

Univerzita Pardubice

Fakulta elektrotechniky a informatiky

Návrh a implementace aplikace typu CRM

Bc. Marek Macák

Diplomová práce

2010

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Bc. Marek MACÁK  
Osobní číslo: I08367  
Studijní program: N2646 Informační technologie  
Studijní obor: Informační technologie  
Název tématu: Návrh a implementace aplikace typu CRM  
Zadávající katedra: Katedra softwarových technologií

### Z á s a d y p r o v y p r a c o v á n í :

Hlavním cílem diplomové práce je návrh a implementace aplikace typu CRM (Customer Relations Management) pro řízení vztahu se zákazníky, vhodnou pro malou a střední firmu. V úvodní, teoretické části bude shrnut význam aplikace pro firemní marketing a vytvořen přehled základních požadavků na aplikaci (např. záznam strukturovaných dat o zákaznících a jejich následné třídění podle konkrétních znaků, tvorba přehledů). Na základě těchto požadavků bude definována základní charakteristika aplikace, její základní funkce a postup implementace. Součástí implementace bude analýza požadavků prostřednictvím UML diagramů pomocí softwarového nástroje Enterprise Architect, návrh datových tříd a jejich grafické vizualizace. Aplikace bude naprogramována v jazyce C# (vývojové prostředí Microsoft Visual Studio) s podporou databázového serveru (MS SQL). Při tvorbě aplikace je třeba důsledně dodržet základní požadavek vhodného uživatelského rozhraní, síťového sdílení aplikace typu klient-server a požadavek na bezpečnost aplikace s důrazem na autentizaci a autorizaci uživatelů.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

GÁLA, Libor, POUR, Jan, TOMAN, Prokop. Podniková informatika. Mgr. Petr Mušálek. 1. vyd. Praha: Grada Publishing, 2006. 484 s. Management v informační společnosti. ISBN 80-247-1278-4. SHARP, John. Microsoft Visual C# 2008 : Krok za krokem. [s.l.] : Computer Press, 2008. 592 s. ISBN 9788025120279. CHLEBOVSKÝ, V. CRM - Řízení vztahů se zákazníky. 1. vydání. Praha: Computer Press, a. s., 2005. ISBN 80-251-0798-1.

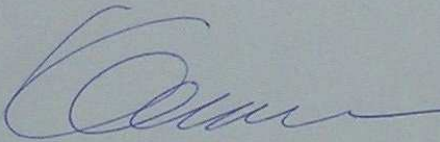
Vedoucí diplomové práce:

**Ing. Jana Holá, Ph.D.**

Katedra zdravotnické informatiky

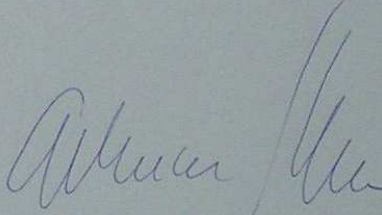
Datum zadání diplomové práce: **30. října 2009**

Termín odevzdání diplomové práce: **21. května 2010**

  
prof. Ing. Simeon Karamazov, Dr.

děkan



  
doc. Ing. Antonín Kavička, Ph.D.

vedoucí katedry

V Pardubicích dne 10. listopadu 2009

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 20. 05. 2010

Bc. Marek Macák

## **SOUHRN**

Obsahem této diplomové práce je popis aplikace typu CRM (řízení vztahů se zákazníky), charakteristika vlastností, funkcionalit a přínosů, které s sebou tento informační systém přináší. Praktická část se zabývá analýzou, návrhem a implementací vzorového programu na řízení vztahů dle požadavků potenciálního zákazníka.

## **KLÍČOVÁ SLOVA**

CRM, řízení vztahů se zákazníky, informační systémy

## **TITLE**

Design and implementation of CRM type application

## **ABSTRACT**

The contents of this work encompass the description of the CRM type application (Customer Relationship Management), its characteristics, functionality and contributions that come with this informational system. The practical part of the paper deals with the analysis, concept and implementation of the sample program for customer relationship management in accordance with the requirements of the prospective client.

## **KEYWORDS**

CRM, Customer relationship management, information systems

## OBSAH

1 Úvod.....	9
2 Teoretická část .....	10
2.1 Informační systémy (IS) .....	10
2.2 CRM, řízení vztahů se zákazníkem.....	12
2.2.1 Funkcionalita CRM.....	14
2.2.2 Integrace CRM s ERP .....	14
2.2.3 Integrace CRM s jinými aplikacemi .....	15
2.2.4 Základní vrstvy (části) CRM.....	16
2.2.5 Mýty o CRM .....	18
2.3 Zavedení CRM systému.....	19
2.3.1 Stanovení firemních požadavků .....	19
2.3.2 Kritické faktory úspěchu .....	20
2.3.3 Přínosy nasazení CRM v podniku .....	21
2.3.4 Volba optimálního systému .....	22
2.3.5 Možnosti pořízení.....	22
2.3.6 Měřitelnost úspěšného nasazení .....	24
3 Praktická část – návrh a implementace aplikace.....	26
3.1 Nástroje použité při vývoji.....	27
3.1.1 Enterprise Architect 7.....	27
3.1.2 Toad Data Modeler 3.....	27
3.1.3 Microsoft SQL Server 2005 express.....	28
3.1.4 Microsoft Visual Studio 2008 .....	28

3.2 Návrh a implementace .....	30
3.2.1 Stanovení požadavků.....	30
3.2.2 Use Case diagramy .....	33
3.2.3 Datový model.....	36
3.2.4 Model návrhových tříd.....	38
3.2.5 Postup implementace.....	41
3.2.6 Řešené problémy při implementaci .....	42
3.3 Zprovoznění aplikace.....	45
3.3.1 Administrace CRM systému .....	46
3.3.2 Návod k obsluze systému .....	47
3.3.3 Zaškolení uživatelů .....	49
3.3.4 Migrace dat.....	50
4 Závěr .....	51

## SEZNAM OBRÁZKŮ

Obrázek 1: Části CRM a jejich funkce .....	17
Obrázek 2: Funkční požadavky obchodních případů .....	31
Obrázek 3: Funkční požadavky správy uživatelů .....	32
Obrázek 4: Seznam požadavků celého systému .....	33
Obrázek 5: Use Case model CRM systému .....	34
Obrázek 6: Příklad užití pro správu zákazníků .....	35
Obrázek 7: Příklad užití pro správu obchodních případů.....	35
Obrázek 8: Datový model aplikace vytvořený v Toad Data Modeleru .....	36
Obrázek 9: Vygenerovaný kód k vytvoření tabulky Produkty.....	37
Obrázek 10: Vygenerovaný kód k vytvoření primárního klíče .....	38
Obrázek 11: Kód k naplnění tabulky Produkty vzorovými daty .....	38
Obrázek 12: Package diagram návrhových tříd .....	39
Obrázek 13: Implementované třídy pro správu produktů .....	40
Obrázek 14: Hlavní formulářové okno programu .....	41
Obrázek 15: Ošetření výjimky při práci s databází.....	42
Obrázek 16: Funkce vracející stejný otisk hesla jako funkce md5 v PHP .....	43
Obrázek 17: Funkce k ověření správnosti emailové adresy .....	44
Obrázek 18: Kód použitý k vykreslení kruhového grafu.....	45
Obrázek 19: Dialogové okno pro přidávání (editaci) uživatele .....	47
Obrázek 20: Dialogové okno pro přidávání (editaci) zákazníka .....	47
Obrázek 21: Okno s kruhovým grafem reportingové funkce .....	48
Obrázek 22: Okno s kruhovým grafem marketingové funkce.....	49



# 1 Úvod

Tématem diplomové práce je implementace informačního systému typu CRM pro menší nebo středí firmu. Práce je rozdělena do dvou logických celků, teoretické a praktické části.

První část (teoretická) se zabývá obecným popisem informačních systémů, dále pak detailněji popisem CRM systému (systém pro řízení vztahů se zákazníky), jeho charakteristikami a přínosy pro podnik. V této kapitole jsou popsány možnosti, jak vybrat do firmy nový CRM systém, aby splňoval kladené požadavky a bylo možno ho integrovat do stávajícího prostředí, a jakým způsobem ho pořídit. Z této části vyplynuly základní obecné požadavky na navrhovaný CRM systém.

Druhá, praktická část práce obsahuje návrh a tvorbu obecného základu jednoduchého CRM systému, který bude možné po zpřesnění požadavků upravit podle přání zákazníka. V této části je popsán vývoj systému v jednotlivých fázích projektu – sběr požadavků, analýza, volba technologií, datový návrh, návrh tříd, implementace a testování. Na konci práce je zobrazena výsledná aplikace s návodem na zprovoznění, administraci a používání.

## 2 Teoretická část

Tato kapitola se zabývá informačními systémy, vymezením tohoto pojmu a obecně přínosy, které může mít nasazení informačního systému v konkrétní firmě. Dále se kapitola věnuje podrobněji CRM systémům a jejich integraci s jinými podnikovými systémy nebo aplikacemi.

### 2.1 Informační systémy (IS)

Pojem **Informační systém** není možné přesně definovat. Neexistuje žádná do-  
slovná definice, protože všichni tvůrci nepoužívají stejnou terminologii a každá  
firma má jiné požadavky. Existují však obecně platné popisy, které se pokouší  
tento termín objasnit. Dobrou definici napsal Vladimír Šmíd:

*„Informační systém (IS) lze chápat jako systém vzájemně propojených informací a pro-  
cesů, které s těmito informacemi pracují. Přičemž pod pojmem procesy rozumíme funk-  
ce, které zpracovávají informace do systému vstupující a transformují je na informace  
ze systému vystupující.“ [4]*

Informační systém je tedy softwarové vybavení, díky kterému lze snadněji řídit  
firemní procesy a díky kterému může management podniku lépe plánovat  
a rozhodovat.

