

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INFORMAČNÍ SYSTÉM PRO ENERGETICKÉ PORADEN- STVÍ

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

LUKÁŠ DAVID

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INFORMAČNÍ SYSTÉM PRO ENERGETICKÉ PORADENSTVÍ

INFORMATION SYSTEM FOR ENERGETIC CONSULTANT SERVICE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

LUKÁŠ DAVID

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. RADEK BURGET, Ph.D.

BRNO 2010

Abstrakt

Bakalářská práce pojednává o informačním systému společnosti BV Consulting. V práci je uveden princip fungování společnosti. Dále jsou zde uvedeny veškeré směrnice společnosti důležité pro chod systému. V práci jsou popsány požadavky společnosti na systém, návrh systému vycházející z těchto požadavků a jeho implementace. Hlavní funkcí systému je poskytnout pracovníkům společnosti jednoduché rozhraní pro správu a evidenci spolupracovníků společnosti, jejich smluv a reportů. Informační systém je založen na technologiích Microsoft Office Access a ASP.NET.

Abstract

This bachelor thesis deals with the information system design for the BV Consulting company. The thesis presents the principles of the company functioning including all the internal regulations that are important for the system design. The thesis describes the requirements on the system, system design based on these requirements and its implementation. The main function of system is providing an interface for administration and registration of employees, their contracts and reports. The implementation is based on Microsoft Office Access and ASP.NET.

Klíčová slova

Informační systém, databáze, diagram případů užití, ER diagram, OLAP, Microsoft Office Access, Microsoft Office Excel, .NET, ASP.NET, Visual Basic, SQL, MDAC/WDAC.

Keywords

Information system, database, use case diagram, ER diagram, OLAP, Microsoft Office Access, Microsoft Office Excel, .NET, ASP.NET, Visual Basic, SQL, MDAC/WDAC.

Citace

Lukáš David: Informační systém pro energetické poradenství, bakalářská práce, Brno, FIT VUT v Brně, 2010

Informační systém pro energetické poradenství

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Radka Burgeta, Ph.D.

.....
Lukáš David
17. května 2010

Poděkování

Na tomto místě bych rád poděkoval svému vedoucímu Ing. Radkovi Burgetovi, Ph.D. za cenné rady při vývoji systému a při psaní bakalářské práce.

© Lukáš David, 2010.

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

Obsah

1	Úvod	3
2	Použité technologie	5
2.1	Microsoft Office Access	5
2.2	Microsofts/Windows Data Access Components (MDAC/WDAC)	6
2.3	.NET Framework	7
2.4	ASP.NET	9
2.4.1	PojemPostBack	9
2.4.2	Bezstavovost protokolu HTTP	10
3	Návrh systému	11
3.1	Společnost BV consulting	11
3.1.1	Princip fungování společnosti	11
3.1.2	Směrnice společnosti	11
3.1.3	Vyplácení provizí	13
3.2	Základní funkce systému	14
3.3	Modely systému	15
3.3.1	Model případu užití	15
3.3.2	Konceptuální model	19
3.4	Logický návrh	21
3.4.1	Chybná data	21
3.4.2	Výsledné schéma databáze	21
4	Implementace	23
4.1	Vytvoření databáze	23
4.2	Implementace administrátorské části systému	23
4.2.1	Implementace GUI	23
4.2.2	Práce s Microsoft Office Excel	24
4.2.3	Práce s databází	24
4.2.4	Určení pozice spolupracovníka	24
4.2.5	Implementace případů užití	25
4.3	Implementace uživatelské (webové) části systému	27
4.3.1	Funkce	27
4.3.2	Zvolená platforma a návrh	27
4.3.3	Rozbor jednotlivých vrstev	28

5	Testování a Analýza získaných dat	31
5.1	Testování	31
5.1.1	Testování administrátorské části	31
5.1.2	Testování webové části	32
5.2	Analýza získaných dat	32
5.2.1	Požadavky BV Consulting	32
5.2.2	Systémová podpora	32
5.2.3	Návrh metody	33
5.2.4	Získané informace	33
6	Závěr	35
A	Obsah CD	37
B	Celkový model databáze	38

Kapitola 1

Úvod

V dnešní době dochází k čím dál většímu propojení počítačů a internetu s každodenním životem. Každým dnem se zvětšuje počet uživatelů internetu a s tím rostou nároky na množství informací uložených na internetových serverech. Tyto informace je nutné efektivně ukládat, uchovávat a dále zpracovávat. Jako příklady je možné uvést internetové bankovníctví, internetové obchody, dnes populární sociální sítě a podobně. Řešení všech těchto požadavků nabízí informační systémy. Informační systém ze své podstaty abstrahuje fyzický systém (např. banky) a napomáhá ke zjednodušení jeho správy.

Cílem této práce je osvětlit čtenáři postup návrhu a implementace informačního systému pro společnost BV Consulting. Jedná se o systém pro evidenci spolupracovníků, seminářů, výpočet provizí a mezi-provizí. Hlavním účelem systému je pomoc sekretářce společnosti s administrací těchto úkonů. Výsledný systém je tedy ve formě desktopové aplikace, běžící na jednom počítači. Nicméně nad tímto systémem bylo implementováno jednoduché webové rozhraní pro jednotlivé spolupracovníky, které může posloužit jako šablona k dalšímu možnému vývoji.

První kapitola je věnována Úvodu. Druhá kapitola se věnuje použitým platformám a to *Microsoft Office Access*, který byl použit pro vývoj desktopové části aplikace. Dále pak vysvětlení rozhraní *MDAC/WDAC*. Poslední částí je vysvětlení *.Net* frameworku a jeho části ASP.NET, na jehož platformě je založena webová část systému.

Třetí kapitola se věnuje návrhu systému. Její první část je věnována vysvětlení principu fungování společnosti BV Consulting. Jsou zde rozebrány veškeré směrnice důležité pro chod společnosti. Dále jsou zde uvedeny požadavky na informační systém a z nich vyplývající model případů užití a ER diagram. V poslední části kapitoly je uveden logický návrh systému, ve kterém jsou uvedeny problémy s daty, jež vznikly v průběhu implementace, a také způsob vyřešení těchto problémů. Dále je zde uvedeno výsledné schéma databáze.

Čtvrtá kapitola se věnuje implementaci informačního systému. V první části je uveden postup vytvoření databáze a implementace administrátorské části systému. Zde je rozebrána implementace GUI, práce s Excelem, práce s databází a detailnější rozbor složitějších případů užití. Druhá část je věnována rozboru funkcí, návrhu a implementaci webové části informačního systému.

První část páté kapitoly se věnuje testování systému. Zde jsou popsány postupy jednotlivých testů a jejich výsledky. Druhá část se zabývá rozбором implementované metody získávání znalostí z informačního systému. V této části jsou uvedeny požadavky zástupců společnosti, jejich rozbor a výsledný návrh metody vycházející z těchto požadavků. Dále jsou zde uvedeny výsledky analýzy získaných informací.

Závěrečná šestá kapitola se věnuje zhodnocení výsledné práce.

Kapitola 2

Použité technologie

2.1 Microsoft Office Access

Microsoft Office Access je součástí balíčku Microsoft Office. Microsoft Office Access je počítačová aplikace od společnosti Microsoft pro vytvoření a správu relačních databází. Microsoft Office Access se používá zejména u malých firem, neboť jeho hlavním použitím jsou desktopové databázové aplikace běžící na jednom počítači. Součástí Microsoft Office Access je propracované vývojové prostředí umožňující jednoduchou tvorbu a správu databázových objektů. Databáze v Microsoft Office Access je kolekce všech tabulek, formulářů, reportů, dotazů, maker a modulů, jež dohromady tvoří výslednou databázovou aplikaci.

Mezi významné výhody patří možnost importu nebo přímého spojení s daty z jiné Access databáze, Excelu, Outlooku, XML, HTML a podobně. Dále aplikace umožňuje přes rozhraní ODBC importovat nebo připojit data z databázových systémů Oracle, MySQL, PostgreSQL a Microsoft SQL Server.

Při vlastní tvorbě aplikace je možno využít velkého množství vestavěných maker, které vystačí pro většinu základních operací, jako je otevření formuláře, přechod na další záznam, vytvoření nového záznamu, otevření tiskové sestavy a podobně. V případě, že je potřeba vytvořit složitější operaci, nabízí Access možnost vytvořit vlastní modul případně modul třídy. Programovacím jazykem je Visual Basic for Applications s využitím základních knihoven Accessu. Pro programování vlastních modulů obsahuje Access programové prostředí *Microsoft Visual Basic*. V tomto prostředí je možné přidat reference na další dostupné knihovny (např. Microsoft Excel Object Library) a využít je k další práci. Microsoft Office Access tak umožňuje spolupráci s Outlookem, PowerPointem, Excelem případně i programy mimo balíček Microsoft Office.

Microsoft Office Access obsahuje několik základních knihoven. Použití těchto knihoven je podobné i přesto, že se knihovny od sebe značně liší. Skutečnost, že máme více knihoven je dána požadavky na různé způsoby vytvoření, přístupu a správy databáze.

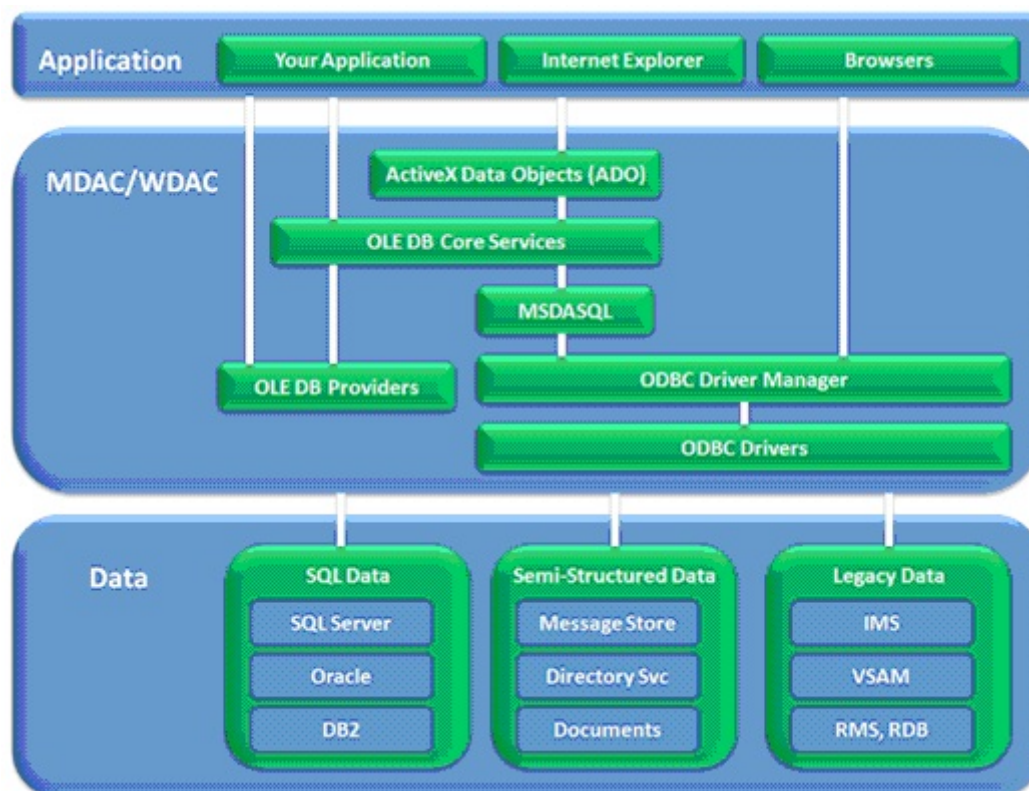
Základní knihovny Microsoft Office Accessu [8] [3]:

- Microsoft Access Object Library: Tato knihovna zajišťuje mechanismus vytvoření a správy aplikace. Knihovna je základní součástí Microsoft Office Access a není nutné jí zvlášť nahrávat.

