

Absolvování individuální odborné praxe

Individual Professional Practice in the Company

Zadání bakalářské práce

Student: **Petr Zubek**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: **Absolvování individuální odborné praxe**
Individual Professional Practice in the Company

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: AutoCont CZ, a.s.
2. Struktura závěrečné zprávy:
 - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
 - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
 - c) Zvolený postup řešení zadaných úkolů.
 - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
 - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
 - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vede odbornou praxi studenta.


Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jan Gaura**


Konzultant bakalářské práce: Ing. Dalibor Žáček

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014


doc. Dr. Ing. Eduard Sojka
vedoucí katedry




prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2014


.....

Chtěl bych poděkovat vedení firmy za umožnění vykonávání bakalářské práce formou odborné praxe a zejména svému konzultantovi, panu Ing. Daliboru Žáčkovi. Dále bych chtěl poděkovat Janu Maršovi a Ing. Janu Gaurovi.

Abstrakt

Práce popisuje průběh vykonávání bakalářského semináře formou individuální odborné praxe ve firmě AutoCont CZ a.s. Na začátku je čtenář seznámen s firmou, ve které byla bakalářská odborná praxe vykonávána. Dále je popsáno pracovní nasazení studenta, popis úkolů, které byly studentovi přiděleny a jejich časová náročnost. Poté je popsán postup řešení daných úkolů. Závěrem práce je shrnuto, kterých okruhů znalostí nabytých během studia bylo při absolvování praxe využito, a které okruhy bylo nutné dodatečně nastudovat a také celkové zhodnocení odborné praxe.

Klíčová slova: AutoCont, odborná praxe, informační systém, privátní cloud

Abstract

This bachelor thesis describes the course of the individual professional practise in the AutoCont CZ a.s. company. At the beginning the reader is familiar with the company, in which was performed bachelor's professional experience. The following describes the dedication of the student, a description of the tasks that were assigned to the student and their time complexity. Then the described procedure for solving the tasks. Finally the work is summarized, which areas of knowledge acquired during their studies at graduation practice was used and which areas it was necessary to additionally study and the overall evaluation of professional practice.

Keywords: AutoCont, professional practice, information system, private cloud

Seznam použitých zkratk a symbolů

API	– Application Programming Interface
CLR	– Common Language Runtime
CRM	– Customer Relationship Management
DMZ	– Demilitarized Zone
DNS	– Domain Name System
EXT4	– Fourth Extended Filesystem
FAT32	– File Allocation Table 32
GUI	– Graphical User Interface
HTTP	– Hypertext Transfer Protocol
HTTPS	– Hypertext Transfer Protocol Secure
IaaS	– Infrastructure As A Service
IIS	– Internet Information Services
IS	– Informační systém
MAC	– Media Access Control
MBR	– Master Boot Record
MPT	– Master Partition Table
MS	– Microsoft
MSDN	– Microsoft Developer Network
NTFS	– New Technology File System
OS	– Operační systém
PaaS	– Platform As A Service
SaaS	– Software As A Service
SAN	– Storage Area Network
SOAP	– Simple Object Access Protocol
SQL	– Structured Query Language
SSIS	– SQL Server Integration Services
UAC	– User Account Control
VLAN	– Virtual Local Area Network
VM	– Virtual Machine
WAP	– Windows Azure Pack
Windows PE	– Windows Preinstallation Environment
WINS	– Windows Internet Naming Service
XML	– Extensible Markup Language

Obsah

1	Úvod	4
2	Popis odborného zaměření firmy a pracovní zařazení	5
2.1	Odborné zaměření	5
2.2	Náplň práce	5
2.3	Pracovní zařazení	5
3	Seznam zadaných úkolů	6
3.1	Aplikace pro přepínání systémů (JumpPart)	6
3.2	Android aplikace pro IS Akro	6
3.3	Privátní cloud	6
3.4	Úpravy v informačním systému Akro	7
3.5	Administrace firemních zákazníků	7
4	Postup řešení zadaných úkolů	8
4.1	JumpPart	8
4.2	Android aplikace pro IS Akro	9
4.3	Privátní cloud	13
4.4	Úpravy v informačním systému Akro	16
4.5	Administrace firemních zákazníků	17
5	Závěr	18
5.1	Uplatnění teoretické a praktické znalosti	18
5.2	Scházející znalosti	18
5.3	Dosažené výsledky v průběhu praxe a jejich zhodnocení	18
6	Reference	19
7	Přílohy	20

Seznam tabulek

1	MBR [3]	8
---	-------------------	---

Seznam obrázků

1	Ukázka webové služby	10
2	Náhled na mřížku kalendáře	12
3	Topologie cloudu	14

1 Úvod

Cílem této bakalářské práce je popis mého působení ve firmě AutoCont CZ a.s. – školicí středisko. Ve firmě jsem dříve pracoval na postu správce sítě, operačních systémů a serverů. Proto jsem si vybral vypracování bakalářské práce formou praxe. V práci se zaměřím pouze na 5. a 6. semestr. Během praxe jsem se snažil zaměřit na programování aplikací. V důsledku toho bylo potřeba konzultovat o problematice nejen s konzultantem odborné praxe, ale i s oddělením Vývoje.

Na začátku práce popíšu, čím se firma AutoCont CZ a.s. (dále jen AutoCont) zabývá, poté popíšu, čím se zabývá její oddělení Školicí středisko. V další části práce Vás obeznámím se zadanými úkoly, které jsem v průběhu praxe konal. Nastíním problematiku a vysvětlím, jak jsem dané problémy řešil.

2 Popis odborného zaměření firmy a pracovní zařazení

2.1 Odborné zaměření

AutoCont CZ je českou soukromou společností, která v oblasti dodávek zboží, služeb a řešení na poli informačních a komunikačních technologií úspěšně působí od roku 1990.

K rychlému rozvoji v prvních deseti letech přispěla výroba vlastní značky osobních počítačů a systematické budování rozsáhlé obchodní a servisní sítě. V druhém desetiletí své existence se AutoCont CZ zaměřil na poskytování komplexních IT řešení a služeb pro firemní klientelu a státní správu, čímž se postupně propracoval mezi nejvýznamnější společnosti v oblasti rozsáhlých a složitých ICT projektů.

Systematické a dlouhodobé budování odborných schopností, pečlivé sledování potřeb zákazníků a v neposlední řadě i promyšlená vnitřní organizace firmy – to vše přispělo k tomu, že je AutoCont CZ v současné době největším a nejvýznamnějším českým dodavatelem informačních a komunikačních technologií.[1]

Oddělení Školicí středisko, jak z názvu vyplývá, se zabývá školením a mimo jiné konzultacemi v oboru IT.