Základními funkcemi pro práci s informacemi jsou:

- sběr informací,
- zpracování informací,
- ukládání a uchování dat,
- analýza dat,
- přenos dat,
- přehledné zobrazení relevantních informací. [4]

Informační systém pomáhá organizovat a přehledně shromažďovat podniková data. Tudíž i když jeden zaměstnanec prochází data a záznamy jiného zaměstnance, nečiní mu problémy se vyznat v organizaci dat – systém vše zobrazuje jednotně a přehledně.

Základní vlastností informačního systému je přínos v oblasti **bezpečnosti**. Pokud jsou zákazníkem kladeny vysoké nároky na bezpečnost dat, lze do informačního systému implementovat historii změn, kde vedoucí pracovníci zjistí, kdo si údaje zobrazil, kdy je změnil, jaká byla původní a nově zadaná hodnota. Dalším opatřením může být zamezení zkopírování celé databáze zákazníků. To je vhodné zejména při dnešní velké migraci zaměstnanců, kde se nezdá stává, že si zaměstnanec při svém odchodu s sebou odnese i citlivá firemní data, jakými může být seznam nejlepších zákazníků, firemní strategii, vývoj a přípravy novinek a další vnitropodnikové informace.

Nasazením informačního systému získá firma výhodu v úspoře času zaměstnanců. Lze totiž v reálném čase generovat finanční, analytické a další výkazy přímo ze systému a bez čekání.

Velkou výhodou je, že pomocí informačního systému lze snadno udržet uchovávaná data **aktuální** a **konzistentní**, a tím pro firmu relevantní. Dojde-li totiž k nějaké změně v datech (např. změna kontaktní osoby u zákazníka), stačí, aby ji jeden ze zaměstnanců zanesl do systému, a údaje se aktualizují na všech místech automaticky.

Informační systémy lze rozdělit podle funkce:

- CRM – řízení vztahů se zákazníky,
- EMC – řízení podnikového obsahu,
- ERP – plánování podnikových zdrojů,
- Business Intelligence. [1]

## 2.2 CRM, řízení vztahů se zákazníkem

CRM (Customer relationship management, česky řízení vztahů se zákazníky) je souhrn softwarových technologií, podnikových procesů a zdrojů určených pro řízení a zajišťování vztahů se zákazníky v oblasti obchodních činností, které jsou primárním zdrojem firemních zisků. [2]

Obvykle je toto řešení postaveno nad databázovým serverem a umožňuje shromažďování, zpracování a využití informací o zákaznících. Tyto informace pak mohou být analyzovány a z nich získávány poznatky a předpoklady k lepší spolupráci a prohlubování obchodních vztahů.

CRM systém by měl být firmě nápomocný v oblastech:

- aktualizace zákaznických potřeb pro udržení současných zákazníků,
- oslovení nových zákazníků,
- využívání znalostí, zkušeností a zpětné vazby od zákazníků při inovaci nabízených produktů,
- identifikace klíčových procesů a významných zákazníků,
- integrace marketingu, prodeje a zákaznické podpory,
- maximalizace zisku při udržení rovnováhy mezi skladováním, prodejem a marketingem. [2]

*„Řízení vztahů se zákazníky je strategie, která se orientuje na vybudování a podporu dlouhotrvajících vztahů se zákazníky. Není to tedy jen technologie, ale změna filosofie společnosti tak, aby důraz byl kladen na zákazníka. Na nedodržování této strategie havaruje většina implementací CRM.“ [5]*

Po většinu 20. století byla orientace marketingu směřována na „4P“:

- product (výrobek),
- price (cena),

- place (umístění),
- promotion (propagace).

V době posledního desetiletí, kdy se do prodeje a marketingu stále více dostávají informační technologie, se urychluje zpracovávání a tvorba analýz trhu. To vede k tomu, že se obchodníci a výrobci začínají orientovat na trh a na to, co si žádá zákazník. Moderním termínem marketingu dnešní doby je tedy „4C“:

- customer total cost (celkové náklady zákazníka),
- customer value (užitková hodnota),
- convenience (komfort),
- communication (komunikace). [5]

Při implementaci a nasazování CRM ale jistě nejde jen o zákaznické potřeby nebo o zvyšování zisku. Cílem by mělo být vybudovat rovnocenné partnerství, založené na loajalitě zákazníka a vedoucí k oboustranné spokojenosti. To ale nemůže zajistit žádná sebelepší aplikace.

Vztah se zákazníky podle moderního pojetí je založen na 4 základních pilířích:

- lidé (zaměstnanci),
- procesy (marketing, prodej, služby),
- technologie (umožňují oslovit velké množství klientu současně),
- data (především analýza uchovávaných dat a porozumění zákaznickým požadavkům vede k plnohodnotnému CRM). [2]

Proto je nutné, aby si firmy uvědomily, že nelze jen nasadit informační systém, ale měly by se snažit i o změnu myšlení všech zaměstnanců. Zásadní změnou je přechod od **produktového vnímání** marketingu k **zákaznickému pojetí** – to znamená, že důležitější jsou potřeby zákazníka, než samotný připravovaný produkt.

### 2.2.1 Funkcionalita CRM

Hlavním úkolem CRM systémů je podpora obchodních procesů v již existujících obchodních vztazích. Zejména má přinášet prohloubení vztahů s obchodními partnery, nabízet nové příležitosti a zvyšovat loajalitu zákazníků.

Mezi hlavní funkce CRM patří:

- evidence zákazníků, jejich obchodních a komunikačních kontaktů,
- sledování zákaznických požadavků a objednávek,
- vytváření nových obchodních příležitostí na základě již uskutečněných kontraktů nebo dle analýz chování zákazníků,
- generování reportů ze zákaznických požadavků,
- příprava a řízení marketingových kampaní. [1]

Výše uvedený výčet funkcí není konečný, nýbrž obsahuje pouze ty základní, které by měl obsahovat každý CRM systém. Další funkce se mohou odlišovat podle individuálních potřeb každého podniku.

### 2.2.2 Integrace CRM s ERP

**ERP (Enterprise Resource Planning**, do češtiny volně přeloženo jako **plánování podnikových zdrojů**) je jeden z nejrozsáhlejších a nejvíce využívaných typů informačních systémů. Integruje soubor procesů souvisejících s produkčními činnostmi podniku. Zejména se jedná o výrobu, logistiku, distribuci, správu majetku, prodej, fakturaci, a účetnictví. [6]

Některé dodávané ERP systémy obsahují CRM jako svou součást. To je ale méně častá varianta a navíc jeden takto robustní systém neobsahuje všechny požadované prvky. Proto se mnohem častěji volí 2 oddělené systémy, které mohou pracovat nad jednou společnou databází obsahující informace o zákaznících, kontaktech a dalších údaje. Informace o uskutečněných obchodech jsou pak uloženy v části CRM a například data o předmětu obchodního vztahu (produktu) jsou získávána z části ERP.

### 2.2.3 Integrace CRM s jinými aplikacemi

Samotný CRM systém je jako samostatná jednotka plně funkční. Nicméně pro větší pohodlí a usnadnění práce si zákazník vybírá řešení, které umožňuje integraci s dalšími aplikacemi, a tím dostává přidanou hodnotu k CRM a úsporu svého času.

Nejběžnější spolupráce s CRM je v rámci aplikací z kancelářského softwaru. Ať už jde o balík Microsoft Office nebo OpenOffice, měla by být zaručena kompatibilita textového editoru, tabulkového procesoru a hlavně emailového klienta. CRM systémy tak nabízí integraci například s klientem Microsoft Outlook, který lze snadno využívat k rozesílání hromadné korespondence nebo obchodních nabídek zákazníkům z informačního systému. [7]

Jiným příkladem integrace může být služba často označovaná jako „HelpDesk“. Jedná se o integraci do intranetového portálu (pro zaměstnance) nebo do webové prezentace (část pro zákazníky), která slouží jako podpora zákazníkům.

Požadovanou, avšak často placenou, formou integrace je spolupráce s webovými službami na Internetu. Tímto způsobem lze ověřit validitu zadané adresy nebo zjistit správnost telefonního či faxového čísla. *„Přínosem pro společnost využívající integraci s webovou službou nejsou jen dodatečné informace, ale i snížené náklady na získání těchto informací, neboť cena těchto informací se dostává i pod cenu časových nákladů, které by byly vynaloženy pouhým přepisováním známých údajů.“* [7]

Trendem poslední doby je spolupráce CRM s mobilními technologiemi. Díky tomuto může mít obchodní zástupce na cestách přístup do pošty, kalendáře, vizitek, informací z terénu od ostatních zaměstnanců, objednávek, nabídek a dalších dat prakticky kdykoli a odkudkoli (pokud to firemní bezpečnostní politiky dovolí).

#### 2.2.4 Základní vrstvy (části) CRM

**Operativní** (někdy označováno **operační**) CRM se zabývá podporou podnikových procesů, kterými jsou prodej, marketing a služby. Umožňuje a zaznamenává veškerou komunikaci mezi firmou a zákazníkem. Hlavním přínosem je možnost sledovat historii komunikace a obchodních procesů mezi subjekty, které jsou v rámci CRM evidovány.

