moderními metodami tvorby aplikací pro mobilní zařízení a zjistil možnosti využití těchto zařízení jako nástroje pro přístup k informačním systémům. Výsledkem této práce je informační systém Mstormm, který bude dostupný realitním makléřům na jejich mobilních telefonech. Systém se podařilo úspěšně dokončit, otestovat a v nejbližší době bude zpřístupněn realitním makléřům. V současné době je systém přístupný na url adrese: <https://stormm-me.digitalcz.cz/>. Na základě provedených analýz byl nakonec celý systém koncipován jako webová aplikace tvořená v prostředí Zend Framework.

Seznam použité literatury a zdrojů informací

- [1] *Wapedia* [online]. 2010-03-28 [cit. 2010-05-20]. Multiplatformní software. Dostupné z WWW: <<http://wapedia.mobi/cs/Cross-platform>>.
- [2] *Tomas.NET* [online]. 2006-04-24 [cit. 2010-05-20]. Souboj tlusté a tenké aplikace. Dostupné z WWW: <<http://tomas-net.blogspot.com/2006/04/souboj-tlust-tenk-aplikace.html>>.
- [3] *Sefira* [online]. 2009 [cit. 2010-05-20]. Bezpečnost. Dostupné z WWW: <<http://www.sefira.cz/cz/reseni-a-sluzby/it-infrastruktury/bezpecnost>>.
- [4] *Wikipedie* [online]. 2010-07-24 [cit. 2010-05-20]. Překladač. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/Překladač>>.
- [5] *Wikipedie* [online]. 2010-03-31 [cit. 2010-05-20]. Interpret (software). Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Interpret_\(software\)](http://cs.wikipedia.org/wiki/Interpret_(software))>.
- [6] *Wikipedie* [online]. 2010-05-12 [cit. 2010-05-20]. Java ME. Dostupné z WWW: <http://cs.wikipedia.org/wiki/Java_ME>.
- [7] *Lupa* [online]. 2007-12-11 [cit. 2010-05-20]. Webové stránky na mobilních telefonech. Dostupné z WWW: <Webové stránky na mobilních telefonech>.
- [8] *Mobil.cz* [online]. 2010-03-24 [cit. 2010-05-20]. Rok 2009 ovládla v České republice Nokia a levné dotykové mobily. Dostupné z WWW: <http://mobil.idnes.cz/rok-2009-ovladla-v-ceske-republice-nokia-a-levne-dotykove-mobily-puq-/telefony.asp?c=A100323_161908_telefony_kor>.
- [9] *Mobil.cz* [online]. 2008-04-07 [cit. 2010-05-20]. Na web kdykoli a kdekoli - kompletní přehled internetových prohlížečů do mobilu. Dostupné z WWW: <http://palmare.idnes.cz/na-web-kdykoli-a-kdekoli-kompletni-prehled-internetovych-prohlizecu-do-mobilu-1yj-/programy.asp?c=A080406_211108_programy_vok>.
- [10] *MobilMania* [online]. 2008-10-27 [cit. 2010-05-20]. Srovnání mobilních webových prohlížečů. Dostupné z WWW: <<http://www.mobilmania.cz/Titulni-strana/Srovnani-mobilnich-webovych-prohlizecu/sc-21-a-1120606/default.aspx>>.
- [11] *Mobil.cz* [online]. 2008-12-02 [cit. 2010-05-20]. Mobilní Internet Explorer 6. Dostupné z WWW: <http://palmare.idnes.cz/mobilni-internet-explorer-6-plna-podpora-youtube-a-zobrazovani-webu-jako-na-pc-1hz-/programy.asp?c=A081117_150134_programy_ram>.