2.2 Náplň práce

Hlavní náplní mé odborné praxe bylo programování, ať už v jazyku C++, C# nebo na platformě Android, tak i různé administrační úkony, do kterých je zahrnuta i tvorba privátního cloudu. Bylo mi umožněno vyzkoušet si, jak fungují procesy ve firmě, kdy bylo potřeba nasadit aplikaci do ostrého provozu, provést bezpečnostní testy a samotné nasazení.

2.3 Pracovní zařazení

Byl jsem zařazen do divize IT infrastruktura (ITI), kde jsem byl oficiálně zařazen do oddělení Školicího střediska. Měl jsem tu možnost kromě oddělení Školicího střediska vyzkoušet i oddělení servisu, kde jsem byl Školicím střediskem na krátkou dobu propůjčen.

3 Seznam zadaných úkolů

V průběhu odborné praxe jsem se setkal s celou řadou úkolů z různých sfér firemního podnikání, ať už v oboru programování nebo správy systémů. Úkoly v celé práci jsou seřazeny v takovém pořadí, v jakém mi byly zadány.

3.1 Aplikace pro přepínání systémů (JumpPart)

První úkol, který mi byl zadán, byla aplikace, jejímž úkolem bylo přepínat operační systémy.

3.1.1 Zadání

Vytvoříte aplikaci, která bude přepínat operační systém. Aplikace nebude používat boot menu. Aplikace musí fungovat v prostředí Windows PE.

3.1.2 Vyjádření časové náročnosti

Úkol mi byl zadán začátkem října 2013. Vyskytly se drobné komplikace s tvorbou vhodné masky, a proto vývoj zabral přibližně **100 hodin**.

3.2 Android aplikace pro IS Akro

Dalším úkolem bylo naprogramovat mobilní aplikaci, která reprezentovala informace z interního informačního systému.

3.2.1 Zadání

Vytvořte aplikaci na platformě Android, která bude portovat klíčové vlastnosti informačního systému Akro z prostředí Rozvrh. Aplikace bude funkční na různorodých zařízeních, konkrétně na zařízeních zaměstnanců.

3.2.2 Vyjádření časové náročnosti

Tento úkol neobsahoval napsat jenom aplikaci pro Android, ale také byla potřeba vytvořit webovou službu, která obsluhuje tuto aplikaci, dále bylo potřeba danou aplikaci nasadit do prostředí, kde jsou jisté bezpečnostní restriktce, a proto si úkol vyžádal **150 hodin**.

3.3 Privátní cloud

Následujícím úkolem bylo vytvořit privátní cloud na platformě Windows Azure.

3.3.1 Zadání

Vytvořte privátní cloudovou službu, která bude snadno obsluhovatelná. Bude sloužit pro laboratoře zaměstnanců a pro výukové virtuální stroje. K dispozici dostanete 3 servery.

3.3.2 Vyjádření časové náročnosti

Pro tento úkol byla potřeba **60 hodin**.

3.4 Úpravy v informačním systému Akro

Předposledním úkolem bylo upravit informační systém dle požadavků. Ve většině se jednalo o dokončení nebo upravení částí informačního systému.

3.4.1 Zadání

Upravíte náš informační systém dle požadavků, které budeme průběžně konzultovat.

3.4.2 Vyjádření časové náročnosti

I přesto, že byla potřeba zorientovat se v cizím kódu, jednalo se o snadné úpravy, proto byla potřeba **80 hodin**.

3.5 Administrace firemních zákazníků

Následujícím a posledním úkolem bylo držet podporu pro firemní zákazníky nad serverovými produkty firmy Microsoft.

3.5.1 Zadání

Budete držet podporu pro firemní zákazníky, co se týče serverových produktů.

3.5.2 Vyjádření časové náročnosti

V oddělení servisu jsem strávil přibližně **30 hodin**.

4 Postup řešení zadaných úkolů

Při řešení všech úkolů, které mi byly zadány, jsem využil teoretických znalostí, které jsem získal během studia jak na střední, vysoké škole, tak během dřívější praxe ve firmě. V průběhu praxe bylo klíčové ovládat anglický jazyk, nýbrž mnoho odborné literatury je psáno právě v tomto jazyce.

4.1 JumpPart

JumpPart je software, který slouží k přepínání partition. Tento software byl vyvíjen za účelem využití ve Školícím středisku. Bylo potřeba přepínat operační systémy bez použití bootladeru, kdy si uživatel vybírá z více operačních systémů a bylo nutné tyto partition schovávat. Jeden z dalších požadavků byla dostupnost softwaru z libovolného operačního systému Windows, s podmínkou dostupnosti z prostředí Windows PE (Preinstallation Environment), kde není dostupný žádný framework. Toto prostředí je používáno pro instalaci operačního systému Windows.

Princip tohoto programu je v podstatě jednoduchý. Teoreticky stačí, abych si binárně načtl prvních 512 bajtů z disku, kde se nachází MBR. V programu prakticky načítám 1536 bajtů, pro potřebu ukládat záložní data MBR, adresu serveru a další údaje.

Z počátku se jevilo obtížným vytvořit masku, tedy strukturu, která by přesně kopírovala MBR. I když jsem použil datové typy, které měly přesně sedět na velikost jednotlivých atributů v MBR, přesto jsem dostával špatné výsledky. Po poradě s konzultantem a prozkoumání dokumentace datových typů na MSDN[2], jsem úspěšných výsledků dosáhl s datovým typem WORD, DWORD a BYTE. To byla první a poslední komplikace při programování této aplikace.

Při načtení masky mě nejvíce zajímala část MBR, kde se vyskytovala tabulka rozdělení disků (MPT nebo-li partition record) a signatura MBR. Signaturu MBR (je to hodnota, která se vyskytuje na konci MBR a obsahuje hodnotu „AA55“) jsem využíval na ověření toho, že jsem záznam MBR správně načtl a že není nijak porušen. Pokud tato podmínka byla porušena, program se regulérně ukončil.