Operativní CRM lze využít k:

- tvorbě a sledování marketingových kampaní,
- automatizaci prodejního procesu. [5]

Do této části patří také aplikace podporující obchodní činnost zaměstnanců (jako jsou řízení příležitostí a nabídek) nebo se starající o objednávky, servis, zákaznické požadavky (podpora a služby zákazníkům). [1]

Druhou vrstvou je **analytické** CRM, jehož úkolem je analyzovat zákaznická data a předpovídat chování zákazníků. Tato část se zaměřuje na poznání zákazníka, jeho preferencí, požadavků, hodnot a rizikovosti. Dochází k analýze marketingových kampaní a poznávání zákazníka s cílem:

- optimalizovat efektivitu marketingových kampaní,
- vyhodnocovat marketingové kampaně,
- hledat nové prodejní kanály,
- udržet stávající zákazníky,
- analyzovat zákaznickou rentabilitu (rozhodovací funkce),
- analyzovat zákazníka, požadavky, tvorba cen. [5]

Na obrázku číslo 1 jsou zobrazeny jednotlivé funkce operativní a analytické části CRM, jejich vzájemné interakce a vazby.





Obrázek 1: Části CRM a jejich funkce (zdroj: Wikipedia [5])

Posledním typem je **kooperační** (neboli **kolaborativní**) CRM, které má za úkol rozšiřovat možnosti komunikace se zákazníky pomocí různorodých kanálů.

*„Operativní CRM nabízí užitečné informace, které vznikají při interakci se zákazníkem, jednotlivým obchodním oddělením, jako je prodej, technická podpora a marketing. Jedná se například o poskytnutí informací o specifických zákaznických požadavcích či dotazů na nové služby z technické podpory prodeje marketingu. Cílem je sdílení těchto informací získaných ze všech oddělení pro zvýšení kvality poskytovaných služeb zákazníkům.“ [5]*

Typickým příkladem operativního CRM jsou kontaktní centra, která pomocí elektronické pošty, hlasové komunikace, webu a dalších komunikačních kanálů zrychlují a zkvalitňují přístup k informacím a servisním požadavkům. [1]

### 2.2.5 Mýty o CRM

V praxi se lze velmi často setkat se zkreslenými představami o CRM. V této kapitole jsou popsány 3 nejčastější omyly v chápání CRM systémů.

**CRM je novota.** *„Není to pravda. Koncepce řízení vztahů se zákazníky se začaly rozvíjet přibližně před dvaceti lety. Bouřlivější rozvoj začal s vytvořením podpůrných nástrojů využívajících možnosti výpočetní a později i komunikační techniky.“* [2]

**CRM je software.** To také není pravda. Ačkoli názor, že CRM musí být vždy nějaká aplikace nebo softwarový balík, je v praxi velmi častý, není pravdivý. V dnešní době, kdy je naprostá většina firem vybavena informačními technologiemi a využívá ke své obchodní činnosti specializovaný software, je CRM reprezentováno aplikací, která umožňuje řízení vztahů se zákazníky. To ale ještě neznamená, že to tak musí být vždy. CRM je pouze soubor marketingových technik, které pomáhají firmě řídit aktivity spojené s péčí o zákazníky. Je-li tedy řeč o CRM, nemusí se jednat pouze o programové vybavení, ale může jít o marketingová pravidla a techniky. Informační systém je jen jednou možnou variantou nástroje pro tyto podnikové aktivity. [2]

**CRM je složité a jeho implementace trvá dlouho.** U tohoto mýtu je nutno podotknout, že se může jednat o velmi subjektivní dojem a závisí na spoustě faktorů.

*„Často se lze v praxi setkat s pohledem ovlivněným ERP systémy a složitostí jejich implementace. Mnoho lidí si vytvořilo o CRM představu, která kopíruje právě ERP systémy. A to rozhodně není správně.“* [2]

Výsledná cena a délka implementace je také závislá na způsobu dodání softwaru (popsáno v kapitole „Možnosti pořízení CRM“) a na zákaznických požadavcích. Obecně lze ale říci, že i po velmi krátké době lze sledovat vliv přínosu informačního systému a pomocí systémových statistik lze jednoduše hodnotit (ne)úspěch implementované koncepce řízení vztahů se zákazníky.

## **2.3 Zavedení CRM systému**

Důvodem, který přivádí vedení firmy k myšlence pořídit informační systém pro lepší správu svých zákazníků, bývá velmi často nespokojenost s používáním stávající neorganizovanou, decentralizovanou agendou, kde si každý zaměstnanec (nebo oddělení) ukládá a spravuje své informace sám. K těmto účelům bývá velmi často použito řešení pomocí tabulek v programu Microsoft Excel nebo dokonce třeba jen pomocí kontaktů a poznámek v adresáři. Takto vedené evidence zákazníků, smluv a obchodních příležitostí jsou neefektivní, zdlouhavé a neposkytují žádné rozumné způsoby vyhledávání a třídění informací.

Dalším problémem bývá interní firemní komunikace. Velké množství firem uskutečňuje komunikaci pouze pomocí e-mailů, instantních komunikátorů a dalšími běžnými prostředky. Koordinace zaměstnanců tak bývá velmi obtížná, z toho vyplývá neoptimální workflow a pomalá komunikace. Dochází k duplicitě, ztrátě a neaktuálnosti interních dokumentů a to má negativní následky pro chod celé firmy. [1]

Proto ve snaze stát se moderní firmou, která může pružně reagovat na stav trhu a pobídky, často dochází k nasazení některého z informačních systémů, který přináší určitou tržní výhodu (v některých případech již nutnost).

### **2.3.1 Stanovení firemních požadavků**

Plánuje-li firma investovat své finanční prostředky do nasazení informačního systému, kterým může být CRM (nebo obecně jakýkoli informační systém), musí na začátku umět jasně stanovit své požadavky, které musí systém pro jejich business přinášet.

Základními požadavky, které jsou klíčové pro implementaci CRM:

- komplexnost (funkce, které zákazník požaduje – evidence zákazníku, vazby příležitostí a smluv k zákazníkům, správu kontaktů, nástroje pro komunikaci a další funkce),
- možnost rozšíření systému (přidání funkcionalit, vazba na jiný systém, přizpůsobení nastavení),
- snadná integrace do stávající ICT infrastruktury,
- spolehlivost systému,
- minimální nároky na správu (firma chce systém používat a věnovat se činnostem, které jsou hlavní podnikatelským záměrem firmy, a ne investovat do správy systému),
- bezpečnost (autentizace, autorizace, integrita dat).

### **2.3.2 Kritické faktory úspěchu**

Samotná implementace informačního systému firmě ale nezaručí automatický úspěch a zvýšení zisků. Proto je ještě před samotným nasazením CRM dobré určit, co od systému očekává, jaký bude jeho přínos pro obchodní aktivity a které faktory mohou ovlivnit (příznivě i negativně) obchodní záměry.

Klíčovými faktory úspěchu firmy tedy jsou:

- dobré odhady náročnosti (časové a finanční) zavedení systému,
- prosazování používání informačního systému vedením firmy,
- zrychlení podnikových procesů,
- jednotná koncepce při zavádění systému do provozu,
- dostatečná příprava uživatelů – školení, předvedení, testování,
- vyžadování dodržování firemní kultury,
- dobré nastavení podnikové strategie,
- dodržování stanovené koncepce při zavádění CRM do provozu,

- testování, ověření funkčnosti, testy bezpečnosti systému,
- ověření znalostí uživatelů o systému a schopností ho snadno a efektivně využívat.

### 2.3.3 Přínosy nasazení CRM v podniku

Pokud se již firma rozhodla nasadit řešení pro řízení vztahů se zákazníky, měla by znát možnosti nového systému a přínosy, které CRM s sebou přináší, a čím se tedy bude odlišovat od podniků, které takový systém nepoužívají.

Základní přínosy CRM systémů jsou:

- zkvalitnění a zrychlení obchodní činnosti firmy,
- optimalizace obchodních procesů,
- komunikace mezi marketingem, prodejem a službami,
- zlepšení možnosti spolupráce mezi obchodními týmy,
- nástroje na sledování historie obchodů,
- lepší evidence zákazníků a obchodních příležitostí,
- možnost propojení s jinými firemními agendami a aplikacemi,
- integrace reportingových nástrojů pro management,
- možnost předpovídat a odhadovat budoucí náklady a tržby,
- zvýšení konkurenceschopnosti, odlišení se od ostatních podniků ve stejném odvětví,
- přístup k datům v reálném čase, využití mobilních technologií,
- nárůst motivace zaměstnanců – mohou využít kvalitního a intuitivního nástroje, který ušetří jejich čas,
- vyšší efektivita zaměstnanců – odstranění nebo automatizace některých úkonů. [2]

### 2.3.4 Volba optimálního systému

Ještě před tím, než dojde na rozhodnutí ohledně volby technického způsobu pořízení CRM systému, je vhodné promyslet různé faktory, které ovlivní výslednou cenu systému, dobu, za kdy bude systém nasazen, a také možnost rozšiřování a přizpůsobení. Při rozhodování je třeba vzít v úvahu:

- Jaké jsou dosavadní firemní zkušenosti s řízením vztahů se zákazníky a příslušnými informačními systémy?
- O jak velkou organizaci jde?
- V jakém řádu se pohybuje počet potenciálních zákazníků (cílová skupina pro řízení vztahů se zákazníky)?
- Jaké množství finančních prostředků je k dispozici?
- Jaké jsou představy a požadavky návratnosti investic do informačního systému (časový horizont)?
- Existuje specializovaný CRM informační systém pro firemní odvětví?
- Jaký typ systému a jak úspěšně využívají firmy obdobného charakteru (velikostně, oslovující podobnou cílovou skupinu zákazníků)?
- Které firemní činnosti ve vztahu k zákazníkům by měly být automatizovány (prodej, marketing, servis)?
- Na jaké úrovni by měla automatizace probíhat (operativní, analytická, kooperační)?
- S jakými firemními informačními systémy bude potřeba nový CRM systém provázat? [2]

Nemusí být jednoduché najít jednoznačné odpovědi na tyto otázky, ale zamýšlení nad nimi může být nápomocno při hledání způsobu pořízení systému.