- Microsoft Data Access Object (DAO): Tato knihovna je také základní součástí Microsoft Office Access a umožňuje tvorbu a správu databáze. Dále knihovna obsahuje nástroje pro jednotlivé operace nad databází jako je například práce s tabulkami, vykonávání dotazů a podobně.
- Microsoft ActiveX Data Objects (ADO): Tato knihovna vznikla za účelem poskytnutí rozhraní (spojení) pro práci s databází Microsoft Access ostatním prostředím. Tuto knihovnu je také možno využít uvnitř samotné aplikace Microsoft Access pro připojení a správu databáze.
- Microsoft ADOX (ADO Extensions): Jak plyne z názvu, jedná se o knihovnu rozšiřující objektový a programový model knihovny ADO zejména v oblasti Data Definition Language a Bezpečnosti.

2.2 Microsofts/Windows Data Access Components (MDAC/WDAC)

Informace uvedené v této kapitole jsou volně převzaty z [5]. MDAC (někdy označované také jako WDAC) je rozhraní, které poskytuje programátorům možnost vytvářet aplikace, které přistupují a pracují s datovými zdroji jednotným způsobem. Datovými zdroji je zde míněno jakýkoliv relační databázový systém, souborový systém, adresářová služba a podobně. Architektura MDAC/WDAC se skládá ze tří komponent a to ADO, OLE DB a ODBC viz obrázek 2.1 .



Obrázek 2.1: Architektura MDAC/WDAC. Převzato z [5]

Jednotlivé komponenty rozhraní MDAC/WDAC zahrnují více podkomponent. Jednotlivé komponenty a jejich použití je uvedeno v následujícím textu (výčet jednotlivých podkomponent jsem převzal z oficiálních stránek firmy Microsoft [5]):

- ADO (zahrnující ADOMD a ADOX). ADO je v hierarchii MDAC/WDAC nejbližší vlastní aplikaci. Pro přístup k datovému zdroji využívá rozhraní OLE DB nebo ODBC.
- OLE DB (zahrnující OLE DB Core Services, SQL Server OLE DB Provider, Oracle OLE DB Provider, OLE DB Provider for ODBC Drivers, Data Shape Provider and Remote Data Provider).
Rozhraní OLE DB pracuje na nejnižší úrovni v hierarchii přístupu k datům. Přes toto rozhraní je možné přistupovat jak k datům z databází, tak k souborovému systému, adresářovým službám a podobně.
- ODBC (zahrnující ODBC Driver Manager, SQL ODBC Driver and Oracle ODBC Driver): Na rozdíl od rozhraní OLE DB slouží toto rozhraní pouze k přístupu k relačním databázovým systémům.

2.3 .NET Framework

Jedná se o softwarový framework určený pro OS Microsoft Windows sloužící pro vývoj webových a desktopových aplikací. Vyšel již v několika verzích počínaje verzí .NET framework 1.0 až po verzi .NET Framework 4. Všechny tyto verze jsou zpětně kompatibilní, což znamená, že v případě vytvoření aplikace ve verzi 1.0 je zajištěno, že nám pojede i s využitím vyšších verzí.

V případě OS GNU/Linux existuje několik projektů snažících se implementovat všechny funkce .NET frameworku tak, aby aplikace vytvořené na této platformě byly spustitelné i na Unixových operačních systémech. Jako příklad je možné uvést projekty Mono, dotGNU, Portable .NET [9].

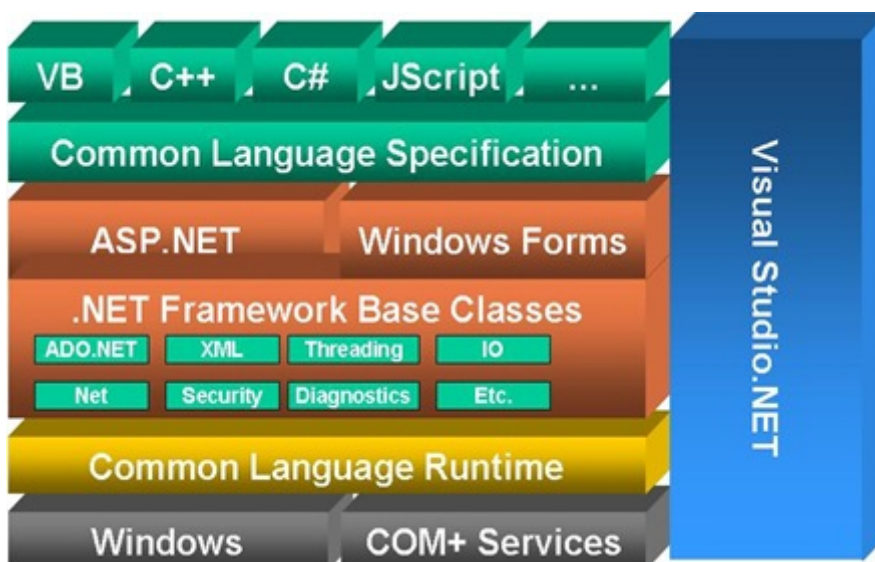
.NET podporuje řadu programovacích jazyků jako C#, VB .Net, C++, J#, Pascal a podobně. Proces kompilace námi napsaného programu (knihovny) je obdobný Javě. Námi napsaný program je pomocí překladače přeložen do mezikódu IL (obdoba Byte kódu v Javě). Výsledkem tohoto překladače je buď spustitelný *exe* soubor nebo knihovna *dll*. Tomuto souboru se také říká Assembly a je to minimální jednotka v .NET, která může být dále používána v jiných programech v .NET. Takto zkompileovaný soubor (knihovna) je potom přenositelný na jakýkoliv počítač, kde je nainstalován .NET framework (v tom spočívá jeho nezávislost na hardware) a zde může být spuštěn pomocí CLR, který jej přeloží do strojového jazyka příslušného počítače a vykoná [2]. Architektura .NET frameworku je uvedena na obr. 2.2.

Vysvětlení architektury .NET framework :

- Common Language Runtime: nejnižší vrstva .NET frameworku pracující přímo s operačním systémem. Jedná se o prostředí, které se postará o vykonání (pomocí překladače JIT) aplikace v mezikódu na daném počítači. Dále se stará o správu paměti pomocí tzv. garbage collector, z čehož vyplývá jedna velká výhoda a to ta, že při programování

v .NET se nemusíme starat o správu paměti, jako je tomu například v C++. Dále se CLR stará o správu výjimek a bezpečnost.

- .NET framework base classes : Jak už plyne z názvu jedná se o množinu knihoven jednotlivých tříd. .NET framework má rozsáhlou základnu knihoven srovnatelnou s knihovnamy Javy. K logickému uspořádání těchto knihoven se používají tzv. namespaces, které vytváří stromovou hierarchii jednotlivých knihoven. Množina všech knihoven se dělí do dvou podmnožin a to *Base Class Library* a *Framework Class Library* [6].
 - Base Class Library: tvoří jádro .NET frameworku. Obsahuje základní funkce pro práci se soubory, pro práci s databázi, XML dokumenty.
 - Framework Class Library: Jedná se o množinu knihoven obsahující BCL knihovnu a spoustu dalších knihoven. Obsahuje veškeré knihovny využitelné v programování v .NET . Zde uvedu pouze ty nejvýznamnější a to jsou:
 - * *ASP.NET*: Množina knihoven pro tvorbu dynamických webových aplikací (podrobněji viz kapitola 2.4).
 - * *ADO.NET*: Množina knihoven umožňující jednotný přístup k datovým zdrojům jako jsou relační databáze, souborový systém a podobně.
 - * *WinForms*: Grafické aplikační programové rozhraní pro tvorbu desktopových aplikací.
- Common language specification (CLS): Jedná se o množinu základních vlastností (pravidel), které musí dané objekty splňovat. CLS pravidla definují podmnožinu společných systémových typů, způsob jejich deklarace, použití a správy při vlastním překlada. Tím, že CLS definuje základní pravidla jednotlivých objektů, umožňuje jazykovou součinnost danou tím, že se vývojáři programující v různých jazycích, mohou spolehnout právě na tyto základní vlastnosti a díky nim se programy, napsané v různých jazycích spolu domluví [4].



Obrázek 2.2: .NET Framework. Převzato z [7]

2.4 ASP.NET

Informace uvedené v této kapitole jsou volně převzaty z [1]. Jedná se o součást .NET frameworku pro tvorbu dynamických webových aplikací a služeb. ASP.NET je nástupce předchozí technologie ASP (Active Server Pages). Hlavním přínosem ASP.NET je fakt, že do tvorby webových aplikací přenesl prvky známé z tvorby desktopových aplikací. Podobně jako u desktopových aplikací se výsledná stránka v ASP.NET skládá z ovládacích prvků (tlačítka, kalendáře, select boxy apod.), tyto ovládací prvky mají své vlastnosti (výška, šířka, barva). Tyto prvky na základě uživateli akce vyvolávají události (stisk tlačítka, změna textu apod.). Na pozadí toho všeho je tedy vyzpělý objektový model, kdy každý ovládací prvek je po softwarové stránce objektem. Tato skutečnost umožňuje vývojářům využít k tvorbě webových aplikací pokročilá vývojová prostředí (Visual Studio apod.) a usnadnit tak jejich vývoj.

Problémem je ale to, že ASP.NET je založen na objektech a událostech, kdežto klientem je pořád webový prohlížeč, který zná pouze HTML, XHTML a javascript. Tento problém je vyřešen tím, že se objekty vyskytují pouze na straně serveru. Ke klientovi se pomocí protokolu HTTP posílá vygenerovaná HTML stránka, obsahující základní webové ovládací prvky.

2.4.1 PojemPostBack

Jedná se o funkci, která slouží k tomu, aby server reagoval na události, ke kterým dochází na straně klienta. Princip je znázorněn na obrázku 2.3. Dojde-li k události (např. stisk tlačítka) na ovládacím prvku, který měl při generování HTML kódu stránky zapnutou funkci PostBack, je pomocí javascriptu zavolán server, kterému jsou předány parametry dané události. Na straně serveru je znovu sestaven objektový model stránky, daná událost je rozpoznána a obsloužena. Po dokončení obsluhy dané události je vygenerována nová podoba HTML stránky, která je poslána zpět klientovi.



Obrázek 2.3: PostBack

2.4.2 Bezstavovost protokolu HTTP

Vzhledem k tomu, že při každé události (mezi jednotlivými PostBacky) je na straně serveru pokaždé znovu sestavován objektový model stránky, vyvstává problém, jak zajistit stav objektů dané stránky nad bezstavovým protokolem HTTP. ASP.NET řeší tento problém kombinací HTML a Javascriptu pomocí dvou základních technik:

- *ViewState* Tato technika spočívá v tom, že si server po každém sestavení objektového modelu stránky zjistí stav jednotlivých ovládacích prvků a tento stav pošle zakódovaný a digitálně podepsaný ve skrytém vstupním poli prohlížeče. Tento zakódovaný řetězec je pak při další PostBack události poslán na server, kde slouží k rekonstrukci stavu ovládacích prvků. Výhody tohoto řešení jsou v tom, že využívá pouze HTML. Nevýhodou je skutečnost přesunu většího množství dat po síti.
- *Session State* Veškeré informace jsou uloženy na straně serveru a klientovi se posílá pouze jednoznačný identifikátor těchto informací. Využívá se u toho cookies případně je identifikátor součástí URL. Výhodou je menší množství dat odesílaných mezi serverem a klientem. Nevýhodou je pak větší zatěž serveru.

Kapitola 3

Návrh systému

3.1 Společnost BV consulting

3.1.1 Princip fungování společnosti

Společnost BV consulting (dále jen "společnost") poskytuje energetické poradenství pro firmy a domácnosti v rámci České republiky. Princip fungování společnosti je založen na systému Multi-level-marketingu, kdy má společnost síť spolupracovníků, jež tvoří strukturu podobnou pyramidě. Tito spolupracovníci zprostředkovávají smlouvy o změně dodavatele elektřiny (dále jen "smlouvy"), za které dostávají provize. Výše těchto provizí je dána směrnicí společnosti. Společnost spolupracuje s jinými firmami (dále jen "partnery"), jejichž služby nabízí a posléze za to dostává provize. Určitou část těchto peněz pak společnost přerozdělí svým spolupracovníkům.