Struktura MBR				
Adresa			Popis	Délka v bajtech
Hex	Oct	Dec		
0000	0000	0	Kód zavaděče	440 (max 446)
01B8	0670	440	Volitelná signatura disku	4
01BC	0674	444	Obvykle nuly; 0x0000	2
01BE	0676	446	Tabulka rozdělení disku (MPT) (4 položky po 16 bajtech, IBM schéma oddílů)	64
01FE	0776	510	Signatura MBR; 0xAA55	2
01FF	0777	511		
Celková délka MBR: 446 + 64 + 2 =				512

Tabulka 1: MBR [3]

V tabulce rozdělení disků byla důležitá hodnota status, tedy zda daný oddíl je „bootovací“ (aktivní) či nikoliv. Hodnota pro aktivní oddíl byla „0x80“ a neaktivní „0x00“. V případě jiné hodnoty je oddíl vyhodnocen jako poškozený. Další důležitým záznamem byl typ daného oddílu, tedy jestli je oddíl naformátován souborovým systémem FAT32, NTFS či ext4 apod.

JumpPart obsahuje tzv. síťový mód, kdy je aplikace spuštěna s parametrem „/n“. Naskytovalo se mnoho řešení, jak tento problém realizovat. Řešení pomocí databáze bylo zbytečně velké a robustní. Také bylo zbytečně řešit problém pomocí serverové aplikace jako protikus. Nezbylo nic jednoduššího, než využít souborových služeb a nasdílet konfigurační soubor. Aplikace měla nastavenou výchozí cestu k souboru, popřípadě bylo možno jí patřičným přepínačem změnit.

Při spuštění síťového režimu si aplikace získala MAC adresu aktivní síťové karty, poté otevřela sdílený soubor. Pokud nenašla svojí MAC adresu v seznamu, tak ji vytvořila a přidala ještě záznam aktivní partition. Takto to šlo velmi efektivně a automatizovaně přepínat hromadně počítače pomocí konfiguračního souboru.

Pro používání na operačních systémech Windows Vista a novějších bylo potřeba, aby si aplikace vyžádala administrátorská práva. Toho se dosáhlo patřičným nastavením v Linkeru, dále v Manifest souboru, kde bylo nastaveno, že UAC spouštěcí úroveň je pouze pro administrátory.

Dalším požadavkem bylo správné použití parametrů při spuštění aplikace. Samozřejmostí bylo zobrazení nápovědy k použití programu. Aplikaci lze spustit s číselným parametrem, který reprezentuje, který oddíl má být aktivní (v rozsahu 0-3). Dalším z parametrů je parametr „/o“, který obnoví tabulku rozdělení disků do stavu, který byl před prvním spuštěním této aplikace. Předposledním parametrem je parametr „/uX“ (kde X je číslo v rozsahu 0-3). Úkolem tohoto parametru je zpřístupnit skrytou partition, tedy její zviditelnění. Poslední parametr je „/n“, který je popsán výše.

4.2 Android aplikace pro IS Akro

Jedná se o informační systém, který je používán ve školicím středisku. Vyvinula jej česká firma Nativa Enterprise s.r.o. IS byl vytvořen v roce 2004. Název je složen ze dvou slov a to Akce a Rozvrhy. IS slouží k vypisování kurzů, plánování termínů (rozvrhů), vedení lektorů, lokalit a podobně. Databáze je provozována na Microsoft SQL Serveru 2008 R2. IS je logicky rozdělen do dvou světů, jeden ze světů jsou výše zmíněné Akce a druhý jsou Rozvrhy. Aplikace pro Akce je naprogramována v prostředí Microsoft Access 2010. Aplikace pro Rozvrhy je napsána v jazyce C#.

Mým úkolem bylo vytvořit aplikaci Rozvrh pro mobilní zařízení na platformě OS Android. Rozvrh je zjednodušeně řečeno plánovací systém, který zobrazuje jednotlivá školení v přehledné mřížce. Tato mřížka je organizována podle dnů a lokalit. Při tvorbě aplikace bylo vyžadováno, aby veškerá komunikace byla uskutečněna skrze protokol SOAP. Aplikace má sloužit čistě jako čtečka a není možno měnit nebo vytvářet data, ať už z bezpečnostních, tak také z důvodů ryze technických.

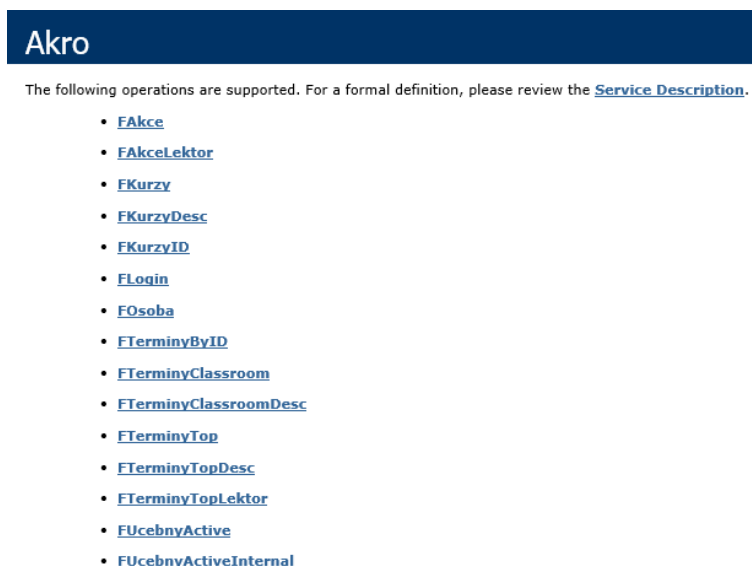
Tvorba aplikace se rozpadla na dva podúkoly. Na webovou službu a samotnou Android Aplikaci.

4.2.1 Akro Webservice

Ještě než se rozepíšu o webové službě, je třeba se zmínit o síťové topologii. Databázový server je umístěn v intranetu, z čehož vyplývá, že není na internetu dostupný. Firemní bezpečnostní politiky jsou takové, že cokoliv se vyskytuje v intranetu, nesmí být publikováno přímo. Tudíž do hry se dostává tzv. Demilitarizovaná zóna (DMZ), kde je umístěn další databázový server a webový server. I přesto, že databázový server je v DMZ, není lehce přístupný. Proto je ideální publikovat data skrze webový server IIS. Databázový server z intranetu zasílá data na databázový server do DMZ v 10 minutovém intervalu. Pro upřesnění, server z intranetu má přístup do DMZ, ale naopak ne (zde narážím na důvody ryze technické, proč nelze měnit a vytvářet data).

Bylo potřebné upravit SSIS balíček. SSIS je komponenta MS SQL Serveru, která umožňuje mnoho možností, jak migrovat data. [4] [5] Bylo třeba tedy dodefinovat, které tabulky a sloupce se mají migrovat. Poté následovalo nahrazení stávajícího balíčku na SQL serveru.