### 2.3.5 Možnosti pořízení

Rozhodne-li se podnik nasadit do firmy CRM systém, je postaven před zásadní otázkou – který je ten správný a jakým způsobem ho pořídit. Na trhu je relativně velké množství hotových řešení a existují i jiné způsoby, jak získat software

na míru přesně podle požadavků. Rozdíl ale může být v mnoha aspektech (například cena, délka implementace, možnost rozšíření a dalších).

Způsoby pořízení CRM aplikace:

- Nejlevnějším způsobem pořízení je využití **open source** řešení, kde lze zdarma stáhnout již nějaké hotové řešení a nasadit ho prakticky ihned jako fungující a ověřené prostředí. Při dodržení licenčních podmínek lze otevřený zdrojový kód využívat, upravovat a přizpůsobovat vlastním požadavkům. To může být ale i velkou nevýhodou, když zdrojový kód produkční aplikace je volně dosažitelný pro hackery. Nejznámějšími open source CRM jsou SugarCRM, SplendidCRM, Hypergate, Vtiger CRM, Cream CRM. [9]
- Další variantou, jak v krátké době nasadit CRM může být zakoupení **hotového uceleného robustního řešení**, které obsahuje všechny potřebné funkcionality. Tato forma je časově nenáročná, avšak představuje jisté počáteční náklady na zakoupení softwaru a potřebných licencí. Nevýhodou může být nemožnost přizpůsobení produktu a chybějící rozhraní pro komunikaci s jinými firemními aplikacemi.
- Tento problém odstraňuje pořízení **hotového řešení se základní funkcionalitou** a jeho následné doprogramování podle požadavků. Tím firma získá aplikaci přesně pro své potřeby s možností spolupráce i s ostatními firemními systémy (například ERP).
- Pokud jsou požadavky natolik odlišné od klasických hotových řešení, musí firma přikročit k vytvoření **úplně celého systému od počátku na zakázku**. Podnik pak má k dispozici CRM s funkcemi přímo na míru pro jejich obchodní činnost, ale musí počítat s vyšší pořizovací cenou a určitou dobou na navržení, implementaci, otestování a nasazení výsledné aplikace. [1]

- V případě, že má firma k dispozici v rámci svých zaměstnanců programátory a jejich znalost problematiky je dostatečná, může přikročit k **vytvoření aplikace interními programátory**. Tento způsob s sebou nenese žádné náklady na vytvoření softwaru, ale podle rozsáhlosti požadavků na aplikaci bude jistou dobu trvat vytvoření a nasazení aplikace, a po tuto dobu nebudou zainteresovaní programátoři schopni plně pracovat na svých obvyklých činnostech v zaměstnání.
- V poslední době hodně využívanou technologií a způsobem nákupu softwaru je **obchodní model SaaS** (Software as a Service), kdy dochází k outsourcingu celého řešení dodávající firmou. Zákazníkovi odpadá starost o pořízení (případně naprogramování) aplikace, její správy a celého provozu. O to vše se stará poskytovatel služby a on nese zodpovědnost na bezchybný provoz. Výhodou jsou zpravidla nižší pořizovací náklady. Výše pravidelných plateb se odvíjí pouze od počtu aktuálně využívaných licencí a tato suma je předem známá, tudíž ji lze zahrnout do finančních plánů a není důvod se obávat nečekaných výdajů. Nevýhodou tohoto způsobu je dlouhodobá závislost na dodavateli služby a také v oblasti zabezpečení. Data jsou uložena u poskytovatele a firmy pak mají pocit, že je nemají přímo pod dohledem, a obávají se o zcizení vnitropodnikových informací. Přitom nemusí dojít k selhání na straně poskytovatele, ale může dojít i k prolomení komunikace cestou přes Internet. Z tohoto důvodu je dnes již spíše nutností než doporučením využívat zabezpečených komunikačních protokolů. [10]

### 2.3.6 Měřitelnost úspěšného nasazení

Hlavní měřítkem úspěšnosti firmy by měl být zisk. Ten je hlavním důvodem obchodní činnosti a nejdůležitějším měřítkem úspěchu firmy.



Mezi další faktory úspěšného nasazení lze zařadit tyto:

- snížení provozních nákladů firmy,
- zrychlení jednotlivých činností obchodního oddělení,
- zefektivnění interní firemní komunikace,
- odhalení neefektivních procesů a jejich odstranění
- vylepšení image firmy,
- udržení stávající zákazníky,
- nárůst nových poptávek a podepsaných smluv,
- snížení času na průběh celé obchodní transakce (od nabídky až po uzavření smlouvy),
- snížení počtu úkonů zaměstnanců na administrativu. [1]

### 3 Praktická část – návrh a implementace aplikace

Obsahem této kapitoly je popis jednotlivých fází návrhu, analýzy a implementace základu vzorové aplikace CRM. Na projektu spolupracuji s kolegou a studentem Bc. Jirkou Limberským, jehož úkolem je vytvořit webovou aplikaci, a mým úkolem je naprogramovat vnitropodnikovou intranetovou aplikaci v jazyce C#.

Během spolupráce došlo k vytvoření společného základu při činnostech:

- stanovení vize projektu – prvotní myšlenka budoucí podoby systému,
- určení plánu projektu – zamyšlení nad hlavními funkcionalitami,
- tvorbě datového modelu – navržení společného datového modelu pro ukládání dat CRM systému (vnitropodnikové aplikace i webové části),
- vzájemné konzultace nad řešenými problémy.

Výsledkem spolupráce by měly být 2 nezávislé systémy přistupující ke společné bázi dat a nabízející praktické shodné funkce. Vnitropodniková intranetová aplikace je určena pro používání zaměstnanci přímo z podniku v rámci lokální zabezpečené sítě a webová aplikace umožňuje přístup kdykoli a odkudkoli přes Internet.

Vývoj systému probíhal v následujících krocích:

- sběr požadavků (spolupráce s vedoucím práce),
- výběr technologie a nástrojů k tvorbě,
- analýza požadavků,
- tvorba datového modelu a datových tříd,
- implementace,
- testování.

### 3.1 Nástroje použité při vývoji

K tvorbě systému bylo využito aplikací, které jako studenti Univerzity Pardubice máme k dispozici a používali jsme je během našeho vzdělávání.

#### 3.1.1 Enterprise Architect 7

*„Enterprise Architect (EA) od společnosti Sparx Systems je kompletní nástroj pro systémovou analýzu a návrh, který pokrývá celý životní cyklus vývoje systému, od zadání požadavků přes analýzu, návrh a testování, vše s využitím diagramů v UML.“ [11]*

Tohoto programu bylo využito k vytvoření popisu v UML (Unified Modeling Language, jazyk umožňující navrhovat a graficky zobrazovat business procesy, systémové funkce, modely a diagramy) ve fázích:

- sběr požadavků,
- tvorba Use Case diagramů,
- modelování datových tříd.

#### 3.1.2 Toad Data Modeler 3

Toad Data Modeler je nástroj, který umožňuje graficky vytvořit a spravovat návrh databáze (logický i fyzický návrh). Podporuje široké spektrum databázových serverů (Oracle, Microsoft SQL, MySQL, PostgreSQL, Microsoft Access a další). Nabízí mnoho funkcí – vytvoření datových entit a vazeb mezi entitami, podporuje dědičnost, stanovení integritních omezení a generování SQL skriptu pro vytvoření databáze podle návrhu. [12]

Nástroje bylo využito ve fázi tvorby datového modelu a byl nápomocný při:

- vytvoření databázových tabulek,
- specifikaci datových typů,
- nastavení primárních klíčů k identifikaci řádků,
- vytváření integritních omezení (cizí klíče),
- definování unikátních hodnot v tabulkách.

### 3.1.3 Microsoft SQL Server 2005 express

Microsoft SQL Server je relační databázový a analytický systém pro produkční báze dat a datové sklady. Systém je založený na architektuře klient/server a je rozdělen na 2 části: serverovou (databázový stroj pro řízení SQL dotazů, stará se o čtení, ukládání a o práci s daty, udržuje data konzistentní) a klientskou část, která je určena pro databázové vstupy a výstupy, správu serveru a generování analýz a reportů. [8]

Pro snazší správu a ladění SQL dotazů bylo využito nástroje Microsoft SQL Server Management Studio Express, který lze zdarma stáhnout a bezplatně využívat na administraci SQL serveru Express. Management Studio bylo použito na úkony:

- nastavení bezpečného ověřování k SQL serveru,
- vytvoření databázových objektů pomocí vygenerovaného SQL skriptu z nástroje Toad Data Modeler,
- vytvoření účtu a nastavení práv pro přístup k objektům,
- spouštění a ladění SQL dotazů,
- zálohování a obnovování databáze z důvodu testování aplikace.