3.1.2 Směrnice společnosti

Směrnice společnosti jsou rozděleny do několika podkapitol. V tabulkách, obsahujících hodnoty výše provizí a počtu požadovaných bodů na jednotlivé pozice, nejsou uvedeny skutečné hodnoty, nýbrž pouze hodnoty ilustrační v rámci zachování firemního tajemství.

Pozice spolupracovníka

Pozice každého spolupracovníka záleží na několika aspektech. Každý spolupracovník začíná na pozici T1. Spolupracovníci dále zprostředkovávají smlouvy pro společnost. Tyto smlouvy jsou dvojího druhu, a to smlouvy s firmami, nebo domácnostmi. Za každou smlouvu má spolupracovník 1,5 bodu. Po dosažení určitého počtu bodů dosáhne spolupracovník, pokud splní i další kritéria, vyšší pozice. Tato pozice se přiznává vždy k prvnímu dni následujícího měsíce. Počet bodů a kritéria potřebné k jednotlivým pozicím jsou uvedeny v tabulce **3.1**.

Každý řádek tabulky **3.1** odpovídá jedné pozici, která je určena hodnotou prvního sloupce. V druhém sloupci je požadovaná hodnota bodů, které musí spolupracovník dosáhnout, aby mohl být na dané pozici. Ve třetím sloupci je požadovaný počet vlastních bodů (do této hodnoty se nepočítají body ze spolupracovníkovy struktury), které musí spolupracovník splnit, aby byl na dané pozici. V dalších čtyřech sloupcích (T1, T2, T3, T4) jsou uvedeny počty požadovaných spolupracovníků ve struktuře na pozici, která je dána jménem sloupce.

Pozice	Body	Vlastní body	T1	T2	T3	T4
T1	0	0	0	0	0	0
T2	50	0	0	0	0	0
T3	250	0	0	0	0	0
T4	750	100	3	2	0	0
T5	1500	100	10	3	1	0
T6	10000	100	30	4	2	1
T7	30000	100	100	6	4	2
BV akcionář	100000	200	50	6	4	2

Tabulka 3.1: Pozice

Provize domácnosti

Velikost provizí za domácnosti je dána tabulkou 3.2. První sloupec tabulky odpovídá skutečnosti, že velikost provize za domácnost nezávisí pouze na pozici spolupracovníka, ale také na počtu smluv uzavřených daný měsíc všemi spolupracovníky společnosti. Hodnoty tohoto sloupce jsou zvoleny tak, že aby výše provizí byla dána tímto řádkem, musí být počet smluv celé firmy větší než hodnota sloupce na daném řádku. Hodnoty ostatních sloupců odpovídají jednotlivým pozicím a provizím na těchto pozicích.

Smlouvy za měsíc BV	T1	T2	T3	T4	T5	T6	T7	T8	GR
0	15	16	17	18	19	21	23	25	30
500	19	20	23	24	25	30	40	45	50
1200	21	27	32	38	42	47	55	65	75

Tabulka 3.2: Provize domácnosti

Provize firmy - nízké napětí

Firmy s nízkým napětím jsou firmy do spotřeby 20MWh za rok. U těchto firem záleží provize na dvou faktorech, a to na pozici spolupracovníka a na distribuční sazbě. Výše jednotlivých provizí je uvedena v tabulce 3.3. Skutečnost, že v tabulce chybí pozice vyšší jak T5 je dána tím, že společnost zatím nevypsala výši těchto provizí. Vlastní funkci systému to však neohrozí, neboť zatím největší pozice v systému je právě T5. Na pozici T6 je ovšem potřeba mnohonásobně více bodů, tudíž čas na doplnění informací je dostačující, navíc vlastní rozšíření spočívá při tomto návrhu pouze v doplnění tabulky o dané sloupce.

Provize firmy - vysoké napětí

U firem s vysokým napětím záleží provize na dvou faktorech, a to na pozici spolupracovníka a na odběru firmy. Výše jednotlivých provizí je uvedena v tabulce 3.4. Odběr firmy je uveden v prvním sloupci tabulky a jeho hodnota určuje, jaký minimální odběr musí mít firma, aby výše provize byla dána hodnotami v tomto řádku. Skutečnost, že v tabulce chybí pozice vyšší jak T5, je dána, stejně jako u provizí s nízkým napětím, tím, že společnost zatím nevypsala výši těchto odměn.

Distribuční sazba	T1	T2	T3	T4	T5	GR
C01d	10	12	14	16	18	17
C02d	15	17	19	21	23	22
C03d	20	25	30	35	40	37
C25d	20	25	30	35	40	37
C26d	20	25	30	35	40	37
C35d	20	25	30	35	40	37
C45d	20	25	30	35	40	37
C55d	20	25	30	35	40	37
C56d	20	25	30	35	40	37
C62d	15	17	19	21	23	22

Tabulka 3.3: Provize firmy - nízké napětí

Odběr	T1	T2	T3	T4	T5	GR
20	40	50	60	70	80	75
100	40	50	60	70	80	75
110	44	55	66	77	88	82
120	48	60	72	84	96	90
130	52	65	78	91	104	97
140	56	70	84	98	112	105
150	60	75	90	105	120	112
160	64	80	96	112	128	120
170	68	85	102	119	136	127
180	72	90	108	126	144	135
190	76	95	114	133	152	142
200	80	100	120	140	160	150
300	100	120	140	160	180	170
400	140	160	180	200	220	210
500	180	200	220	220	240	230

Tabulka 3.4: Provize firmy - vysoké napětí

3.1.3 Vyplácení provizí

Vyplácení provizí probíhá ve dvou fázích. Nejdříve se vyplatí 15% provize a posléze 85% provize a meziprovize.

15% provize

Vyplácení 15% provizí probíhá na konci každého měsíce. Vyplácí se smlouvy, jejichž *datum uzavření* je mezi patnáctým dnem dva měsíce zpět, konče čtrnáctým dnem měsíc zpět. Tudíž například když vypočítávám provize na konci března, počítají se smlouvy, jejichž datum uzavření je od 15. ledna do 14. února.

85% provize

Vyplácení těchto provizí probíhá vždy v polovině aktuálního měsíce. Vyplácí se provize a meziprovize za smlouvy, u nichž je *datum zahájení odběru* rovno prvnímu dni předcháze-

jícího měsíce. Takovéto smlouvy mohli být uzavřeny v rozdílných měsících. Toto je dáno rozdílnou vypovídající lhůtou jednotlivých smluv, danou předchozím dodavatelem elektřiny. Tyto skutečnosti je nutno brát v potaz, neboť provize za jednotlivé smlouvy závisí nejenom na pozici daného spolupracovníka, ale také na počtu smluv uzavřených v daném měsíci v rámci celé společnosti.

Modelová situace: V případě výpočtu provizí k datu 15. 3. 2010 se proplácí smlouvy s datem zahájení odběru rovným 1. 2. 2010. Datum uzavření smlouvy může být u takovýchto smluv v rozmezí čtyř měsíců. Jedná se o období druhého až pátého měsíce zpětně. V našem případě tedy v období od 1. 9. 2009 do 31. 12. 2009.

Meziprovize

Meziprovize dostávají vedoucí za smlouvy, které uzavřel jiný spolupracovník z jejich struktury. Meziprovize se v celkové výši vyplácí zároveň s 85% provizí, což znamená, že se z meziprovize nevyplácí 15% záloha. Výši meziprovize je možné popsat matematickou rovnicí:

$$meziprovize_{vedouci} = provize_{vedouci} - provize_{spolupracovnik}$$

Meziprovize je tedy dána rozdílem částky, kterou by dostal za smlouvu daný vedoucí, kdyby ji uzavřel on sám, od částky, kterou za ni dostal spolupracovník pod ním v hierarchii. Například když spolupracovník na pozici T1 uzavře smlouvu, dostane za ni 15 Kč. Jeho vedoucí, který je na pozici T3 by za ni dostal 17 Kč. Výsledek je, že meziprovize pro tohoto vedoucího činí 2 Kč. Pokud tento vedoucí má také svého vedoucího, tak se jeho meziprovize počítají úplně stejným způsobem.

3.2 Základní funkce systému

- Import databáze natypovaných smluv
 - Asistentka každý týden dostane od spolupracovníků smlouvy v papírové podobě, které zatypuje do systému partnerů. Přes webové rozhraní si asistentka může kdykoliv nechat vyexportovat soubor ve formátu xlsx obsahující takto natypované smlouvy (dále jen "reporty"). Systém musí poskytovat rozhraní pro import tohoto souboru.
- Import databáze zprovizovaných smluv
 - Smlouvy, za které dostane společnost od partnera provizi, přijdou ve speciálním souboru ve formátu xlsx. Systém musí poskytovat rozhraní pro import tohoto souboru.
- Výpočet provizí
 - Systém musí umožňovat výpočet provizí a meziprovizí jednotlivých spolupracovníků na základě naimportovaných reportů a smluv. Provize se počítají dvakrát v měsíci (15% zálohová platba, 85% zbylá platba). Výsledkem takového výpočtu jsou provizní reporty jednotlivých spolupracovníků ve formátu xlsx.
- Aktualizovatelná databáze spolupracovníků

- Každý spolupracovník musí projít seminářem a složit test, kde dostaneme jeho základní údaje. Systém musí evidovat číslo spolupracovníka, které je mu přiděleno na semináři, jeho jméno a číslo vedoucího.
 - V rámci vzdělávací akademie spolupracovníci absolvují boční semináře. Tyto účasti musí systém evidovat (datum, téma semináře, typy přednášek, cena a způsob úhrady).
 - Systém musí poskytovat rozhraní pro správu jednotlivých spolupracovníků. Tímto je myšleno jednoduché procházení mezi spolupracovníky, zjišťování jejich pozice, počtu smluv apod. Dále možnost tisku seznamu jednotlivých smluv.
 - Spolupracovníci mohou tvořit obchodní síť. Každý spolupracovník musí mít přístup k náhledu na svoji obchodní síť. Systém musí umožňovat jednotlivým spolupracovníkům prohlížet smlouvy ostatních spolupracovníků ve struktuře.
- Aktualizovatelná databáze klientů
 - Vzniká a aktualizuje se při importu natypovaných smluv.
 - Databáze seminářů
 - Systém by měl umět nahlašovat seminaristy na školení. Evidovat formy úhrady, záloh, umožnit přehlašování na jiné termíny atp.
 - Databáze podkladů, pomůcek, kalkulátorů
 - Databáze odkazů na materiály ke stažení. Rozdělené do sekcí pro vedoucí, seminaristy, obchodníky, vedení společnosti, administrativu.

3.3 Modely systému

Z výše uvedených požadavků na funkčnost systému a data, které má systém spravovat, byly navrženy následující dva modely.