Úkol webové služby je jednoduchý, vytvořit mezivrstvu mezi databází a Android zařízením, která publikuje data do internetu. K databázi se přistupuje přes klasický SQL Connection, data jsou přístupná skrze protokol SOAP.



Obrázek 1: Ukázka webové služby

SOAP je protokol, jenž je určen pro výměnu dat, která jsou strukturována do XML. Toto vše je realizováno skrze webové služby. XML dokument má kořenový element Envelope (obálka), ten obsahuje dva elementy, Header (hlavička) a Body (tělo), přičemž hlavička není povinná. Často se používá pro přenos pomocných informací (id uživatele nebo autentizace). V těle jsou obsaženy informace, ty obsahují identifikaci volané služby a předávané parametry. Je třeba podotknout, že SOAP používá jmenných prostorů, jež identifikují jednotlivé části XML. [7] [8]

V mém případě je SOAP obálka doplněna o hlavičku, která obsahuje uživatelské jméno a heslo. Tyto položky jsou šifrovány algoritmem Rijndael, kde je použit 128 bitový klíč. Vyšší nešel použít z důvodu limitace na zařízení Android. Celý SOAP protokol běží pod HTTPS, takže ve výsledku jsou všechny položky šifrovány. Vše je nastaveno tak, že položky

v hlavičce jsou povinné. Pokud nejsou vyplněny nebo jsou vyplněny špatně, webová služba vrací prázdný objekt. Protokol SOAP má výhodu, že není potřeba ručně serializovat objekt, o tuto činnost se stará CLR (common language runtime).

Nepříjemným překvapením bylo, že názvy tabulek a sloupců v databázi byly v češtině. Tudíž se ve zdrojovém kódu jak webové aplikace, tak klientské aplikace může vyskytovat český název, který reprezentuje nějakou část databáze. Bylo vhodné dodržet názvy dle databáze.

4.2.2 Akro Android Aplikace

Akro aplikace pro Android je protikus webové služby Akro. Operační systém Android nemá přímou podporu MS SQL, to byla další z výhod, proč použít protokol SOAP. Pro usnadnění práce byla použita kSOAP2 knihovna [9]. Po průzkumu ve firmě a návrhu komponent byla zvolena jako nejnižší verze API 15, což odpovídá verzi Androidu 4.0.3. Aplikace vyžaduje oprávnění k internetu a přístup ke stavu připojení, ten je vyžadován z důvodu ověřování, zda má zařízení přístup k internetu.

Při spuštění aplikace je vyžadováno přihlášení k serveru. Aplikace a přístup k serveru je poskytnut vedením Školícího střediska. Po úspěšném přihlášení je zobrazena hlavní obrazovka, která obsahuje informace o přihlášeném uživateli. Pokud uživatel je lektor, jsou mu na hlavní obrazovce zobrazeny dodatečné informace o nadcházejících školeních.

Při navrhování aplikace byl kladen důraz na to, aby byly dodrženy návrhové vzory, co se týče uživatelského rozhraní. Z tohoto důvodu byl využit prvek Navigation Drawer někdy také Left Drawer, který velmi usnadnil pohyb uživatele po aplikaci. Je to prvek v uživatelském rozhraní, jednoduše řečeno je to menu, které se vysune z levé strany displeje, při použití gesta nebo stisknutí příslušné ikony. Menu obsahuje všechny důležité úkony, jako návrat na hlavní obrazovku, kalendář, prohlížeč kurzů, prohlížeč termínů a další.

Vytvoření kalendáře byl nejtěžší úkol na této aplikaci. Z důvodu zachování kompatibility s počítačovou verzí bylo potřeba vytvořit kalendář tak, aby jeden řádek reprezentoval jeden den a celý obsah displeje obsahoval jeden měsíc s tím, že se dá samozřejmě srolovat nahoru a dolů. Už při tomto zadání je jasné, že se nedá využít klasických kalendářů v Androidu. Nabízí se řešení použití dvourozměrné mřížky tzv. GridView. V Akro kalendáři je třeba zvýraznit danou akci grafickým objektem a v případě, že akce trvá více než jeden den, je třeba objekt rozšířit na patřičný počet dnů. To v praxi znamená, jeden objekt musí obsadit více polí. Toto bohužel není ve standardním GridView možné. Proto bylo nevyhnutelné využít alternativního prvku, který není standardně obsažen v systému Android. Nabízelo se řešení napsat si prvek sám, ale bylo potřeba daný problém velmi rychle vyřešit. Psaní vlastní komponenty by bylo velmi časově náročné, a proto jsem zvolil prvek Staggered GridView[10], který je nabízen Android komunitou pod licencí Apache License v2.0. Tento prvek vyniká právě tím, že lze objekty v mřížce snadno roztáhnout. Ale ani toto řešení se neobešlo bez problémů, neboť komponenta obsahovala chyby (bugy). První z nich byl problém s rolováním obrazovky, kdy v náhodných případech přestalo rolování fungovat. Toho jsem dosáhl zakázáním recyklace mezipaměti (cache), jejímž úkolem bylo mazat části obrazovky, které nejsou vidět. Další chyba se projevovala, pokud zařízení bylo v landscape (na šířku, naležato) režimu. Komponenta způsobila pád celé aplikace. Chyba se vyskytovala v algoritmu, jenž měl naplnit mřížku objekty. Algoritmus v landscape režimu špatně počítal odsazení objektů v mřížce. Stačilo pouze přidat podmínku, která ověřila, zda počet sloupců byl shodný s počtem sloupců, když bylo zařízení v režimu na výšku. Pokud toto nebylo splněno, použilo se odsazení pro první řádek mřížky.

Kalendář dále umožňuje přepínat se mezi lokalitami pomocí gesta doprava nebo doleva. A samozřejmě je, že v kalendáři lze měnit datum po měsíci nebo roce. Což je prvek, který slouží k výběru data. Tento prvek je realizován pomocí DatePickerDialog (třída, řešící logiku výběru data). Tento prvek je upraven tak, že skrývá položku pro výběr dne. Uvedený prvek dědí z DialogFragment, jež je podle směrnic Androidu silně doporučeno. DialogFragment je podporován až od verze API 11, to odpovídá verzi Androidu 3.0. V případě nižší verze by bylo potřeba využít zastaralých prvků nebo si napsat prvek vlastní, v mém případě to však neplatí, neboť nejnižší podporovaná verze aplikace je 4.0.3.