- [12] *Internet pro všechny* [online]. 2010-03-01 [cit. 2010-05-20]. Mobilní Internet v České republice - kompletní přehled. Dostupné z WWW: <<http://www.internetprovsechny.cz/clanek.php?cid=223>>.
- [13] *Seminarky.cz* [online]. 2007-01-01 [cit. 2010-05-20]. Druhy připojení k internetu. Dostupné z WWW: <<http://www.seminarky.cz/Druhy-pripojeni-k-internetu-11895>>.
- [14] CRM. In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, [cit. 2010-05-20]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/CRM>>.
- [15] *Hospodářské noviny* [online]. 2002-03-19 [cit. 2010-05-20]. Je obtížné zvládnout řízení vztahů se zákazníky. Dostupné z WWW: <<http://hn.ihned.cz/c1-10850290-je-obtizne-zvladnout-rizeni-vztahu-se-zakazniky>>.
- [16] *Businessworld* [online]. 2007-03-07 [cit. 2010-05-20]. CRM systémy nejsou jen pro velké firmy. Dostupné z WWW: <<http://businessworld.cz/produkty-a-sluzby/crm-systemy-nejsou-jen-pro-velke-firmy-3239>>.
- [17] *CRM portál* [online]. [cit. 2010-05-20]. 10 faktorů pro výběr CRM IS. Dostupné z WWW: <<http://www.crmportal.cz/redakcni/uvod-10-faktor-pro-vybr-crm-is>>.
- [18] *Wikipedie* [online]. 2010-10-15 [cit. 2010-05-20]. PHP. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/PHP>>.
- [19] *Vlastův blog* [online]. 2007-03-11 [cit. 2010-05-20]. Zend Framework - Hello World projekt. Dostupné z WWW: <<http://vavru.cz/php/zend-framework-hello-world-projekt-a-mvc-pattern>>.
- [20] *Wikipedie* [online]. 2006-11-29 [cit. 2010-05-20]. MySQL. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/MySQL>>.
- [21] *Wikipedie* [online]. 2010-04-14 [cit. 2010-05-20]. JavaScript. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/JavaScript>>.
- [22] *Interval* [online]. 2007-05-27 [cit. 2010-05-20]. JavaScript s jQuery. Dostupné z WWW: <<http://interval.cz/clanky/javascript-s-jquery-lehky-uvod/>>.
- [23] *Jak psát web* [online]. 111 [cit. 2010-05-20]. Různé prohlížeče. Dostupné z WWW: <<http://www.jakpsatweb.cz/prohlizece.html>>.
- [24] *Safari Reference Library* [online]. 2010-03-24 [cit. 2010-05-20]. Configuring the Viewport. Dostupné z WWW: <<http://developer.apple.com/safari/library/documentation/AppleApplications/Reference/SafariWebContent/UsingtheViewport/UsingtheViewport.html>>.

- [25] *Programujte* [online]. 2008-07-24 [cit. 2010-05-20]. Ajax – 2. lekce. Dostupné z WWW: <<http://programujte.com/?akce=clanek&cl=2008062200-ajax---2-lekce>>.
- [26] *Wikipedie* [online]. 2010-0513 [cit. 2010-05-20]. Session. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/Session>>.
- [27] *Zend Framework : Programmer's Reference Guide* [online]. 2006-01-01 [cit. 2010-05-20]. Global Session Management. Dostupné z WWW: <http://framework.zend.com/manual/en/zend.session.global_session_management.html>.
- [28] *Zdroják* [online]. 2008-10-11 [cit. 2010-05-20]. Bezpečnost na webu - přehled útoků na webové aplikace. Dostupné z WWW: <<http://zdrojak.root.cz/clanky/prehled-utoku-na-webove-aplikace/>>.
- [29] *Amikelive* [online]. 2007-05-27 [cit. 2010-05-20]. Modifying Zend_Db_Adapter_Abstract::quoteInto to accept multiple question marks. Dostupné z WWW: <http://tech.amikelive.com/node-5/modifying-zend_db_adapter_abstractquoteinto-to-accept-multiple-question-marks/>.
- [30] *Religis* [online]. 2009-04-23 [cit. 2010-05-20]. Bezpečnost webových aplikací. Dostupné z WWW: <<http://www.religis.cz/bezpecnost-webovych-aplikaci/>>.
- [31] *Zend Framework: Documentation : Programmer's Reference Guide* [online]. 2006-01-01 [cit. 2010-05-20]. Standard Filter Classes. Dostupné z WWW: <<http://framework.zend.com/manual/en/zend.filter.set.html>>.
- [32] *Interval* [online]. 2002-06-06 [cit. 2010-05-20]. Bezpečnost především - použití SSL. Dostupné z WWW: <<http://interval.cz/clanky/bezpecnost-predevsim-pouziti-ssl/>>.
- [33] *JQuery* [online]. 2010-01-01 [cit. 2010-05-20]. Documentation - Main Page. Dostupné z WWW: <http://docs.jquery.com/Main_Page>.
- [34] *EKINI* [online]. 2007-08-28 [cit. 2010-05-20]. JQuery AJAX in Zend Framework . Dostupné z WWW: <<http://blog.ekini.net/2007/08/28/jquery-ajax-in-zend-framework/>>.
- [20] *Zend Framework* [online]. 2006-01-01 [cit. 2010-05-20]. Zend Framework Documentation. Dostupné z WWW: <<http://www.framework.zend.com/docs/overview>>.
- [23] *Huihoo* [online]. 2006-07-11 [cit. 2010-05-20]. Zend_Db_Table. Dostupné z WWW: <<http://docs.huihoo.com/php/zend/ZendFramework-0.1.5/documentation/end-user/es/zend.db.table.html>>.

Seznam příloh

Struktura s přílohami na CD

1. Projekt Mstormm
2. Webová aplikace jako samostatné rozhraní informačního systému pro mobilní zařízení.doc
3. Webová aplikace jako samostatné rozhraní informačního systému pro mobilní zařízení.pdf