### 3.1.4 Microsoft Visual Studio 2008

Jedná se o vývojové prostředí (IDE) od Microsoftu, které lze použít pro tvorbu konzolových aplikací, grafických aplikací s Windows formuláři, webových stránek a webových služeb na platformách Microsoft Windows, Windows Mobile, .NET (z angličtiny „dotnet) Framework a Microsoft Silverlight.

Visual Studio nabízí přehledné prostředí pro rychlé psaní a generování zdrojového kódu a umožňuje debugování a krokování aplikace při testování. Podporovány jsou jazyky: C/C++, C#, J#, Visual Basic, XML, HTML, JavaScript. [13]

## Výhody Visual Studia a technologie .NET:

- Jazyková nezávislost – všechny podporované jazyky se kompilují do zprostředkujícího jazyka (Common Intermediate Language), díky čemuž mohou různé programy lépe spolupracovat.
- Objektově orientované programování - .NET Framework je plně založen na objektově orientovaných principech.
- Knihovna tříd Visual Studia obsahuje připravené třídy základních funkcí s dobře popsányými metodami a nápovědou.
- Integrovaná komponenta ADO.NET, která zpřístupňuje aplikaci možnost komunikovat s relačními databázemi a dalšími datovými zdroji.
- Sada tříd a metod podporující standard XML. To s sebou přináší větší nezávislost na platformě a umožňuje exportovat a importovat data z různých aplikací, či dokonce jiných operačních systémů.
- Zcela soběstačné sestavení výsledné aplikace. Technologie .NET umožňuje soukromé sestavení, které vytvoří spustitelný soubor, jež stačí umístit do souborového oddílu operačního systému, a není nutno načítat cokoli z dll knihoven či z registru Windows.
- Podpora vícevláknových procesů, které na moderních vícejádrových počítačích výrazně snižují dobu vykonání dané operace. Visual Studio obsahuje prvky pro snazší debugging a testování paralelně běžících procesů a navíc zobrazuje aktuální činnost prováděnou jiným vláknem aplikace, čímž pomáhá odhalit špatně laditelné chyby v paralelním programování.
- Vytváření aplikací pro 64bitové systémy (Visual Studio je dostupné pouze v 32bitové verzi, ale od verze Professional umožňuje vytvářet sestavení pro 64bitovou architekturu). [13]

## 3.2 Návrh a implementace

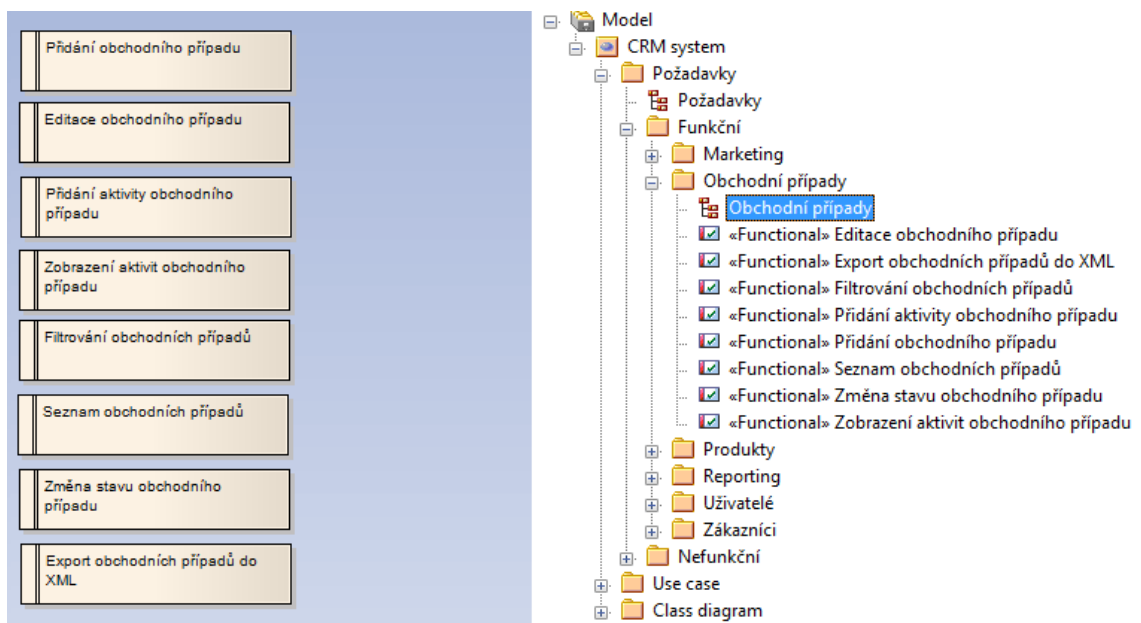
### 3.2.1 Stanovení požadavků

Základní charakteristikou navrhovaného systému je vytvoření aplikace podle síťové architektury klient/server, kde klient i server jsou striktně odděleny a jejich vzájemná komunikace probíhá prostřednictvím protokolů počítačové sítě. Na straně klienta probíhá zpracování dat (vstupy, výstupy), zobrazování informací a je zde umístěna většina aplikační logiky. Na serverové straně je uložena centrální báze dat společná pro vnitropodnikovou intranetovou i webovou aplikaci.

Funkční požadavky:

- Systém zabezpečený autorizací a autentizací uživatele.
- Uživatelé mají možnost měnit hesla.
- Systém s více uživatelskými rolemi (obchodník, manažer, administrátor), zaměstnanec může být členem pouze jedné.
- Ošetření vstupních hodnot ve formulářích.
- Nutnost výběru z výčtu hodnot u položek adresa (kraj, okres, obec) – ošetření proti zadávání neexistujících nebo neplatných kombinací.
- Administrátor může exportovat celou databázi – zálohování.
- Manažer může prohlížet a editovat obchodní případy ostatních.
- Systém poskytuje manažerům marketingové a reportingové funkce.
- Systém bude rozdělen do více logických celků – uživatelé (zaměstnanci), zákazníci (s kontaktními osobami), produkty, obchodní případy).
- Správa uživatelů – přidávání, editování, přiřazení role, filtrování, export, možnost nastavení hesla. Login musí být unikátní.
- Správa zákazníků – přidávání, editování, správa jeho kontaktních osob, filtrování, export, možnost odeslat mail zákazníkovi.
- Správa produktů – přidávání, editování, filtrování, export.

- Správa obchodních případů – přidávání, editování, správa aktivit ke každému obchodnímu případu, filtrování, export. Na každý obchodní případ se může vázat více produktů. Každý zaměstnanec si spravuje svoje obchodní případy. Manažer může spravovat obchodní případy i ostatních zaměstnanců.
- Základní funkce jsou přístupné všem uživatelům.
- Marketingové a reportingové funkce jsou přístupné jen manažerům.
- Správu produktů mají k dispozici manažeři a administrátor.
- Správa uživatelů je dostupná jen administrátorovi.

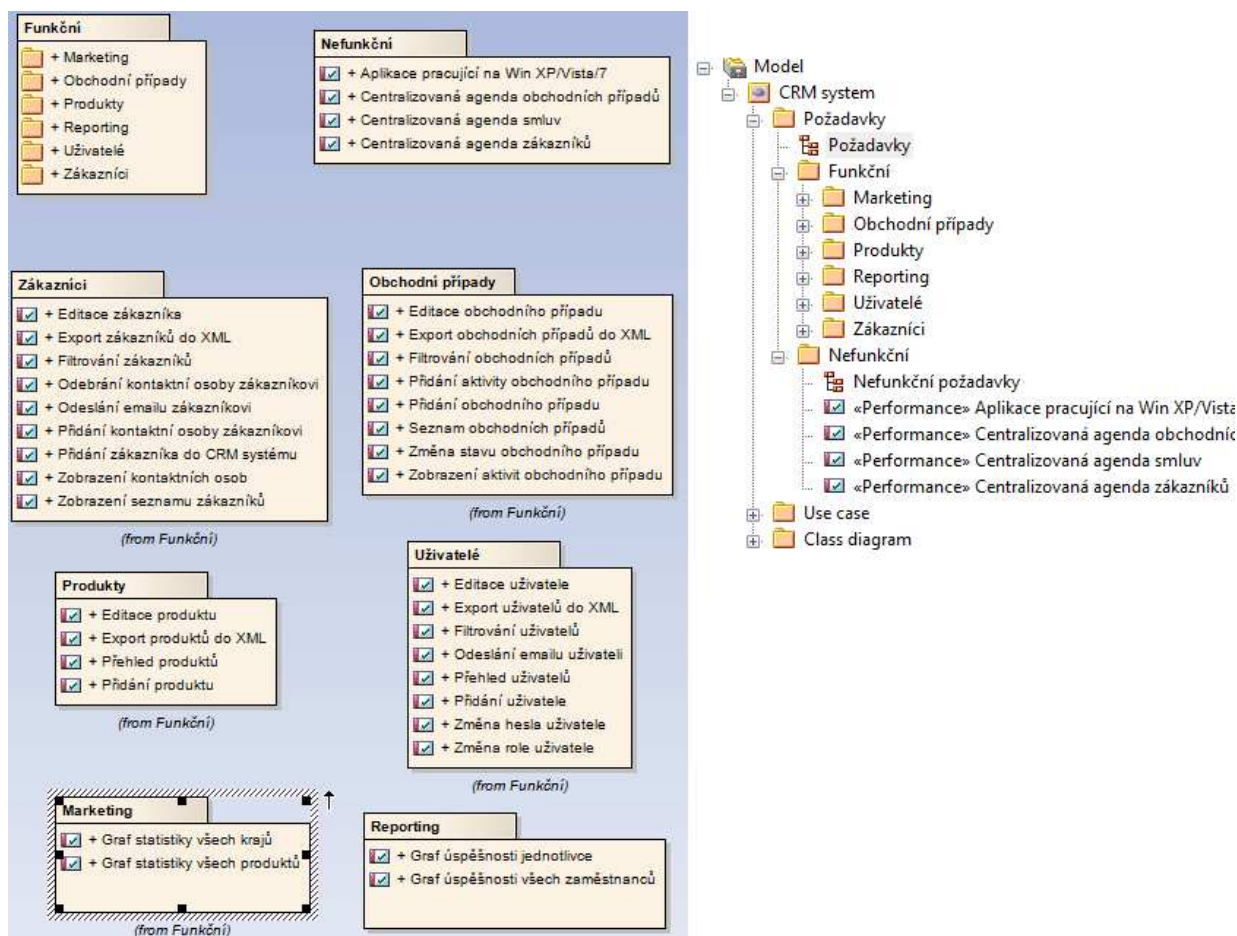


Obrázek 2: Funkční požadavky obchodních případů.