	ADMI	LANDE
13. (neděle)		
14. (pondělí)	PRONSS	20412C 6421B
15. (úterý)		
16. (středa)	PRONSS	
17. (čtvrtek)	EX2	
18. (pátek)		
19. (sobota)		
20. (neděle)		
21. (pondělí)		
22. (úterý)	6292A	2778 SQLJ

Obrázek 2: Náhled na mřížku kalendáře

Zajímavý je algoritmus, který plní objekty do mřížky a stojí za to jej zmínit. Bylo důležité seřadit správně položky v databázi na úrovni SQL. Řadilo se dle data začátku kurzu a dle pozice v kalendáři (pozicí je myšlen sloupec, kde se vykreslí objekt). První fází algoritmu bylo projít výsledek SQL příkazu a spočítat si počet dnů úseku daného termínu, uložit si identifikátor termínu, ukazatel na pořadí výsledku SQL a datum začátku kurzu. Další část algoritmu lineárně procházela sloupce a řádky. Pro každý sloupec bylo nutné si uchovávat tzv. proměnnou distanc. Do té byl ukládán počet dnů daného termínu. Pokud byl distanc různý od nuly pro daný sloupec, sloupec byl automaticky přeskočen. Dále algoritmus musel rozlišovat jednotlivé barvy, ty jsou stejné jako v počítačové verzi, kdy žlutá barva označuje veřejný kurz, červená barva značí stornovaný kurz, modrá barva neveřejný kurz, fialová barva je pro testy v testovacím středisku, zelená barva značí pronájem. Je zde omezení pro prohlížení kalendáře v režimu na výšku, kdy displej zařízení není dostatečně široký, a proto není vidět 4. sloupec (v Akru může být libovolný počet sloupců, ale primárně se využívají 4 sloupce). Při otočení zařízení do landscape režimu jsou vidět všechny sloupce. Pokud se krátce dotkneme objektu v mřížce, bude zobrazena informační bublina, která nás informuje o plném názvu kurzu. Při dlouhém stisknutí bude zobrazena kontextová nabídka, z té si můžeme vybrat z několika možností. Nejdůležitější možností je položka informace, která nám zobrazí úplné informace o termínu, tedy o jaký kurz se jedná, kdo je lektor, jaká je cena apod.

Další ze součástí aplikace je prohlížeč kurzů, ten umožňuje rolovat mezi kurzy. Bylo potřeba si napsat vlastní rozhraní, které umožňuje filtrování dle jednotlivých položek a ne-standardní zobrazení jednotlivých položek kurzů. Samozřejmostí je, že kurzy lze řadit dle svého data zveřejnění. Stejně analogicky je napsán prohlížeč termínů.

Poslední položka v menu je nastavení, které obsahuje změnu adresy serveru, volbu lokalit, které budou v kalendáři zobrazeny. Zobrazení všech učeben, při volbě této možnosti jsou zobrazeny i externí lokality, které nejsou často využívány. A dále se vyskytuje nastavení, zda se termíny, či kurzy mají řadit zpětně či počet zobrazených kurzů v prohlížeči.

Velmi velký důraz byl kladen na optimalizaci aplikace, co se týče rychlosti. V náročných úkolech bylo použití vláken nevyhnutelné, cílem bylo využít potenciál vícejádrových zařízení. Snahou bylo vytvořit velmi rychlou a stabilní aplikaci. Během vývoje bylo třeba aplikaci testovat na více zařízeních.

4.3 Privátní cloud

Cílem tohoto projektu bylo vytvořit cloudovou službu pro interní účely firmy, nejčastěji pro počítačové laboratoře zaměstnanců, či virtuálních školících strojů pro Školící středisko. Platforma pro privátní cloud byla zvolena Windows Azure.

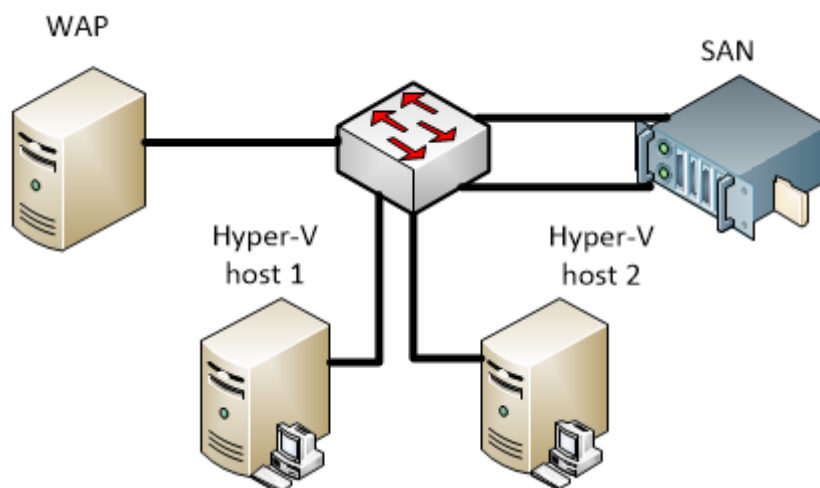
Produkt, který Microsoft nabízí veřejnosti, se nazývá Windows Azure Pack (WAP), ten je totožný se službou Windows Azure, kterou Microsoft provozuje. WAP je sada nástrojů, která zajišťuje poskytnutí samoobslužného portálu (portál pro správu služeb, portál pro administraci samotného WAP) a svázání Microsoft produktů a technologií. Je nutno podotknout, že samotný WAP je zdarma. Jedná se o verzi WAP 2.0, ale běžně se s tímto označením nesetkáte. Dřívější verze se nazývala Windows Server for Hosting Service Provider. Tento dřívější název nebyl přesně vystihující, že by mělo jít o cloudovou službu, proto byl zvolen nyníější název WAP.

Cloudy se dělí dle nabídky služeb, avšak samotný cloud může poskytovat více služeb, než jen jednu.