3.3.1 Model případu užití

Model případů užití je uveden na obrázku 3.1. Rozbor jednotlivých případů užití:

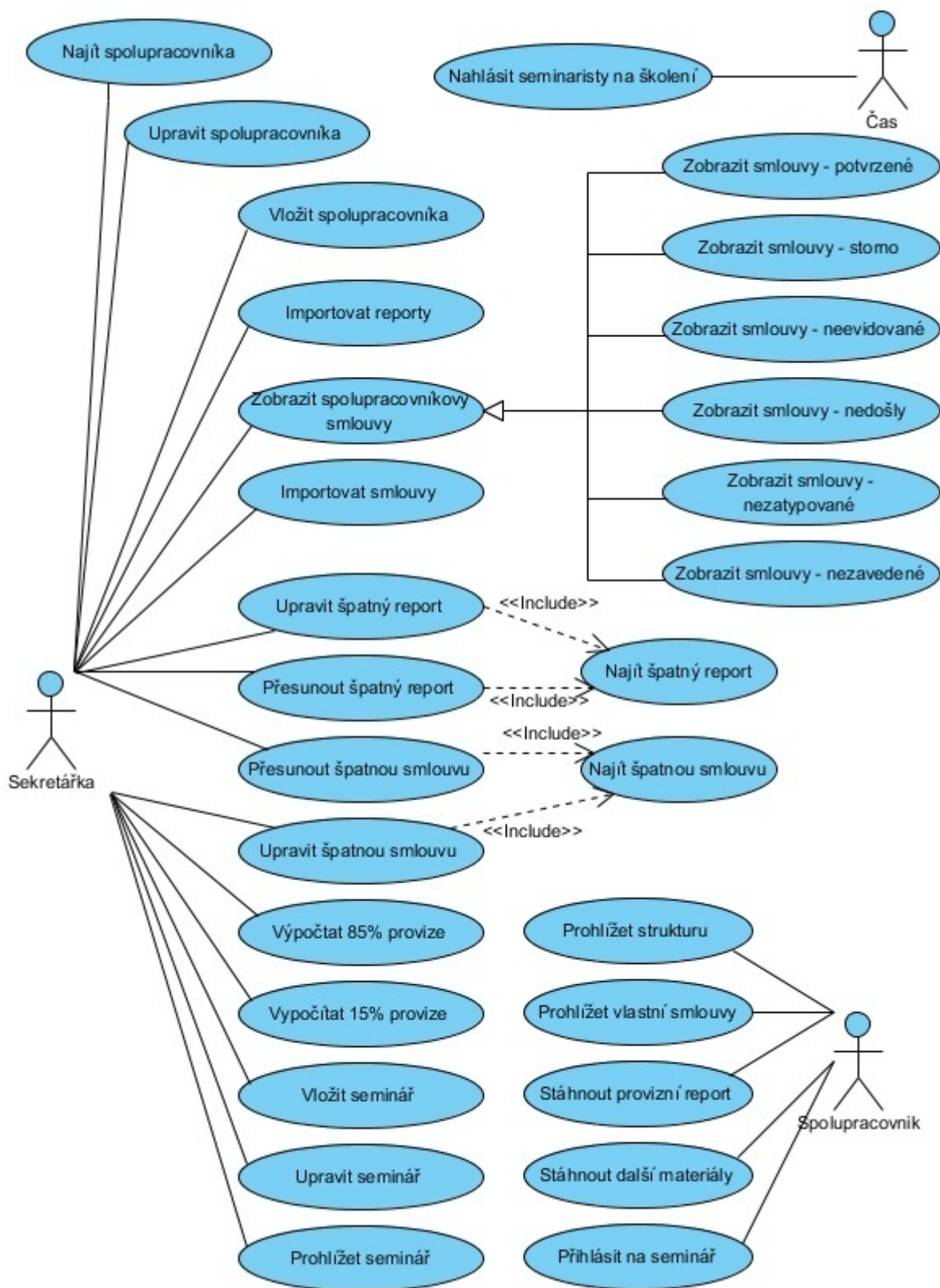
- *Najít spolupracovníka*: Najde spolupracovníka podle jeho osobního čísla a v případě, že daný spolupracovník neexistuje, vypíše příslušné upozornění.
- *Upravit spolupracovníka*: Umožňuje upravit jméno spolupracovníka a číslo vedoucího, protože je možné, aby spolupracovník přešel k někomu jinému do struktury. V případě špatného čísla vedoucího vypíše systém příslušné hlášení.
- *Vložit spolupracovníka*: Umožňuje vložit nového spolupracovníka. Systém zároveň kontroluje správnost údajů, a v případě chyby vypíše příslušné chybové hlášení. V případě úspěšného vložení systém zjistí, jestli v tabulce *Report špatné* neexistují nějaké reporty, odpovídající nově vloženému spolupracovníkovi. Jestliže takovéto reporty nalezne, přesune je do tabulky *Report*, neboť tyto reporty jsou již správné (jejich cizí klíč odpovídá právě vkládanému spolupracovníkovi). Pro každý takto přesunutý report systém zjistí, jestli mu neodpovídá nějaká smlouva z tabulky "Potvrzené smlouvy špatné", a pokud ano, danou smlouvu přesune do tabulky správných smluv.

- *Importovat reporty:* Zajišťuje import reportů. Takovýto import začíná výběrem souboru ve formátu xlsx, který obsahuje jednotlivé reporty pro import. Takovýto soubor musí obsahovat data přesně daného formátu což znamená, že musí mít přesný počet sloupců v předem daném pořadí. Po úspěšném výběru souboru dojde k vlastnímu importu. Vzhledem k tomu, že není zaručena správnost všech reportů v souboru, případně to, že již v systému nenachází, systém každý vkládaný report kontroluje. Každý takto vkládaný report může dopadnout jedním z následujících způsobů:
 - import reportu proběhne v pořádku
 - report již je v systému v tabulce *Report* a nedojde tudíž k jeho importu
 - hodnota cizího klíče daného reportu neodpovídá žádnému spolupracovníkovi, report je importován do tabulky *Report špatné*
 - hodnota cizího klíče daného reportu neodpovídá žádnému spolupracovníkovi a zároveň report již je v tabulce *Report špatné*. V tomto případě k importu nedojde.

Po zpracování všech reportů systém vypíše statistiku importu.

- *Zobrazit spolupracovníkovy smlouvy:* Umožňuje zobrazit spolupracovníkovy smlouvy, dělené podle kategorií uvedených v use case diagramu.
- *Importovat smlouvy:* Zajišťuje import smluv. Podobně jako import reportů, začíná import smluv výběrem příslušného souboru formátu xlsx. Obsah tohoto souboru musí být v přesně daném formátu což znamená, že musí mít přesný počet sloupců v předem daném pořadí. Vzhledem k tomu, že není zaručena existence reportu odpovídajícího dané smlouvě, systém každou smlouvu kontroluje a v případě, že příslušný report neexistuje, ji importuje do tabulky *Potvrzené smlouvy špatné*. Po dokončení systém vypíše statistiku importu. Import smluv se od importu reportů liší v jedné podstatné části, a to té, že při importu smluv systém nekontroluje, jestli již daná smlouva je v systému, neboť k této situaci nemůže dojít (vždy se importují pouze nové smlouvy).
- *Upravit špatný report:* Umožňuje upravit některé atributy u špatných reportů a posléze tyto změny uložit.
- *Přesunout špatný report:* Umožňuje přesun špatných reportů do tabulky správných reportů v případě, že již report odpovídá některému ze spolupracovníků.
- *Upravit špatnou smlouvu:* Umožňuje upravit některé atributy u špatných smluv a posléze tyto změny uložit.
- *Přesunout špatnou smlouvu:* Umožňuje přesun špatných smluv do tabulky správných smluv v případě, že již smlouva odpovídá některému z reportů.
- *Vypočítat 85% provize:* Vytvoří adresářovou strukturu jednotlivých spolupracovníků, kterým se mají vyplácet provize. Každému takovému spolupracovníkovi vytvoří vlastní provizní výpis v podobě souboru formátu xlsx.
- *Vypočítat 15% provize:* Stejně jako u 85% provizí, vytvoří adresářovou strukturu s provizními výpisy jednotlivých spolupracovníků, jemuž se mají vyplácet 15% provize.
- *Vložit seminář:* Umožňuje vložit nový seminář. Pokud existuje seminář stejného data, vypíše příslušné chybové hlášení.

- *Upravit seminář:* Umožňuje opravit jakýkoliv z atributů semináře.
- *Prohlížet seminář:* Umožňuje prohlížet jednotlivé semináře, dále pak spolupracovníky, jež jsou přihlášení k daným seminářům.
- *Prohlížet strukturu:* Umožňuje spolupracovníkům prohlížet svoji vlastní strukturu. U každého spolupracovníka ve struktuře umožňuje zobrazit seznam všech jeho smluv rozdělených do dvou kategorií a to, smlouvy v pořádku a smlouvy s chybou.
- *Prohlížet vlastní smlouvy:* Umožňuje spolupracovníkům prohlížet si svoje vlastní smlouvy, jejich statusy, stav.
- *Stáhnout provizní report:* Umožňuje stáhnout provizní reporty za jednotlivé měsíce, kdy daný spolupracovník uzavřel alespoň jednu smlouvu, popř. někdo z jeho struktury uzavřel alespoň jednu smlouvu.
- *Stáhnout další materiály:* Umožňuje stáhnout aktuální směrnice společnosti a další materiály informující o chodu společnosti.
- *Přihlásit na seminář:* Umožňuje přihlásit se k jednotlivým seminářům.



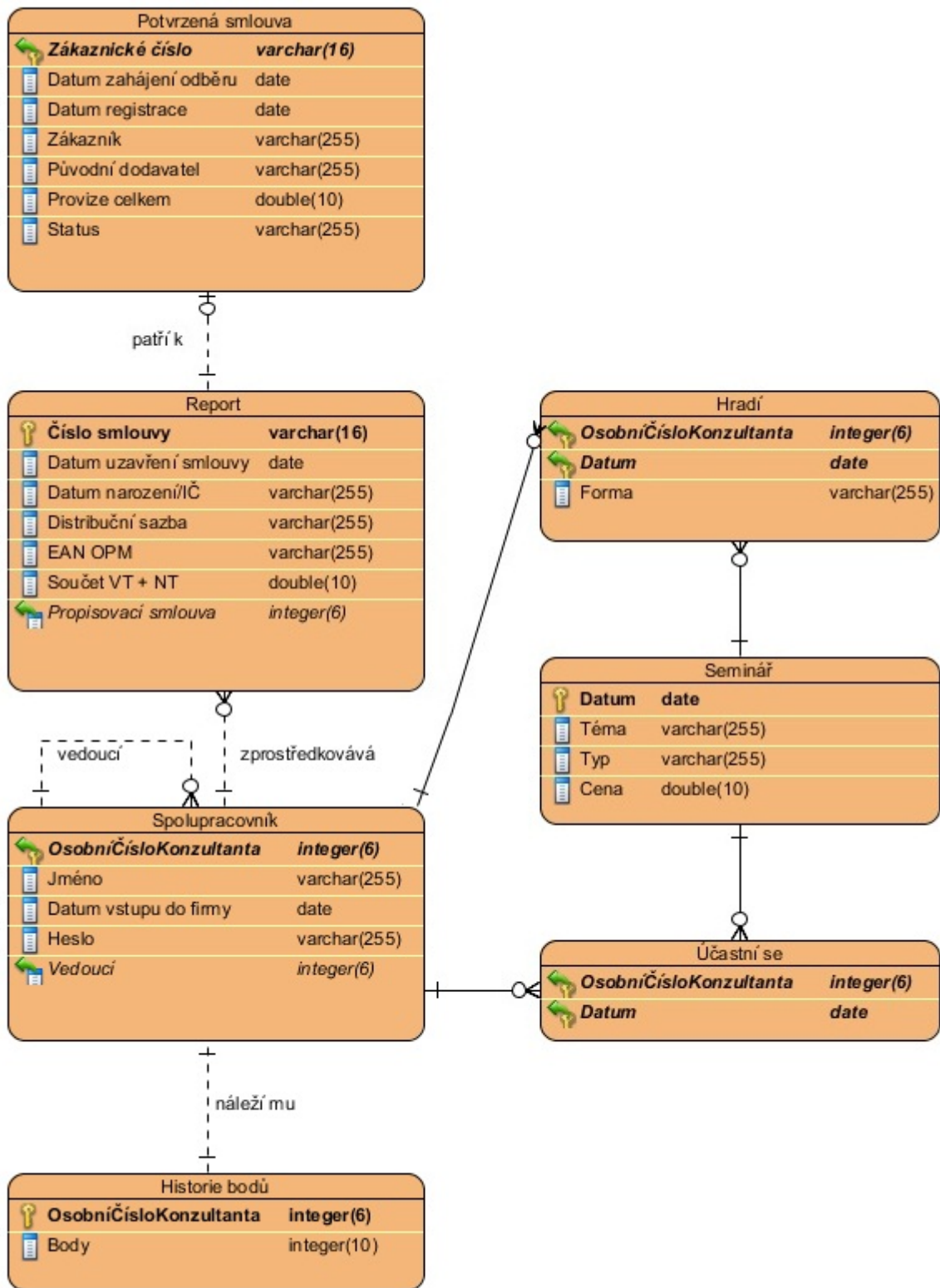
Obrázek 3.1: Use case diagram

3.3.2 Konceptuální model

Konceptuální model je uveden ve formě ER diagramu na obrázku 3.2. Model je tvořen sedmi entitními množinami. U entitních množin *Potvrzená smlouva* a *Report* jsou uvedeny pouze atributy důležité z hlediska funkčnosti systému, neboť entitní množina *Potvrzená smlouva* má 23 atributů a entitní množina *Report* má 99 atributů. Veškeré tyto atributy musí být v systému uvedeny k případnému dohledání, nicméně z pohledu konceptuálního návrhu jsou bezvýznamné. Z pohledu normalizace je výsledný model v 3. normální formě. Tato skutečnost je dána tím, že veškeré řádky všech tabulek v databázi jsou natolik unikátní, že v rámci těchto tabulek neexistuje žádný neklíčový atribut, který by nebyl plně závislý na jediném kandidátním klíči tj. primárním klíči. Dále se v návrhu nevyskytuje žádná tranzitivní závislost neklíčového atributu na primárním klíči.