Na obrázku číslo 2 je zobrazen soupis funkčních požadavků pro obchodní případy zpracovaný v programu Enterprise Architect. Na obrázku číslo 3 jsou sepsány požadavky na správu uživatelů. Obdobným způsobem jsou v Enterprise Architectu zpracovány funkční požadavky i pro zbylé části systému.







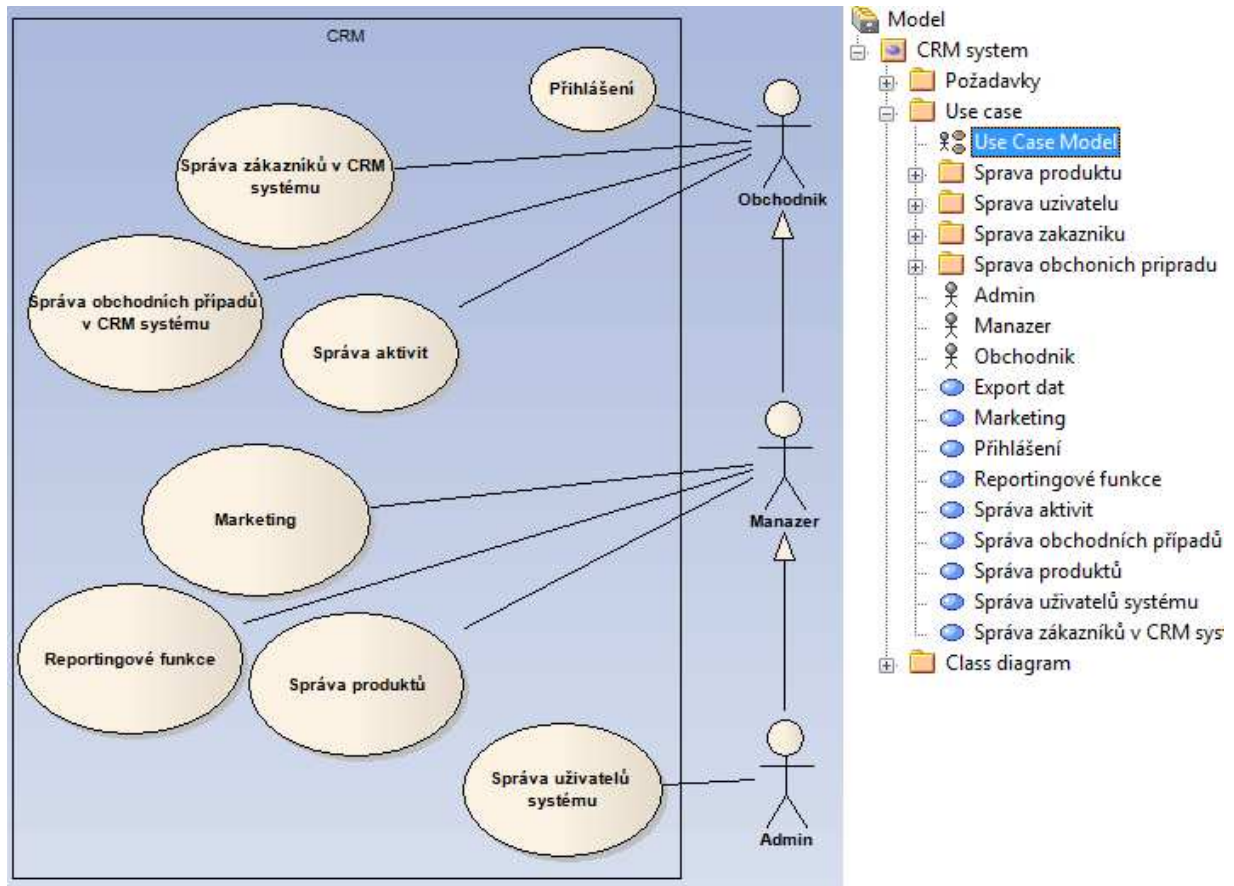
Obrázek 4: Seznam požadavků celého systému.

### 3.2.2 Use Case diagramy

Na základě požadavku, že systém má rozdílné chování podle uživatelských rolí, byli stanoveni 3 aktéři: *obchodník*, *manažer* a *administrátor*.

Obchodník má k dispozici pouze základní funkce pro práci se zákazníky (přivání, editaci, správu jejich kontaktních osob), obchodními případy (přidávání, editaci) a možnosti zadávání aktivit a změn stavů u obchodních případů. Manažer má k dispozici všechny funkce jako obchodník a navíc se mu zpřístupní reportingové funkce, marketing, správa produktů a možnost nahlížet a upravovat obchodní případy ostatních zaměstnanců. Administrátor pak přebírá všechny funkce manažera a k tomu navíc získává oprávnění ke správě systémových uživatelů a zálohování. Tuto problematiku mnohem lépe popisuje ob-

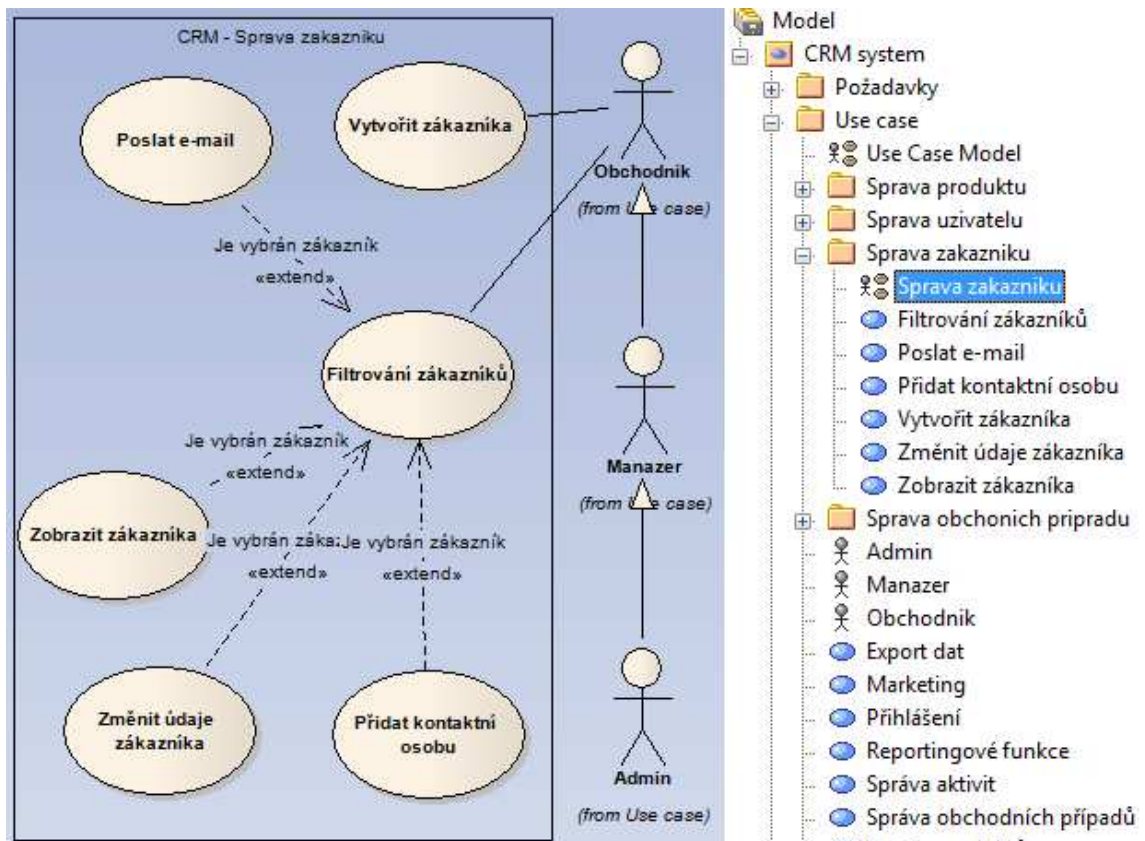
rázek číslo 5, kde jsou zobrazeni jednotliví aktéři a jim přidělená oprávnění k funkcím CRM systému.