- **Infrastruktura jako služba (Infrastructure As A Service, IaaS)**, je služba, jenž je velmi podobná tradičním hostingům, kde firma bude využívat hostovaného prostředí jako logického rozšíření prostor datacentra. Servery jsou pronajímány na základě potřeby, jak fyzické servery, tak virtuální. Správce, který spravuje infrastrukturu, má plnou kontrolu nad softwarovou konfigurací. Někteří poskytovatelé mohou také povolit flexibilitu hardwaru, což způsobuje, že tato služba bývá zpravidla nejdražší. Toto řešení nabízí plnou svobodu v tom, co se na daném stroji bude provozovat, avšak údržba infrastruktury je čistě na uživateli. Častým zástupcem jsou servery, virtuální servery, load balancery (zařízení k vyrovnání zátěže síťového provozu), sítě.
- **Platforma jako služba (Platform As A Service, PaaS)**, je služba, která je o stupeň výše než IaaS. Plní potřeby těch, kteří chtějí budovat a provozovat vlastní aplikace jako služby. Poskytovatel služby nabízí přednastavené virtualizované aplikační prostředí, kde uživateli je umožněno napsat svou aplikaci. Zde je minimalizována potřeba správce IT ze strany uživatele, protože správa serveru spadá pod správu poskytovatele. Častým zástupcem jsou databáze, webové servery, vývojové nástroje.
- **Software jako služba (Software As A Service, SaaS)**, služba, která je opět výše než PaaS. Je to nasazení softwaru, kdy aplikace je hostována poskytovatelem. Většinou

nebývají žádné vstupní závazky, tudíž zákazník nepodléhá dlouhodobým smlouvám (není to pravidlem, ale v praxi to tak většinou bývá). Díky tomu, že nejsou dlouhodobé smlouvy pravidlem, zákazník může přerušit službu kdykoliv. Pro SaaS je základní infrastruktura a konfigurace softwaru neviditelná, proto se zákazník musí spokojit s funkcemi, které jsou poskytnuty jako výchozí (nelze dodatečně měnit nějaké konfigurace softwaru). Zástupcem jsou CRM, emaily, hry, apod. Běžné aplikace, které spadají pod SaaS jsou Google Apps, Microsoft Office 365. [11] [12]

Já se při tvorbě privátního cloudu zaměřím na Iaas a PaaS, protože bude potřeba často aplikace a operační systém různě konfigurovat. Z důvodu kompatibility a doporučení ze strany Microsoftu je vhodné používat operační systém Windows Server 2012 a vyšší. Je třeba zajistit 5 serverů, ať už virtuálních nebo fyzických. Na každém z nich budou nainstalovány jednotlivé role. Je potřeba serveru, na kterém bude nainstalována Active Directory služba, Active Directory Federation Services a Active Directory Certificate Services. Na další server je třeba nainstalovat samotný Windows Azure Pack (WAP). Na další server je třeba nainstalovat Service Provider Foundation, který je součástí produktu System Center 2012 – Orchestrator. Na následující server je potřeba SQL serveru, kde bude připravena instance pro připojení WAPu. Na poslední server je třeba nainstalovat System Center Virtual Machine Manager 2012 (SC VMM), který spravuje minimálně jednoho Hyper-V hosta (v mém případě se jedná o hosty dva), to znamená, že je potřeba ještě jednoho serveru, tentokrát musí být fyzický. Na Hyper-V hosta není potřeba instalovat plnohodnotný systém, ale stačí nainstalovat edici Microsoft Hyper-V Server 2012 a vyšší. Jedná se o edici, která podporuje jenom roli Hyper-V a neobsahuje GUI. Výhoda této edice je, že je zdarma.



Obrázek 3: Topologie cloudu

Dle obrázku výše je jasná počáteční topologie. Máme k dispozici 3 servery a jedno SAN úložiště. Jeden server využijeme k hostování klíčových služeb WAP, kde všechny tyto servery jsou virtualizovány. Zbývající servery budou využity jako Hyper-V hosti pro SC VMM. Všechny servery jsou nainstalovány operačním systémem Windows Server 2012 R2. Nainstalujeme tedy všechny klíčové role/služby.

Abychom se dostali k prvotní konfiguraci WAPu, je důležité znát port, na kterém je nasazena webová stránka, která nám toto umožňuje. Port pro tuto stránku je 30101. V prvotní

konfiguraci WAPu je potřeba vložit údaje pro spojení s SQL serverem, tedy jeho serverový název, typ autentifikace (je doporučeno používat SQL server autentifikaci), poté uživatelské jméno a heslo SQL server administrátora. Pak je nutné zadat heslo ke konfiguračnímu úložišti. Díky tomuto heslu se lze dostat k citlivým údajům uživatelů a toto heslo bude sloužit i jako administrátorské heslo ke všem nově nasazeným počítačům. Pro ověření, že instalace proběhla úspěšně, můžeme nahlédnout do administrace IIS, kde uvidíme nově nasazené weby a také do administrace SQL serveru, kde budou vytvořeny nové tabulky.

Pokud jsme udělali vše správně, měli bychom být schopni se přihlásit na web na portu 30091. Jedná se o administraci Azure portálu. Pokud jsme si nechali vygenerovat certifikáty, které jsou podepsány samy sebou, budeme se potýkat na celém portálu s výstrahou, že certifikát není vydán důvěryhodnou certifikační autoritou. Pro přihlášení k portálu je vhodné použít administrátorský doménový účet.

V administraci zvolíme VM Cloudy. Než je začneme využívat, je nutné Azure propojit s potřebnými službami. Zde využijeme System Center Service Provider Foundation, který zaregistrujeme. Vložíme servisní URL, která bývá standardně na portu 8090 (<https://spf-server:8090>). Dále je potřebné vložit servisní účet, pro který jsme SPF nainstalovali a heslo. Pokud jsme nainstalovali Service Management Automation (SMA), je vhodné jej také zaregistrovat. SMA nám umožňuje snadno monitorovat, automatizovat tvorbu a nasazení. Tato služba běží standardně na portu 9090. Zaregistrujeme i službu Service Provider Usage.

Přesuneme se na virtuální stroj, kde je nainstalován SC VMM. Tento stroj řídí veškerou virtualizaci v cloudu. Zde je nutné mít registrovaného Hyper-V hosta. Nadefinujeme logické sítě včetně IP rozsahů a dalších důležitých parametrů (DNS, WINS, VLAN). Dále je nutné definovat disková úložiště, v mém případě SAN úložiště. Vytvoříme v rámci SC VMM cloud, který bude propagován do WAPu, zde je potřeba nakonfigurovat, jaké skupiny zdrojů budeme využívat (Hyper-V hosty), do kterých logických sítí bude spadat námi tvořený cloud, zda budeme využívat load balancerů, jaká úložiště budeme využívat, jaké bude mít kapacity (z hlediska hardwarového výkonu), zda jsou omezení, na jaké virtualizační technologii bude tento cloud založen (VMware ESX, Citrix Xen, Microsoft Hyper-V) a jaká knihovna bude nabídnuta.