Rozbor a vysvětlení volby všech entitních množin a vztahů mezi nimi:

- entitní množina *Spolupracovník*: reprezentuje jednotlivé spolupracovníky společnosti. Tato entitní množina vstupuje do 5 vztahů, neboť je jádrem celého systému. Prvním vztahem je vztah reflexní mezi dvěma entitami této entitní množiny. Tento vztah zajišťuje funkčnost struktur spolupracovníků v rámci společnosti, kdy každý ze spolupracovníků může mít pod sebou libovolný počet dalších spolupracovníků a kdy každý spolupracovník má pouze jednoho vedoucího. Druhý vztah je mezi entitními množinami *Spolupracovník* a *Report*. Kardinalita a členství daného vztahu jsou určeny na základě skutečnosti, že jednotliví spolupracovníci mohou patřit k libovolnému množství reportů a že každý report patří právě k jednomu spolupracovníkovi. Třetí vztah je vztah mezi entitními množinami *Spolupracovník* a *Historie bodů*. Kardinalita vztahu je 1:1, což je dáno tím, že každému spolupracovníkovi odpovídá maximálně jeden záznam v entitní množině *Historie bodů*. V posledních dvou vztazích, mezi entitní množinou *Spolupracovník* a *Seminář*, je použita tzv. vazební entitní množina a to z toho důvodu, že kardinalita těchto vztahů byla M:M. Vztahy s vazebními entitními množinami *Hradí* a *Účastní* se reprezentují skutečnost, že systém musí evidovat formu plateb jednotlivých seminářů a také účasti spolupracovníků na jednotlivých seminářích.
- entitní množina *Report*: reprezentuje jednotlivé reporty. Tato entitní množina vstupuje do 2 vztahů. První vztah byl vysvětlen výše u entitní množiny *Spolupracovník*. Druhý vztah je mezi entitními množinami *Report* a *Potvrzené smlouvy*. Kardinalita a členství tohoto vztahu jsou určeny na základě skutečnosti, že jednotlivým reportům odpovídá maximálně 1 potvrzená smlouva.
- entitní množina *Potvrzená smlouva*: reprezentuje konkrétní smlouvy. Obsahuje informaci o konečném stavu smlouvy (atribut *status*), tzn. informaci, jestli smlouva byla zákazníkem opravdu dodržena, případně jestli byla stornována apod. Tyto informace jsou pak důležité k tvorbě provizních reportů.
- entitní množina *Seminář*: Entitní množina reprezentující jednotlivé semináře. Vstupuje do dvou vztahů (*Účastní se* a *Hradí*) s entitními množinami *Spolupracovník*. Oba vztahy byly vysvětleny výše v textu.
- entitní množina *Historie bodů*: Obsahuje počet bodů, který se přičítá danému spolupracovníkovi do kariéry. Důvod vzniku této entitní množiny je uveden v části logického



Obrázek 3.2: Entity relationships diagram

návrhu.

3.4 Logický návrh

Při návrhu databáze je potřeba zohlednit ještě jednu skutečnost, a to jsou problémy s *chybnými daty*. Jejich rozbor je uveden v následujících podkapitolách.

3.4.1 Chybná data

Při realizaci informačního systému došlo k nashromáždění obrovského množství chybných dat (jedná se o přibližně o 900 reportů a 950 smluv). Chybovost těchto dat vychází z ER diagramu. Jedná se o *reporty* se špatnými nebo prázdnými cizími klíči, které tudíž nejde přiřadit k určitému spolupracovníkovi. Dále se jedná o smlouvy, u kterých je špatný cizí klíč. Na základě požadavku od zadavatele, aby tyto smlouvy byly evidovány v systému, jsou v systému vytvořeny dvě pomocné tabulky *Report špatné* a *Potvrzené smlouvy špatné*, které obsahují veškeré tyto smlouvy. Dále je v systému implementováno rozhraní pro úpravu těchto smluv a reportů.

Historie bodů

Vzhledem ke skutečnosti, že v systému existuje velké množství smluv, které nelze nikomu přiřadit, ale které někdo uzavřel, výsledné provize a pozice spolupracovníků, vypočítané systémem, neodpovídají skutečnosti. Nicméně většinu těchto smluv již nikdo nedohledá, tudíž byla do systému přidána další pomocná tabulka *Historie bodů*, která obsahuje body spolupracovníků, jež nějaké body získali v době před koncem roku 2009. Tyto body se pak těmto spolupracovníkům započítají do pozice a do provizí, čímž dojde k eliminování problému se špatnými smlouvami z předchozích let. U smluv počínaje rokem 2010 se již nevyskytují problémy s chybnými daty.

3.4.2 Výsledné schéma databáze

Vzhledem k výše zmíněným skutečnostem obsahuje výsledný systém 13 tabulek. Grafický model všech těchto tabulek je uveden v příloze na obrázku **B.1**. U tabulek *Report*, *Report špatné*, *Potvrzené smlouvy* a *Potvrzené smlouvy špatné* nejsou uvedeny všechny atributy, neboť je jejich počet velký a do obrázku by se nevešly. Seznam tabulek se stručným popisem (shrnutím) je uveden zde:

- *Spolupracovník*: Tabulka reprezentující jednotlivé spolupracovníky. Má 4 atributy.
- *Report*: Tabulka reprezentující jednotlivé reporty. Má 99 atributů.
- *Potvrzené smlouvy*: Tabulka reprezentující konečné smlouvy. Má 23 atributů.
- *Historie bodů*: Tabulka vycházející z požadavku, aby někteří spolupracovníci měli určitý počet bodů přičtený do kariéry. Vysvětlení v kapitole **3.4.1**. Tabulka má 2 atributy.
- *Hradí*: Tabulka reprezentující vztah mezi tabulkami *Spolupracovník* a *Seminář*. Má 2 atributy.

- *Účastí se*: Tabulka reprezentující vztah mezi tabulkami Spolupracovník a Seminář. Má 3 atributy.
- *Seminář*: Tabulka reprezentující jednotlivé semináře. Má 4 atributy.
- *Pozice*: Hodnoty a sloupce tabulky korespondují s tabulkou 3.1. Tabulka slouží k uchování informací, které jsou potřeba k určení pozice spolupracovníka.
- *Odměny domácnosti Bicorn*: Obsahuje výši jednotlivých odměn za domácnosti u partnera Bicorn.
- *Odměny firmy nízké napětí*: Obsahuje hodnoty odměn za firmy s nízkým odběrem. Jedná se o firmy s roční spotřebou menší jak 20 kWh.
- *Odměny firmy vysoké napětí*: Obsahuje hodnoty odměn za firmy s vysokým odběrem.
- *Potvrzené smlouvy špatné*: Tabulka se všemi špatnými smlouvami. Má stejně atributů jako tabulka *Potvrzené smlouvy*.
- *Report špatné*: Tabulka se všemi špatnými reporty. Má stejně atributů jako tabulka *Report*.

Kapitola 4

Implementace

4.1 Vytvoření databáze

Vzhledem k žádosti zástupců firmy BV Consulting byla pro implementaci systému zvolena databázová aplikace Microsoft Office Access. Data do systému byla vložena nainportováním souborů formátu *xlsx*, obsahujících jednotlivé spolupracovníky, smlouvy a reporty. Dále byly vytvořeny funkce pro inicializaci systému do podoby, odpovídající integritním omezením kladeným na systém, tzn. přesunutí špatných reportů a smluv do pomocných tabulek.

4.2 Implementace administrátorské části systému

Administrátorskou částí je myšlena část systému, která je implementována v Microsoft Office Access. Tato část systému umožňuje veškeré operace, které jsou popsány v diagramu případů užití u kterých je účastníkem sekretářka. V této části textu je vysvětlen postup vývoje systému, dále jsou zde vysvětleny veškeré případy užití uvedené v kapitole [3.3.1](#).

4.2.1 Implementace GUI

Tvorba GUI v Microsoft Office Access je velice obdobná tvorbě aplikací v prostředích jako jsou *Visual Studio*, *Netbeans* apod. Základním prvkem pro tvorbu GUI je formulář. Do takového formuláře pak můžeme vložit jednotlivá tlačítka, textová pole, seznamy apod. U každého takového prvku můžeme nastavit obsluhu jednotlivých událostí. Pro obsluhu událostí můžeme použít vestavěná makra, případně si napsat vlastní proceduru ve Visual Basicu.

Při implementaci a návrhu GUI jsem postupoval tak, že jsem si nejdříve navrhl obecný vzhled formulářů, odpovídající vzhledu webových stránek společnosti (barva, styl). Tento návrh jsem předvedl zástupcům společnosti, kterým se líbil. Posléze jsem navrhl obecnou strukturu systému.

Struktura systému

Systém se skládá celkově z dvanácti formulářů.

- *Menu*: Úvodní formulář, obsahující menu pro volbu požadované operace.

- *Spolupracovníci*: Formulář jednotlivých spolupracovníků. Formulář zobrazuje informace o jednotlivých spolupracovnících.
- *Špatné reporty*: Poskytuje rozhraní pro úpravu špatných reportů a jejich přesun do správných reportů.
- *Špatné smlouvy*: Poskytuje rozhraní pro úpravu špatných smluv a jejich přesun do smluv správných.
- *Menu vložit*: Obsahuje menu pro výběr, jestli chceme vložit spolupracovníka nebo seminář.
 - * *Modifikace spolupracovník*: Poskytuje rozhraní pro vložení nového spolupracovníka.
 - * *Vložit seminář*: Poskytuje rozhraní pro vložení nového semináře.
- *Výpočet provizí*: Poskytuje rozhraní pro import smluv a reportů, výpočet 15% a 85% provizí.
- *Analýza dat*: Obsahuje možnosti nastavení jednotlivých analýz a jejich vykonání.
- *Administrace seminářů*: Poskytuje rozhraní pro kompletní správu seminářů, účastníků seminářů, evidenci plateb.

4.2.2 Práce s Microsoft Office Excel

System ve velkém množství operací pracuje se soubory ve formátu xlsx případně xls. Veškerá tato práce je řízena programovým kódem na pozadí systému. K práci s těmito soubory systém využívá knihovnu *Microsoft Excel 12.0 Object Library*, kterou je možné připojit přímo do Access aplikace. Můžeme tak přímo z Accessu otevírat, vytvářet či modifikovat soubory tohoto formátu, čehož se využívá zejména u operací importu reportů (smluv), výpočtu provizí a analýzy dat.