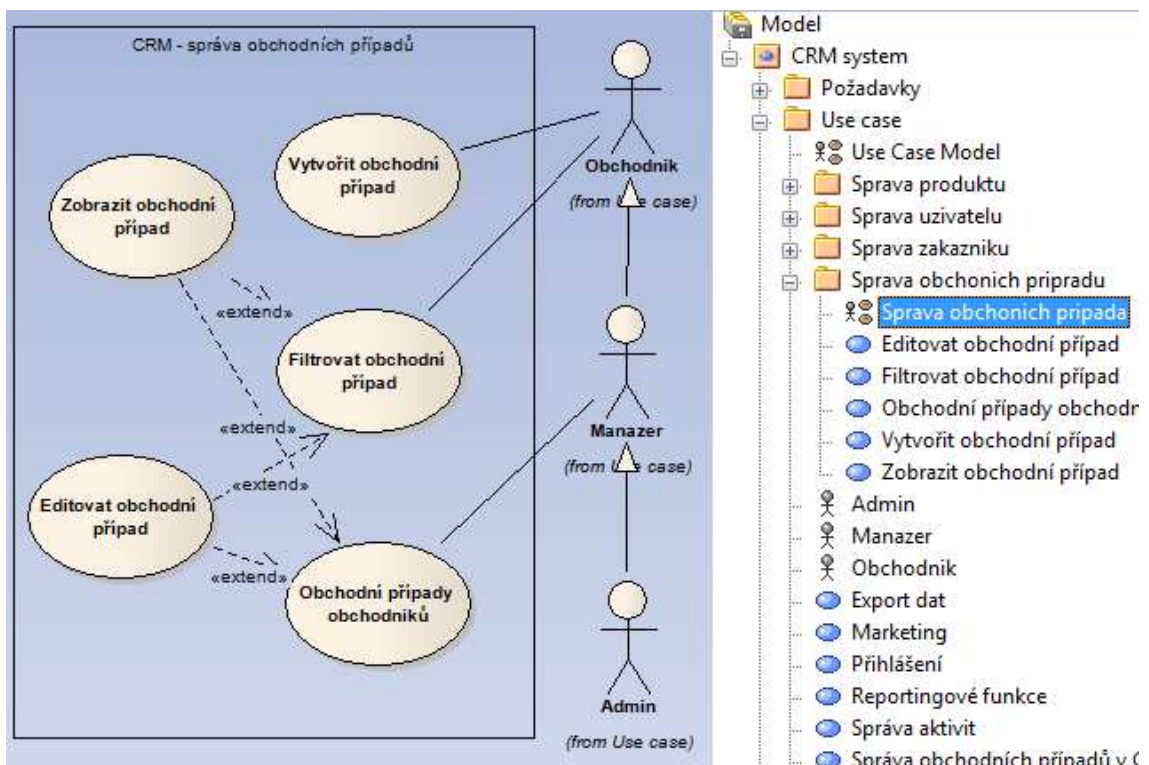
Obrázek 5: Use Case model CRM systému.

Pro lepší objasnění dané problematiky byly vytvořeny Use Case diagramy pro jednotlivé zde zobrazené základní funkce CRM systému.

Na obrázku číslo 6 lze přehledně vyčíst strukturu případu užití pro správu zákazníků a na obrázku číslo 7 strukturu pro správu obchodních případů. Zbylé Use Case diagramy jsou přiloženy v projektu vytvořeném v programu Enterprise Architect.



Obrázek 6: Příklad užití pro správu zakazníků.



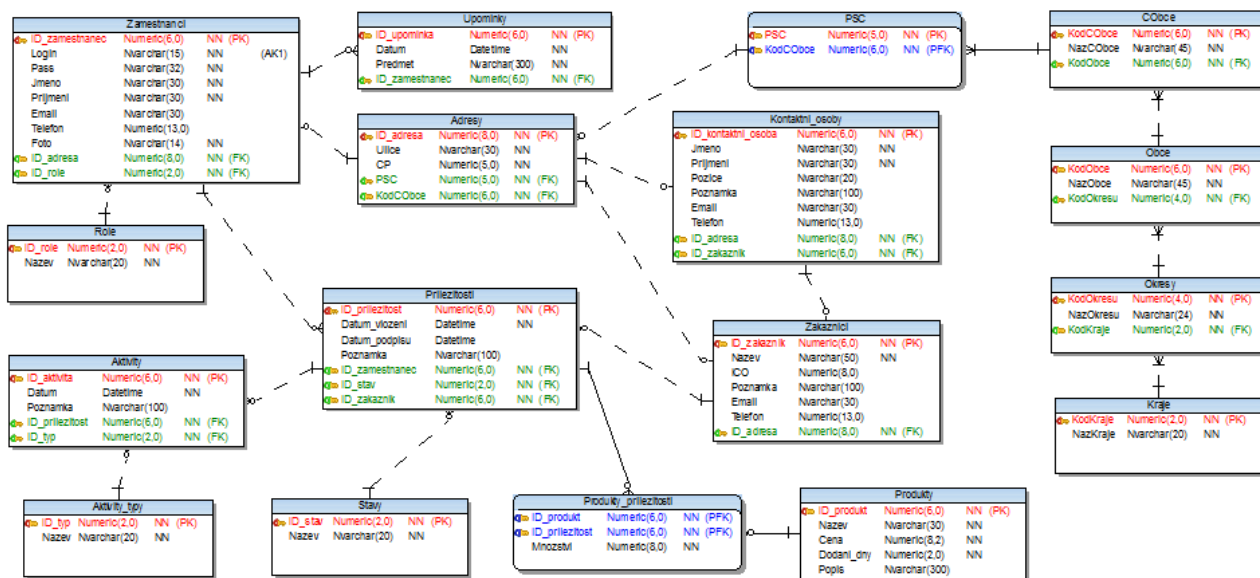
Obrázek 7: Příklad užití pro správu obchodních případů.

### 3.2.3 Datový model

Na základě stanovených požadavků a nutnosti uchovávat data a nastavení pro CRM systém byl navržen datový model zobrazený na obrázku číslo 8.

Vytvořené schéma obsahuje entity s jejich atributy, primární a cizí klíče, vazby mezi entitami, omezení na unikátnost hodnos. Model byl vytvořen ve 3. normální formě, tudíž splňuje podmínky:

- nedělitelnost atributů (atomické hodnoty ve sloupcích),
- jednoznačná identifikace každého řádku tabulky pomocí primárního klíče (všechny neklíčové atributy jsou jeho funkcí),
- žádný atribut tabulky není závislý na jiném atributu téže tabulky.



Obrázek 8: Datový model aplikace vytvořený v Toad Data Modeleru.

Základními kameny celého modelu jsou entity:

- *Zaměstnanci* – tabulka pro uchování dat o uživateli systému. Jsou evidovány hodnoty jméno, příjmení, přihlašovací jméno (login – musí být unikátní), heslo (ukládán je pouze md5 otisk hesla), kontaktní údaje a vazba na uživatelskou roli do tabulky rolí.

- *Zákazníci* – tabulka se všemi firemními zákazníky obsahuje atributy název firmy, sídlo firmy a volitelně IČO, poznámku k zákazníkovi a kontaktní údaje. Dále je ke každému zákazníkovi možno přidávat neomezený počet kontaktních osob.
- *Produkty* – tabulka s informacemi o nabízených (zprostředkovaných) produktech. Evidovány jsou povinně hodnoty název, cena, počet dní k dodání produktu a volitelně libovolný popis o maximální délce textového řetězce 300 znaků.
- *Příležitosti* – tabulka s jednotlivými obchodními případy, váže se na uživatele (zaměstnance) a na zákazníka, přičemž každý obchodní případ má pouze jednoho odpovědného zaměstnance a jednoho zákazníka. V rámci příležitosti lze evidovat více předmětů obchodní transakce (vazba M:N přes vazební tabulku na produkty), každý případ má jedinečný identifikátor, aktuální stav a seznam aktivit, které byly vykonány před uzavřením kontraktu.

Pomocí nástroje Toad Data Modeler byl z datového návrhu vygenerován skript pro vytvoření databázových objektů. Následující odstavec je část vygenerovaného kódu v jazyce SQL sloužící k vytvoření tabulky produktů.

```

CREATE TABLE Produkty
(
  ID_produkť Numeric(6,0) IDENTITY(1,1) NOT NULL,
  Nazev Nvarchar(30) NOT NULL,
  Cena Numeric(8,2) NOT NULL,
  Dodani_dny Numeric(2,0) NOT NULL,
  Popis Nvarchar(300) NULL
)

```

Obrázek 9: Vygenerovaný kód k vytvoření tabulky Produkty.

Na obrázku číslo 10 je kód vytvářející primární klíč nad sloupcem *ID\_protukt* tabulky *produkty*:

```
ALTER TABLE Produkty ADD CONSTRAINT PK_PRODUKTY PRIMARY
KEY (ID_produkť)
go
```

Obrázek 10: Vygenerovaný kód k vytvoření primárního klíče.

Naplnění tabulky *produkty* testovacími daty (SQL inserty):

```
INSERT INTO Produkty (ID_produkť, Nazeť, Cena, Dodani_dny, Popis)
VALUES (1, 'Produkt1', 100.00, 3, NULL);
INSERT INTO Produkty (ID_produkť, Nazeť, Cena, Dodani_dny, Popis)
VALUES (2, 'Produkt2', 200.00, 5, NULL);
INSERT INTO Produkty (ID_produkť, Nazeť, Cena, Dodani_dny, Popis)
VALUES (3, 'Produkt3', 300.00, 1, NULL);
INSERT INTO Produkty (ID_produkť, Nazeť, Cena, Dodani_dny, Popis)
VALUES (4, 'Produkt4', 400.00, 3, NULL);
```

Obrázek 11: Kód k naplnění tabulky *Produkty* vzorovými daty.