Knihovna je důležitá kapitola, protože zde rozhodujeme, co náš cloud bude nabízet za operační systémy či aplikační platformy (Exchange, SharePoint, SQL). Pro knihovnu je nutné dodat virtuální disky. Systémy, které jsou na těchto virtuálních discích, musí projít tzv. procesem Sysprep (System Preparation). Tento proces se postará o to, že operační systém bude mít resetované SID (Security ID). Dále na tomto disku nainstalujeme patřičnou platformu. Poté vygenerujeme balíček rozšiřujících zdrojů (resextpkg).

Vrátíme se k administraci Azure portálu. Ve VM Clouds zvolíme Cloudy a dále zvolíme možnost, že chceme použít existujícího VM Cloudu, aby nám poskytl virtuální stroje. Zadáme tedy doménový název našeho SC VMM počítače a zvolíme registraci. Poté máme Azure propojeno s SC VMM. Zvolíme galerii a naimportujeme balíček rozšiřujících zdrojů. Daný stroj v galerii nastavíme, že se má zveřejnit uživatelům.

Nakonfigurujeme SQL server pro PaaS řešení. Konfigurovat můžeme jak SQL, tak MySQL cloudy. Záleží čistě na nás, co chceme poskytovat. V mém případě to bude MS SQL. Vzhledem k tomu, že v cloudu, který budu provozovat nebude až tak velký nápor na SQL, mohu si dovolit nakonfigurovat SQL, které již máme nainstalováno a slouží k řízení Azure. Nadefinuji přístupové údaje a velikost databáze, kterou bude SQL server nabízet.

Vytvoříme plán. Plánem je myšleno, jaké tarify budou nabízeny. Zde definuji výkon, který budu nabízet, popřípadě do jakého cloudu bude daný tarif patřit a další limity. Dále je potřeba

vytvořit uživatele, který bude mít přístup do cloudu, při tvorbě uživatele definuji, jaký tarif bude využívat. Uživatel má přístupný portál pro Azure uživatele na portu 30081, kde může využívat své služby. Azure může být administrován skrze PowerShell, kde je velká škála příkazů, pomocí kterých lze velmi rychle a efektivně Azure spravovat. Azure tak podporuje knihovnu doplňků, kde lze vybírat z mnoha produktů, které mohou být nasazeny v cloudu jak různé frameworky (PHP, Django), tak hotové produkty (Kentico CMS, Moodle, Drupal apod.).

4.4 Úpravy v informačním systému Akro

Informační systém Akro už byl představen v části, kdy jsem programoval aplikaci pro Android. V tomto případě se jednalo o drobné úpravy a dodělávky v informačním systému. Podotýkám, že se samotné úpravy nijak nedotkly aplikace pro Android, a tudíž nebylo potřeba provádět jakékoliv úpravy i v samotné aplikaci.

Nejčastěji se jednalo o následující úpravy, doplnění polí do existujících tabulek s následným provázáním do uživatelského rozhraní, úpravy a opravy formulářů pro uživatelské rozhraní a návrhy nových formulářů. Při plnění tohoto úkolu bylo velmi důležité orientovat se v cizím kódu.

První z následujících úprav bylo přidat do tabulky Projektů pole pro číslo projektu v Axaptě. Axapta je účetní informační systém, který používá firma AutoCont. Je třeba nějak identifikovat projekt mezi těmito systémy. Dále bylo potřeba přidat sloupec, který identifikuje hlavního obchodníka, který daný projekt získal. Poté bylo potřeba provázat tyto sloupce s aplikací. Podotýkám, že tyto změny ovlivňovaly část informačního systému Akce. Dále bylo potřeba vytvořit formulář pro Projekty, kde je možnost filtrování dle parametrů.

Další úprava se jednala ohledně formuláře firem, kde bylo potřeba vytvořit v kontaktních osobách položku, zda osoba je obchodníkem. Dále bylo potřeba upravit řazení osob, aby se řadily dle příjmení, původní řazení bylo dle identifikace v tabulce databáze. Novou funkci, kterou bylo třeba naimplementovat, byla funkce, která umožňuje přesouvat existující termín mezi lokalitami.

Nejvýznamnější úpravou bylo rozšíření IS Akro o evidenci techniky (technické zdroje). Školící středisko provádí školení u zákazníků, kdy se na místo zákazníka dováží technika, tedy notebooky. Tudíž bylo potřeba evidovat, jaké počítače jsou u kterých zákazníků a které počítače jsou nevyužity. Tato změna vyžadovala úpravy v databázi. Konkrétně bylo nutné vytvořit tabulku Technika, Skupiny Zdrojů a vazební tabulku mezi Skupinou Zdrojů a tabulkou Termíny. Dále bylo potřeba vytvořit formulář pro editaci jednotlivých technických zdrojů, kde je možno vyplnit konfiguraci počítače, jeho označení v rámci organizace a jeho přiřazení do skupiny zdrojů. Do stávajícího formuláře byla vložena i administrace skupin zdrojů. Dále bylo potřeba ve formuláři termín přidat kartu technických zdrojů, kde se technické zdroje přidružují k termínům. V aplikaci Rozvrh byl vytvořen křížový pohled (stejná logika jako u Akro Android aplikace, kdy řádky jsou vyjádřeny jako dny, ale sloupce místo termínů jsou v tomto případě samotné zdroje). Vytvoření křížového pohledu v tomto případě bylo jednoduché, oproti srovnání s aplikací pro Android, vzhledem k tomu, že prostředí C# poskytovalo vhodné prvky, pro vytvoření křížového pohledu.

Dále bylo potřeba v aplikaci Rozvrh přidat formulář pro úpravu DPH. Ve stávající aplikaci byla hodnota nastavena staticky v kódu, což bylo nevhodné řešení.

4.5 Administrace firemních zákazníků

Posledním úkolem byla administrace firemních zákazníků a poskytování konzultace ohledně serverových produktů Windows, při plnění tohoto úkolu jsem byl propůjčen na oddělení servisu. Zde bylo úkolem reagovat na servisní požadavky a sjednávat urychlenou nápravu. Vzhledem k tomu, že se chystám na certifikační zkoušku na Microsoft Exchange (jedná se o poštovní server), tak mi nejčastěji byly přiřazovány servisní požadavky ohledně Microsoft Exchange.

Nejčastěji jsem pracoval ve virtualizovaném prostředí, které bylo postaveno na řešení od firmy VMware. I úpravě konfigurace ve VMwaru jsem se nevyhnul. Dále bylo důležité procházet události, které se staly na serveru, zejména bezpečnostní události. Poté bylo nutné řešit případné chyby a upozornění.