4.2.3 Práce s databází

Práce s databází je v Microsoft Office Access velice podobná nástroji phpMyAdmin v oblasti správy databázového systému MySQL. Uživatel může procházet jednotlivé tabulky, mazat, vkládat a upravovat jednotlivé řádky, aniž by k tomu bylo potřeba implementovat zvláštní rozhraní. Dále aplikace obsahuje nástroje pro přímý import dat do jednotlivých tabulek, případně export dat z tabulek. Výše zmíněné funkce jsou funkce vestavěné do vývojového prostředí Microsoft Office Access. Systém nicméně implementuje několik programových funkcí pro práci s databází. Zejména se jedná o možnost programového importu jednotlivých smluv a reportů, dále pak o kompletní správu seminářů a jednotlivých spolupracovníků. Veškeré tyto operace jsou hlídány, aby nedošlo k narušení referenční integrity.

4.2.4 Určení pozice spolupracovníka

Jedná se o jednu z nejdůležitějších funkcí systému. Tato funkce je využívána při procházení jednotlivých spolupracovníků, při výpočtu provizí apod. Pozice spolupracovníka se určuje na základě jeho osobních bodů a bodů ze struktury. Za každou smlouvu má spolupracovník 1,5 bodu. Počet vlastních smluv je určen jedním sql dotazem. Pro výpočet počtu smluv ve struktuře je v systému funkce, která rekurzivně prochází celý podstrom spolupracovníků, jež jsou pod daným spolupracovníkem. U každého takového spolupracovníka se zjistí počet jeho smluv a přičte se k celkovému počtu smluv. Po zjištění výsledného počtu vlastních a strukturních smluv se v tabulce *Pozice* zjistí příslušná pozice.

4.2.5 Implementace případů užití

Vzhledem k tomu, že se jedná o rozbor administrátorské části, jsou zde uvedeny pouze případy užití, u nichž je účastníkem sekretářka. Při implementaci jednodušších případů užití se systém drží postupů popsanych v kapitole 3.3.1. Jedná se konkrétně o případy užití *Najít spolupracovníka, Upravit spolupracovníka, Vložit spolupracovníka, Importovat reporty, Zobrazit spolupracovníkovi smlouvy, Importovat smlouvy, Upravit špatný report, Přesunout špatnou smlouvu, Vložit seminář, Upravit seminář.*

Výpočet 85% provizí a meziprovizí

Implementace Princip výpočtu 85% provizí byl uveden v kapitole 3.1.3. Výsledkem výpočtu 85% provizí a meziprovizí je, na pokyn zadavatele, adresářová struktura obsahující provizní reporty jednotlivých spolupracovníků, kteří mají nárok na odměnu v daném období. Ukázka provizního reportu je uvedena na obrázku 4.1. Jedná se o soubor formátu xlsx, obsahující tři listy s celkovou provizí, provizí vlastní a provizí strukturální (meziprovizí). Vlastní výpočet probíhá ve dvou fázích.

	A	B	C	D	E	G
7						
8	Provizní výpis podrobný - vlastní					
9						
10	Partner	Produkt	Datum sepsání smlouvy	Jméno Klienta	Pozice	Provize 85%
11	Bicorn	Domácnost	14.11.2009	Jiří Kupka	T1	16,15 Kč
12	Bicorn	Domácnost	14.11.2009	Roman Hájek	T1	16,15 Kč
13	Bicorn	Domácnost	14.11.2009	Radek Tomášek	T1	16,15 Kč
14	Bicorn	Domácnost	14.11.2009	Škopík Zdeněk	T1	16,15 Kč
15	Bicorn	Domácnost	14.11.2009	Skoupá Zdeňka	T1	16,15 Kč
16	Bicorn	Firma	14.11.2009	Astim s.r.o	T1	32,15 Kč
17						112,90 Kč
18						
19	Provizní výpis podrobný - storna					
20						
21	Partner	Produkt	Datum sepsání smlouvy	Jméno Klienta		
22	Bicorn	Domácnost	14.11.2009	Přimová Romana		
23	Bicorn	Domácnost	14.11.2009	Ingnáček Karel		
24						
25	Provizní výpis podrobný - oprava					
26						
27	Partner	Produkt	Datum sepsání smlouvy	Jméno Klienta	Chyba	
28	Bicorn	Domácnost	14.11.2009	Štveráček Luboš	NEEVIDUJEME	
29						

Obrázek 4.1: provizní Report

První fáze jsou pouze provize za vlastní produkci. Tato fáze začíná předpočítáním počtu smluv uzavřených v rámci celé firmy v jednotlivých počítaných měsících. Tyto hodnoty jsou pak použity při určování výše odměn, které závisí i na výkonnosti celé firmy. Dále se vhodným sql dotazem zjistí, kteří spolupracovníci uzavřeli alespoň jednu smlouvu k

tomuto období. Pro každého takového spolupracovníka systém zavolá funkci, jež se postará o samotný výpis provizí za vlastní produkci. Tento výpis probíhá v několika krocích. Nejdříve se vypíše do provizního reportu pouze smlouvy s domácnostmi, u kterých nedošlo ke stornování, případně jiné chybě. Při výpisu těchto smluv je potřeba dodatečně spočítat pozici daného spolupracovníka v daném měsíci, neboť v jednotlivých měsících může být spolupracovník na různých pozicích. Zde je použita optimalizace, která spočívá v seřazení všech smluv podle data uzavření, což umožňuje počítat pozici spolupracovníka v rámci jednoho měsíce pouze jedenkrát.

Posléze dojde k výpočtu provizí za firmy. Tento výpočet probíhá obdobně jako výpočet provizí za domácnosti, jediný rozdíl je při konečném určení provize viz směrnice společnosti. Dalším krokem je výpis všech stornovaných smluv z daného období. Posledním krokem je výpis chybných smluv.

Druhá fáze je výpočet meziprovizí. Zde prvním krokem je výpočet meziprovizí ze správných smluv. Celý tento výpočet probíhá následovně. Systém zjistí, jakou provizi za danou smlouvu dostal její zprostředkovatel, tyto údaje uloží do struktury a zavolá rekurzivně příslušnou funkci pro svého vedoucího. V této funkci se zjistí pozice vedoucího v daném měsíci, odpovídající provize a posléze, je-li jeho provize vyšší, vypočítá se jeho meziprovize a zapíše se do provizního reportu. Tento postup se opakuje do doby, než systém narazí na fiktivního spolupracovníka, reprezentujícího celou společnost. Tento jediný nemá svého vedoucího a zde rekurze končí a funkce se vrací zpět k původnímu spolupracovníkovi. Tento proces se opakuje pro všechny smlouvy s domácnostmi a firmami. V případě stornovaných a chybových smluv dochází taktéž k volání rekurzivní funkce, nicméně zde již bez jakéhokoliv počítání, pouze se zápisem těchto smluv jednotlivým vedoucím do provizních reportů.

Výpočet 15% provizí

Implementace Princip výpočtu 15% provizí je uveden v kapitole 3.1.3. Stejně jako u 85% provizí je výsledkem adresářová struktura, obsahující jednotlivé provizní reporty ve formátu xlsx. Na rozdíl od 85% provizí obsahuje výsledný report pouze dva listy s celkovou provizí a s výpisem vlastních smluv. Z 15% provize se nevyplácí meziprovize. Vlastní implementace probíhá tak, že se nejdříve zjistí výkon celé firmy ve dvou počítaných měsících, který se později používá pro výpočet výše odměn. Posléze systém zjistí pomocí sql dotazu, kteří spolupracovníci uzavřeli alespoň jednu smlouvu v daném období. Pro tyto spolupracovníky systém zavolá funkci, jež každému vytvoří jeho provizní report. Vlastní tvorba reportu probíhá v několika krocích. Nejdříve se vypíše všechny smlouvy domácností, které jsou v pořádku, posléze firmy. Dále se vypíše smlouvy stornované a na závěr smlouvy s jinou chybou. Výsledný report se uloží do adresáře odpovídající danému spolupracovníkovi.

4.3 Implementace uživatelské (webové) části systému

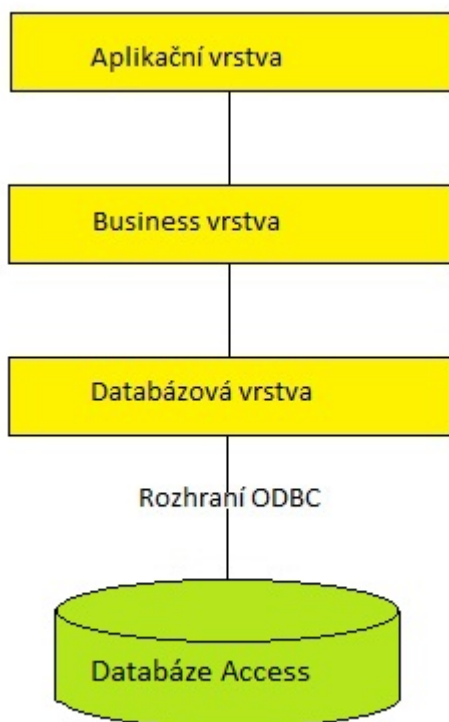
4.3.1 Funkce

Funkcí webové části informačního systému je poskytnutí jednoduchého uživatelského rozhraní jednotlivým spolupracovníkům společnosti. Pomocí tohoto rozhraní si jednotliví spolupracovníci mohou prohlížet veškeré smlouvy, jež zprostředkovali a které jsou v systému. Dále pak zobrazit informace o své současné pozici ve společnosti (pozice, počet smluv v pořádku, počet lidí ve struktuře apod.). Systém také umožňuje správu seminářů. Poslední funkcí systému je kompletní mapa struktury daného spolupracovníka.

4.3.2 Zvolená platforma a návrh

Vzhledem k tomu, že jako databázový systém byla použita aplikace Microsoft Office Access, jsem jako platformu pro web zvolil .NET framework, resp. jeho část ASP.NET s použitím programovacího jazyka C#. Volbu této platformy jsem učinil kvůli snadnějšímu přepsání administrátorské části systému do .NET, než např. do PHP.

Systém se skládá ze tří vrstev. Jedná se o tyto vrstvy *aplikační*, *business* vrstva a *databázová* vrstva. Hierarchie jednotlivých vrstev je uvedena na obrázku 4.2. Výhodou tohoto návrhu je budoucí jednodušší předělání systému. Tímto je myšleno například přechod z webové aplikace na desktopovou aplikaci případně změna databázového systému apod.



Obrázek 4.2: Architektura systému

4.3.3 Rozbor jednotlivých vrstev

Aplikační vrstva

Jedná se o vrstvu tvořící grafické uživatelské rozhraní pro jednotlivé spolupracovníky. Vrstva je tvořena sedmi webovými stránkami poskytujícími příslušné uživatelské rozhraní. Jako ukázka je, na 4.3, uveden náhled na vzhled stránky jednotlivých smluv daného spolupracovníka.