Dále jsem řešil problematiku poštovního serveru. Nejčastěji se jednalo o problém, že nepřichází pošta. To bylo v mnoha případech zapříčiněno různými problémy. Jeden z problémů byl ten, že v síti zákazníka se rozšířila virová infekce, která rozesílala spam a zákazník se dostal do seznamu spamérů. Zde bylo nutné odstranit infekci v síti a kontaktovat příslušné organizace, aby firmu odstranily ze seznamu. Dalším z problémů byla špatná synchronizace poštovního klienta, kdy pomohla úprava v registrech. Smazání poštovního profilu a opětovné nastavení. Další problémy se vyskytovaly ohledně replikace mezi poštovními servery, kdy byl porušen replikační mechanismus.

Jedním ze zajímavých projektů bylo řešení migrace z cloudové služby Office 365 na Microsoft Exchange 2013 pro Město Petřvald, kdy bylo třeba změnit MX DNS záznamy, zmigrovat poštovní schránky a sdílené kalendáře. Bylo potřeba zjistit, jak je vše nastaveno na Office 365, například zda se pošta z něčí schránky nepřeposílá dále apod.

Jedním z úkolů byla také tvorba skriptů v PowerShellu. PowerShellu se také někdy přezdívá příkazová řádka nové generace. Příkazy jsou nazývány „cmdlety“ a většinou se skládají ze slovesa a podstatného jména (např. Add-ADUser). PowerShell může využívat objektů .NET, tudíž se v tomto prostředí dá snadno programovat. Jedním ze skriptů, které jsem vytvářel, bylo zjištění všech dostupných počítačů v síti a jejich skenování portů. Další skript se zaměřoval na Hyper-V, kdy bylo potřeba hromadně přepínat počítače ze statické paměti na dynamickou paměť apod.

5 Závěr

5.1 Uplatnění teoretické a praktické znalosti

V průběhu výkonu své odborné praxe jsem ve velké míře uplatnil znalosti získané z více předmětů, které jsem absolvoval v průběhu studia. V průběhu praxe byla potřeba znalostí jazyka SQL, Java, C++, C# a také práce s Windows Serverem, administrace Microsoft SQL Serveru a různými serverovými produkty firmy Microsoft. Znalosti, které se týkají programování jsem získal absolvováním předmětů:

- Programovací jazyky (PJ1, PJ2)
- Algoritmy (ALG1, ALG2)
- Úvod do databázových systémů (UDBS)
- Vývoj informačních systémů (VIS)
- Tvorba aplikací pro mobilní zařízení 2 (TAMZ2)
- Databázové a informační systémy (DAIS)

Pro administraci serverů a zejména tvorbu privátního cloudu, jsem získal zkušenosti během dřívější praxe ve firmě Autocont a své zkušenosti v této oblasti jsem dále rozvíjel.

5.2 Scházející znalosti

Znalosti, které mi chyběly, byly následující: Jak funguje protokol SOAP a práce s ním, také bylo potřeba nastudovat autentifikaci v SOAP protokolu. Dále práce s knihovnami v Androidu, zde narážím na kSOAP2 knihovnu [9], kterou jsem použil. Chyběla mi znalost integračních služeb Microsoft SQL Serveru, zde to byla velká kapitola, kterou bylo třeba nastudovat a poté aplikovat v praxi, konkrétně práce s transformačními balíčky. Se scházejícími znalostmi mi velmi pomohl konzultant ze strany firmy.

5.3 Dosažené výsledky v průběhu praxe a jejich zhodnocení

Absolvování odborné praxe hodnotím jako velmi pozitivní a přínosnou zkušenost. Vyzkoušel jsem si pracovat na rozsáhlých projektech a pokročilých technologiích. Také jsem se naučil další programovací techniky. Produkty, které jsem vyvíjel, jsou nasazeny v plném provozu a hojně využívány. Dále jsem měl zkušenost absolvovat procesy ve firmě, co se týče nasazení aplikace, tedy vyřizování a „vyběhání“ jednotlivých oddělení. V oddělení Školicího střediska budu po praxi pokračovat nadále.

6 Reference

- [1] AUTOCONT CZ A.S. *Profil společnosti* [online]. 2014 [cit. 2014-04-25]. Dostupné z: <http://www.autocont.cz/o-spolecnosti/profil-spolecnosti>
- [2] *Windows Data Types* [online]. 2012 [cit. 2014-04-25]. Dostupné z: <http://msdn.microsoft.com/en-us/library/windows/desktop/aa383751%28v=vs.85%29.aspx>
- [3] NORTON, Peter, Scott H CLARK a Peter NORTON. *Peter Norton's new inside the PC*. 1st ed. Indianapolis, Ind.: Sams, c2002, ix, 625 p. ISBN 06-723-2289-7.
- [4] *Integration Services (SSIS) Packages* [online]. © 2014 [cit. 2014-04-25]. Dostupné z: <http://technet.microsoft.com/en-us/library/ms141134.aspx>
- [5] *SQL Server Integration Services* [online]. 2014 [cit. 2014-04-25]. Dostupné z: <http://technet.microsoft.com/en-us/library/ms141026.aspx>
- [6] *Soap Headers Authentication in Web Services* [online]. 19.4.2006 [cit. 2014-04-25]. Dostupné z: http://aspalliance.com/805_Soap-Headers-Authentication-in-Web-Services
- [7] *Využití webových služeb a protokolu SOAP při komunikaci* [online]. 1999- [cit. 2014-04-25]. Dostupné z: <http://www.kosek.cz/diplomka/html/websluzby.html>
- [8] *SOAP Version 1.2* [online]. 27.4.2007 [cit. 2014-04-25]. Dostupné z: <http://www.w3.org/TR/soap12-part0/>
- [9] *KSOAP2* [online]. 2014 [cit. 2014-04-25]. Dostupné z: <https://code.google.com/p/ksoap2-android/>
- [10] *StaggeredGridView* [online]. 2013 [cit. 2014-04-25]. Dostupné z: <https://github.com/maurycyw/StaggeredGridView>
- [11] Cloud služby společnosti Microsoft. *Cloud služby společnosti Microsoft* [online]. © 2014 [cit. 2014-04-25]. Dostupné z: <http://blogs.technet.com/b/technetczsk/p/cloud-computing.aspx>
- [12] *Microsoft Azure for Enterprises* [online]. 1.2.2010 [cit. 2014-04-25]. Dostupné z: <http://msdn.microsoft.com/en-us/magazine/ee309870.aspx>

7 Přílohy

I. CD Bakalářská práce