Popis jednotlivých stránek

- *Prihlaseni.aspx*: Na tuto stránku jsou přesměrováni všichni spolupracovníci, kteří doposud nebyli autorizováni. Stránka poskytuje rozhraní pro zadání loginu a hesla a jeho odeslání k autentizaci. Dále stránka kontroluje, jestli bylo zadáno jak heslo, tak login a v případě, že nebylo, upozorní na to spolupracovníka. V případě, že autentizace proběhne v pořádku, stránka vytvoří příslušné cookies, díky kterému získá daný uživatel přístup do systému.
- *Index.aspx*: Jedná se o úvodní stránku, na kterou se dostanou spolupracovníci po úspěšném vstupu do systému. Stránka obsahuje aktuality společnosti, nejbližší konané semináře. Stránka dále umožňuje odeslání emailu se žádostí o návštěvu jednotlivých seminářů.
- *Profil.aspx*: Obsahuje základní informace o přihlášeném spolupracovníkovi. Jedná se o pozici, počet lidí ve struktuře, počet smluv ve struktuře, počet vlastních smluv a počet vlastních stornovaných smluv.
- *Seminare.aspx*: Zobrazuje semináře, na které je daný spolupracovník přihlášen. Dále u každého semináře zobrazuje status, jestli již byl uhrazen či nikoliv.
- *Smlouvy.aspx*: Na této stránce se zobrazí spolupracovníkovi výčet všech smluv rozdělených do dvou kategorií podle statusu dané smlouvy. V první kategorii jsou všechny smlouvy, u nichž nedošlo k žádné chybě, v druhé kategorii jsou ostatní smlouvy.
- *Soubory.aspx*: Stránka umožňuje stažení některých souborů, týkajících se chodu společnosti např. provizní směrnice apod.
- *Struktura.aspx*: Stránka poskytuje grafický náhled na strukturu spolupracovníka. Spolupracovník vidí všechny spolupracovníky ve své struktuře. U každého takového spolupracovníka si může zobrazit výčet všech jeho smluv rozdělených opět do dvou kategorií podle statusu.

Business vrstva

Poskytuje rozhraní mezi aplikační vrstvou a databázovou vrstvou. Obsahuje 7 tříd. Každá třída poskytuje metody pro obsluhu funkcí, které byly zmíněny u jednotlivých webových stránek. Názvy tříd jsou zvoleny tak, aby odpovídali názvu webové stránky, jež je instancí dané třídy obsluhována. Jedinou výjimkou je třída *My_Dataset*, která slouží jako podpůrná třída k zajištění funkčnosti zobrazení struktury spolupracovníka.

Výčet jednotlivých tříd

- *BS_Prihlaseni*
- *BS_Index*

- *BS_Profil*
- *BS_Seminare*
- *BS_Smlouvy*
- *BS_Struktura*
- *My_Dataset*

The screenshot displays the BV Consulting web application. At the top, there is a navigation menu with links: Úvodní strana, Profil, Struktura, Vlastní smlouvy, Moje semináře, Soubory ke stažení, and Ohlásit se. Below the menu, the page title is "Smlouvy v pořádku:". This is followed by a table listing contracts in order.

Zákazník	Číslo smlouvy	Datum smlouvy
Kneblová Ludmila	777020065	14.12.2009
mančík Radim	777020061	14.12.2009
Vyščor Igor	777020055	14.12.2009
Kubicová Jindřiška	777020070	14.12.2009
Matalová Božena	777020519	14.12.2009
Pospiech Petr	777020078	14.12.2009
Pospiech Petr	777020076	14.12.2009
Šafaříková Irena	777018445	11.12.2009

Below the first table is a pagination bar with the text "1 2 3 4 5 6 7 8 9 10 ...".

The next section is titled "Chybné smlouvy (viz. status)". It contains a table listing contracts with incorrect status.

Zákazník	Číslo smlouvy	Datum smlouvy	Status
Bureš Pavel	777020073	14.12.2009	storno
Sláma Pavel	777020049	14.12.2009	storno
Zatloukal Tomáš	777020043	14.12.2009	nevidujeme
Strouhal Jan	777020552	14.12.2009	storno
Příkryl Hynek	777020555	14.12.2009	storno
Příkryl Hynek	777020554	14.12.2009	nevidujeme

Below the second table is a pagination bar with the text "1 2 3 4".

Obrázek 4.3: ukázka IS

Databázová vrstva

Jedná se o vrstvu, která již přímo pracuje s databází. Vzhledem k tomu, že většina požadavků na databázi spočívá v načtení velkého množství dat, jedná se o nejrobustnější vrstvu návrhu. Veškeré třídy a jejich metody používají pro připojení k databázi rozhraní ODBC a připojovací řetězec (tzn. connection string), jenž určuje databázový systém, ke kterému se jednotlivé metody připojují. Tento připojovací řetězec je definován v konfiguračním souboru

web.config. Z tohoto souboru si ho jednotlivé třídy, pomocí příslušných metod, zjistí a použijí ho k připojení k databázi. Toto řešení má tu výhodu, že v případě změny databázového systému stačí pouze změnit takto nakonfigurovaný připojovací řetězec. Databázová vrstva je tvořena šesti třídami.

Výčet jednotlivých tříd

- *DB_Prihlaseni*: Třída poskytuje jedinou metodu k ověření, jestli v databázi existuje spolupracovník s daným heslem a loginem. K tomuto zjištění používá parametrický sql dotaz, čímž zabrání útoku známému jako *SQL poisoning*, kterým by mohl případný útočník smazat některou z tabulek apod.
- *DB_Profil*: Třída poskytuje metody pro zjištění pozice přihlášeného uživatele, počtu lidí v jeho struktuře, počet smluv (správných, špatných) a počet smluv ve struktuře.
- *DB_Smlouva*: Třída reprezentující jednu konkrétní smlouvu daného spolupracovníka.
- *DB_Smlouvy*: Třída poskytuje metody pro získání kolekce smluv (instance třídy *DB_Smlouva*) daného spolupracovníka.
- *DB_Seminari*: Třída reprezentující jeden konkrétní seminář se statusem, zdali byl již uhrazen.
- *DB_Seminare*: Třída poskytuje metody pro získání všech seminářů, které ještě neproběhly. Dále poskytuje metodu pro získání seminářů konkrétního spolupracovníka.
- *DB_Struktura*: Třída poskytuje jedinou metodu, která vrátí dataset všech spolupracovníků společnosti.

Kapitola 5

Testování a Analýza získaných dat

5.1 Testování

5.1.1 Testování administrátorské části

Testování funkcí spolupracovníka

Testování administrátorské části jsem prováděl v několika krocích, které korespondovaly s vývojem systému. Nejdříve jsem testoval formulář *Spolupracovníci*, který zobrazuje informace o jednotlivých spolupracovnících. Zde jsem postupoval tak, že jsem si vytipoval pár spolupracovníků, kteří měli již nějaké smlouvy uzavřeny a kteří měli alespoň pár aktivních lidí ve struktuře. U každého takovéhoho spolupracovníka jsem si udělal mapu jeho struktury, zjistil počty smluv jednotlivých spolupracovníků a pak zkontroloval kompletní strukturu od spolupracovníků nejnižší až po daného spolupracovníka. Všechny tyto údaje seděli s údaji, jež jsem si mohl nechat zobrazit pomocnými dotazy. Nejdůležitější poznatek z tohoto testování je ten, že funkce, která počítá pozici spolupracovníka k nějakému datu, fungovala dle předpokladů. Tato funkce je základem celého systému, neboť většina operací závisí na pozici spolupracovníka k určitému datu.

Testování výpočtu provizí

Vzhledem k tomu, že funkce na určování pozice byla již řádně otestována, jsem se soustředil na správnost vygenerovaných provizních reportů. Testoval jsem to na měsících, ve kterých se počítaly provize za smlouvy, jejichž počet byl v řádech několika stovek. Po úspěšném vytvoření provizních reportů systémem jsem si u každého období vytipoval přibližně patnáct smluv. Každou takovou smlouvu jsem zkontroloval, jestli je zapsaná v osobní produkci správnému spolupracovníkovi. Dále jestli je zapsána do struktur jednotlivých spolupracovníků, jestli meziprovizí odpovídá požadované výši a podobně. Všechny tyto údaje měly správné hodnoty. Dále jsem si jednoduchým dotazem zjistil seznam všech spolupracovníků, kteří uzavřeli smlouvy v daném období a porovnal ho s adresářovou strukturou, vygenerovanou při výpočtu provizí. Ani zde jsem nenašel žádného spolupracovníka, který by uzavřel alespoň jednu smlouvu a neměl vytvořen vlastní provizní report. V této chvíli jsem také poskytl systém zástupcům společnosti, kteří si ho mohli vyzkoušet a zkontrolovat také, jestli výsledné hodnoty opravdu odpovídají skutečnosti. Odpovědi jsem se ale nedočkal vzhledem k interním problémům společnosti.

Testování dalších funkcí

Testování zbylých funkcí již nebylo tak složité, neboť se jedná o operace vložení seminářů, import reportů apod. Při testování importu reportů a smluv jsem vytvořil několik testovacích souborů s reporty a smlouvami. Posléze jsem vykonal vlastní import a výsledky porovnal s očekávanými výsledky. Všechny výsledky opovídaly. Při testování ostatních operací funkčnost odpovídala požadované funkčnosti.

5.1.2 Testování webové části

Testování webové části systému bylo o poznání snazší, než testování administrátorské části systému. Je to dáno tím, že webová část poskytuje pouze rozhraní, které podává jednotlivým spolupracovníkům informace o jejich pozici ve společnosti (vlastní smlouvy, struktura, smlouvy ve struktuře a podobně).

Informace zobrazované v profilu spolupracovníka přímo korespondují s informacemi, jež si může administrátor zobrazit ve formuláři jednotlivých spolupracovníků. Testování této části tedy proběhlo tak, že jsem pouze porovnal výsledky zobrazované administrátorskou částí s výsledky webové části. Tyto výsledky se zákonitě musely shodovat. Při testování mapy struktury a smluv jednotlivých spolupracovníků ve struktuře jsem využil data, testovaná v administrátorské části systému. Všechny tyto výsledky se shodovaly. Tímto jsem považoval testování za úspěšné.

5.2 Analýza získaných dat

5.2.1 Požadavky BV Consulting

Požadavky zástupců společnosti na metody predikce a získávání znalostí můžeme shrnout do 4 bodů.

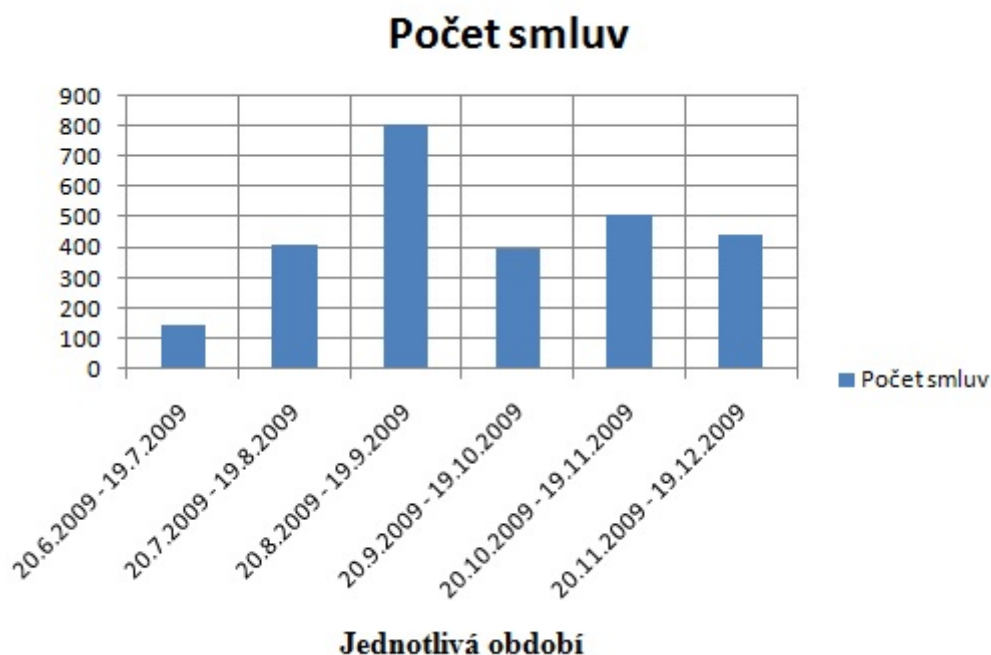
Prvním požadavkem je analýza počtu smluv v závislosti na počasí, ročním období. *Druhým* požadavkem je zkoumání úspěšnosti v jednotlivých regionech v závislosti na mentalitě lidí. *Třetím* požadavkem je zkoumání počtu uzavřených smluv v závislosti na nezaměstnanosti obyvatel. *Čtvrtým* požadavkem je zkoumání, zdali reference v určitém regionu zvětšují úspěšnost v daném regionu v následujících časových obdobích. Tímto je myšlena skutečnost, jestli se v případě, že v jednom měsíci uzavřu v konkrétní oblasti určitý počet smluv, zvětší počet uzavřených smluv v následujícím měsíci.

5.2.2 Systémová podpora

Vzhledem k absenci některých informací v systému jsou některé požadavky implementovány nepřímou. Konkrétně se jedná o první tři požadavky. U prvního požadavku nám schází jakákoliv informace o počasí v jednotlivých dnech (týdnech). U druhého požadavku chybí informace o mentalitě lidí v daných regionech. U třetího požadavku se jedná o nezaměstnanost lidí v daných regionech. Všechny výše uvedené požadavky podporuje systém nepřímou. Pouze čtvrtý požadavek systém podporuje přímo, neboť potřebné informace pro tuto analýzu jsou v systému uchovávány.

5.2.3 Návrh metody

Výsledná metoda je navržena tak, aby poskytovala jakýsi OLAP nástroj, umožňující zjištění vývoje počtu smluv v závislosti na čase, regionu a statusu smluv. Jedná se o třídimenzionální datový model, jehož dimenze jsou čas, region a status smlouvy. Faktem v jednotlivých dimenzích je počet smluv. Systém umožňuje jednotlivé dimenze konfigurovat. V případě času je možno nastavit časová období na týden případně měsíc, dále pak nastavit počet období, která chceme analyzovat a datum počátku analýzy. Region můžeme nastavit podle kraje, města a ulice. V případě poslední dimenze systém umožňuje analyzovat počet všech smluv, počet správných smluv, počet smluv se zvoleným statutem a poměr počtu smluv s konkrétním statutem k počtu všech smluv. Výsledkem takovéto analýzy je soubor formátu xls, který obsahuje jednotlivé tabulky odpovídající analýze a k ní příslušné grafy.



Obrázek 5.1: ukázka analýzy

5.2.4 Získané informace

Navržená metoda podporuje všechny čtyři požadavky společnosti. Tato skutečnost je dána tím, že všechny požadavky se vztahují buď na určitý region nebo na určité časové období. Ukázka výstupu analýzy je uvedena na obr.5.1. Interval této analýzy byl zvolen na jeden měsíc, počet opakování na šest, neboť smlouvy z jiných období se v systému nenachází. Na této analýze je patrná souvislost s prvním požadavkem. Analýza sice přímo neříká, jaký vliv má počasí na počet uzavřených smluv, neboť informace o počasí v systému není, nicméně se dá vyzorovat, že v letním období je počet smluv větší než v podzimním období. Z této skutečnosti lze usoudit, že teplé počasí a delší období světla mají pozitivní vliv na počet uzavřených smluv. Zde je potřeba si ale uvědomit, že množství informací, které máme k dispozici, je nedostačující. Data, která máme k dispozici, jsou pouze z období šesti měsíců a to je, k zjištění některých závěrů, velice krátká doba.

Druhý a třetí požadavek je implementovanou metodou do jisté míry splněn také, neboť je možnost nastavit si různé regiony analýzy. Na základě těchto výsledků a informací o mentalitách (nezaměstnanosti) lidí v různých oblastech je možné usoudit, jaký vliv má daná mentalita (nezaměstnanost) na počet uzavřených smluv. Nicméně i zde je nedostatek informací k nějakým relevantním závěrům, neboť po důkladné analýze jsem zjistil, že téměř 3/4 smluv jsou z města Brna. Tato skutečnost je dle mého názoru dána tím, že firma sídlí v Brně a většina spolupracovníků je z Brna, než tím, že by v Brně byli lidé mnohem přívětivější, případně by zde byla vyšší nezaměstnanost.

Čtvrtý požadavek jako jediný jde určit přímo z databáze, aniž by bylo potřeba dohledávat další informace, neboť všechny potřebné informace k analýze se v databázi nacházejí. Na základě dostupných informací (smluv) jsem nicméně zjistil, že se v jednotlivých oblastech nevyskytuje vzrůstající počet uzavřených smluv, tudíž reference lidí v dané oblasti nemají zřejmě vliv na úspěšnost uzavřených smluv.

Kapitola 6

Závěr

Cílem bakalářské práce bylo vytvořit informační systém pro společnost BV Consulting. Jako platformu jsem, na žádost zástupců společnosti, zvolil Microsoft Office Access a ASP.NET. Při návrhu a tvorbě informačního systému jsem využil znalostí nabytých v předmětech věnujících se problematice informačních systémů. V průběhu implementace byla největším problémem komunikace se zástupci společnosti, která nakonec téměř ustala vzhledem k interním problémům ve společnosti. Důsledkem toho je to, že jsem v posledních dvou měsících tvořil systém zejména dle svého uvážení. Jedná se o kompletní webovou část systému a část administrátorské části. V důsledku těchto skutečností jsem bohužel neměl ke konci práce žádnou zpětnou vazbu od společnosti na výsledný systém a systém se tedy v praxi nakonec nepoužívá.

Při tvorbě jsem kladl důraz na jednoduché a intuitivní ovládání. Systém implementuje všechny uvedené požadavky, některé však pouze rámcově. Časově nejnáročnějšími operacemi jsou výpočty provizí, kde se čas při počtu smluv v řádech několika stovek pohybuje v hodinách, toto by mohlo být námětem pro další vývoj systému. Dalším možným rozšířením systému by mohla být větší provázanost webové části s částí administrátorskou, možnost přes webové rozhraní pořádat vlastní semináře, přihlašovat se na semináře a podobně. Celkově se zde nabízí možnost kompletního předělaní systému pouze do ASP.NET, čímž by bylo možné veškeré operace provádět přes webové rozhraní. Toto rozšíření je možné i u Microsoft Office Access, nicméně každý takový uživatel musí mít tuto aplikaci nainstalovanou, což skýtá problém zaplacení balíku Microsoft Office. Dále je možné systém rozšířit v oblasti bezpečnosti komunikace mezi webovou částí a databází Access za využití SSL a HTTPS.

Bakalářská práce mi vylepšila znalosti informačních systémů. Seznámil jsem se s novými databázovými systémy a celkově jsem získal větší přehled na poli informačních systémů a jejich využití. Dále jsem se lehce seznámil s technologiemi OLAP, což se mi jistě bude hodit v dalším studiu. Při realizaci jsem se také mnohokrát setkal s problémy v komunikaci se zástupci společnosti a ověřil si tak, že ne vždy je možné aplikovat postup známý z předmětů IUS a IDS, a že je někdy nutné při vývoji a návrhu trochu improvizovat.

Literatura

- [1] Běhálek, M.: ASP.NET [online].
<http://www.cs.vsb.cz/behalek/frvs/2005/dotnet/asp/>, 2010.
- [2] Ruttkay, L.: *předmět IW5 - .Net přednáška*. 2010.
- [3] Viescas, J.; Conrad, J.: *Mistrovství v Microsoft Office Access 2007*. Computer Press, 2008, iSBN 978-80-251-2162-7.
- [4] WWW stránky: Common Language Specification [online].
<http://msdn.microsoft.com/en-us/library/12a7a7h3.aspx>.
- [5] WWW stránky: Data Access Technologies Road Map [online].
<http://msdn.microsoft.com/en-us/library/ms810810.aspx>.
- [6] WWW stránky: Difference between BCL and FCL.
<http://social.msdn.microsoft.com/forums/de-DE/netfxbcl/thread/b44f8f72-cc11-457b-8537-b491e871c1a9/>.
- [7] WWW stránky: DotNet Framework Fundamentals [online].
<http://technotes.towardsjob.com/dotnet/dotnet-framework-fundamentals/>.
- [8] WWW stránky: Microsoft Access Libraries [online].
<http://www.functionx.com/vbaccess/Lesson07.htm>.
- [9] WWW stránky: Microsoft .NET a Linux [online].
<http://www.linuxexpres.cz/software/microsoft-net-a-linux>.

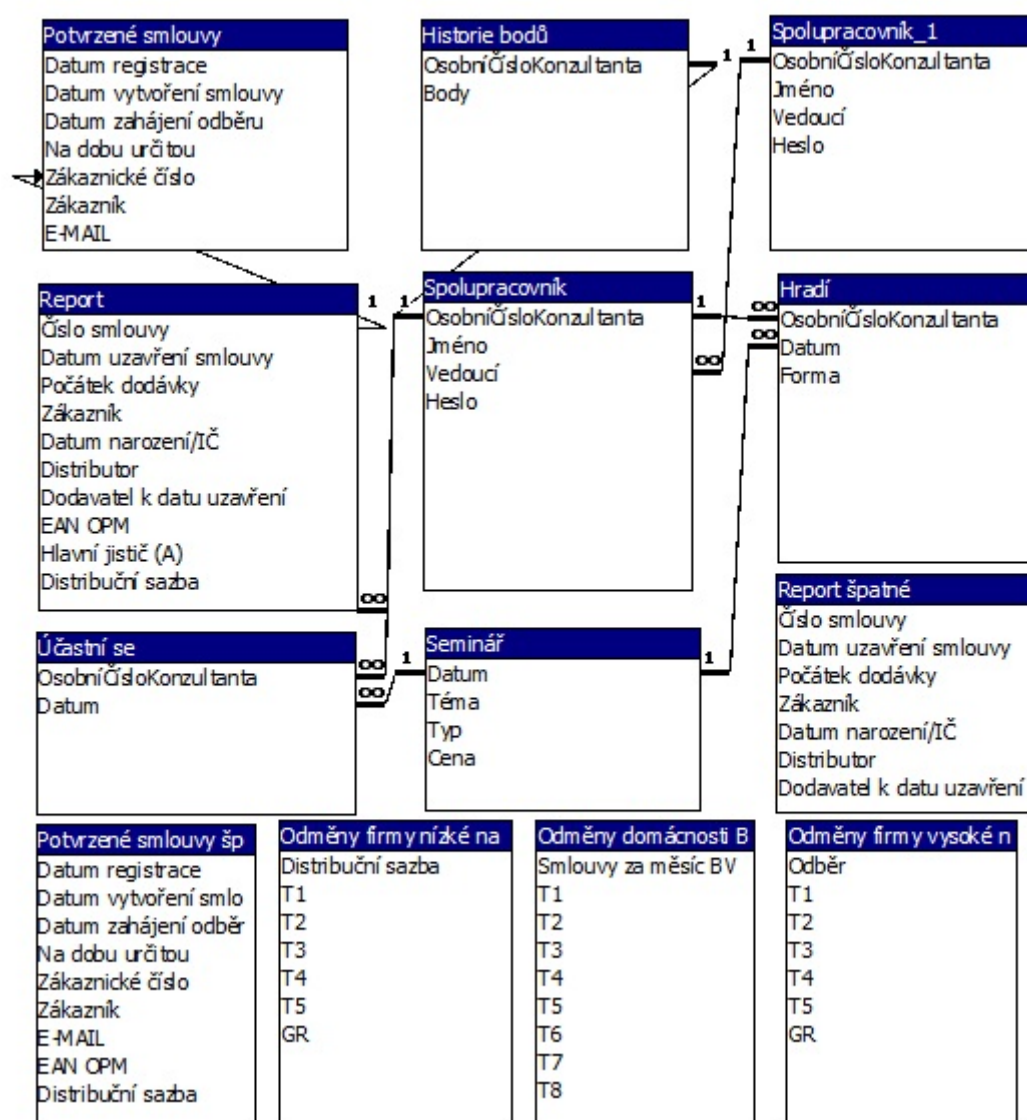
Příloha A

Obsah CD

- Elektronická verze technické zprávy bakalářské práce ve formátu PDF spolu se zdrojovými kódy pro Latex
- Zdrojové kódy systému
- Testovací soubory pro import dat do systému
- Dokumentace, manuál

Příloha B

Celkový model databáze



Obrázek B.1: Výsledný model všech tabulek databáze