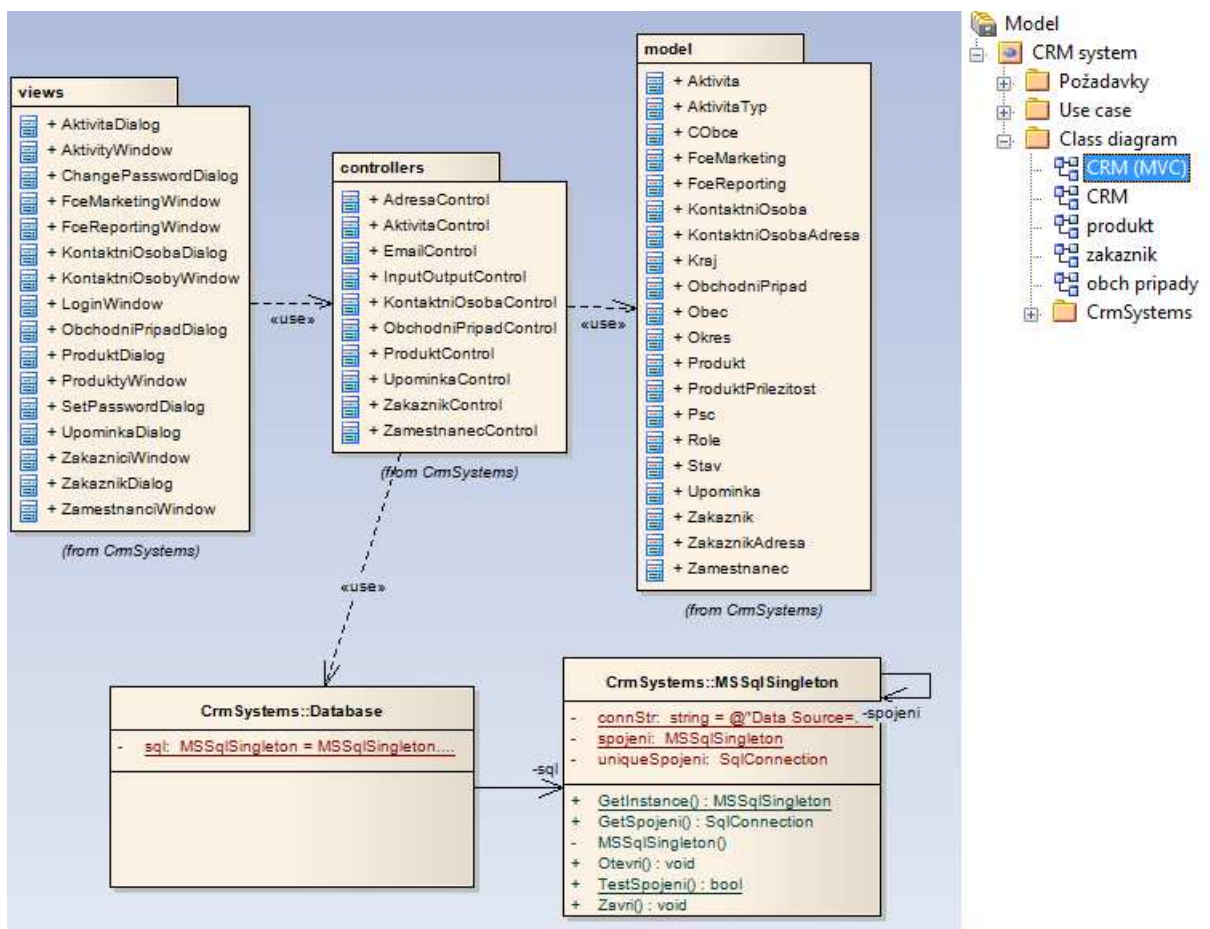
### 3.2.4 Model návrhových tříd

Model návrhových tříd by měl u každé třídy obsahovat kompletní sadu atributů, jejich názvy, datové typy (s nepovinně nastavenými výchozími hodnotami) a přístupová oprávnění. Dále by měl specifikovat výčet potřebných metod včetně parametrů s jejich datovými typy a také návratové hodnoty metod.

Protože se očekává, že CRM bude patřit mezi rozsáhlejší informační systémy, byla již na začátku návrhu aplikace rozdělena podle modelu MVC (Model, View, Controller – softwarová architektura oddělující aplikační logiku, datové třídy a grafická rozhraní aplikace).

## Struktura CRM a funkce tříd:

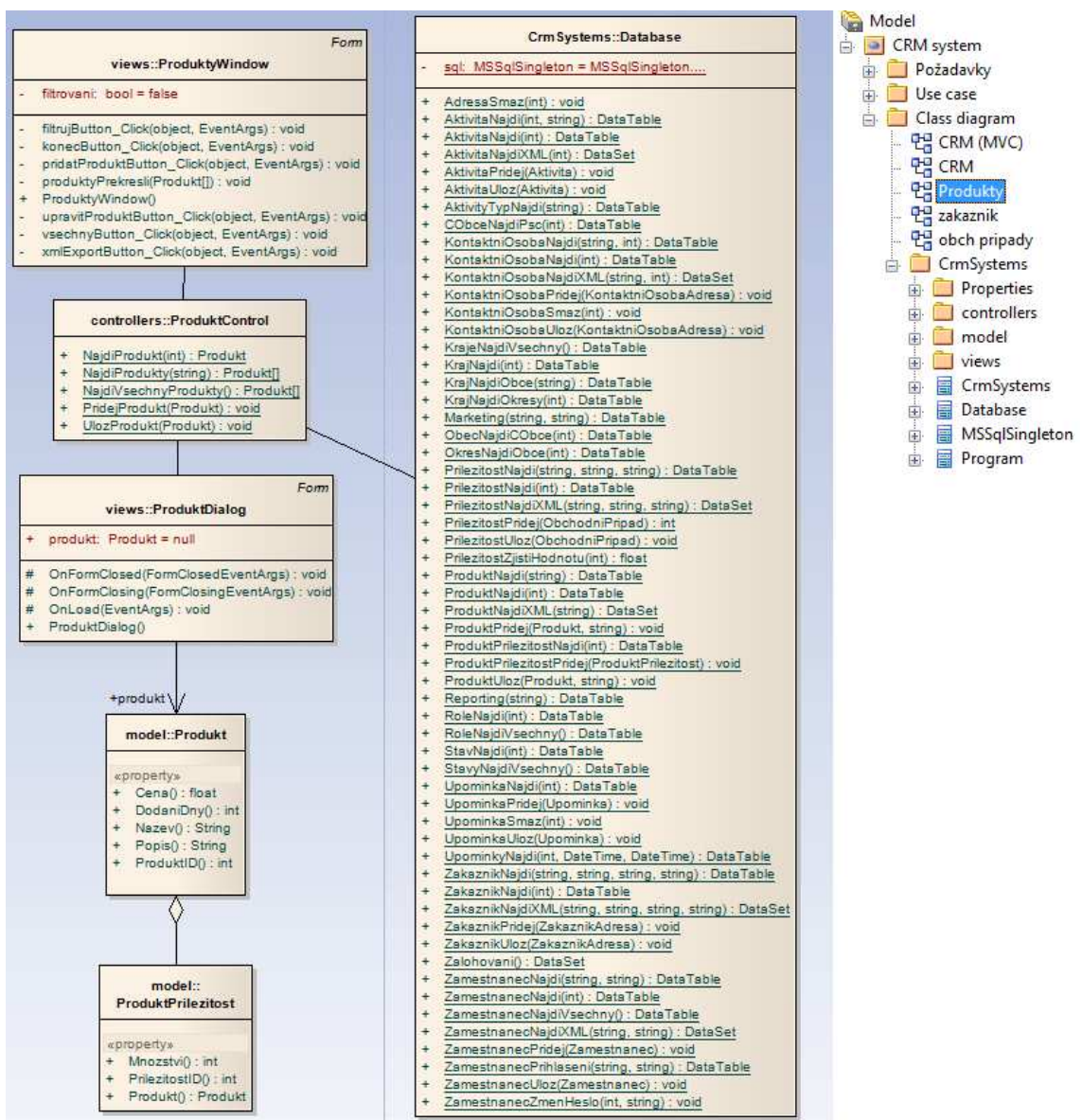
- Model – obsahuje datové třídy, které reprezentují informace uložené v centrální bázi dat.
- View (grafické rozhraní) – skládá se formulářových oken a dialogů.
- Controller (řídící třídy) – tato část obsahuje převážnou část aplikační logiky. Zde dochází ke zpracování dat z modelu, jejich transformaci a předání do formulářů.
- Třída database – zde jsou implementovány metody pro práci s databází. Všechny SQL dotazy (select, insert, update, delete) jsou umístěny v této jedné třídě. Výhoda tohoto řešení je v tom, že při změně databázového serveru za jiný stačí přeprogramovat metody v této jedné třídě a není nutno procházet všechny datové třídy.



Obrázek 12: Package diagram návrhových tříd.

- Třída MSSqlSingleton – poskytuje připojení k databázi pomocí návrhového vzoru Singleton.

Vzájemné propojení architektury MVC a seznam tříd obsažených v jednotlivých částech je možné vidět na obrázku číslo 12. Protože celý CRM systém obsahuje velké množství tříd (jejich kompletní výčet je v projektu v Enterprise Architectu), tak jen pro představu lze na obrázku číslo 13 vidět návrhové třídy pro modul obstarávající správu produktů.



Obrázek 13: Implementované třídy pro správu produktů.

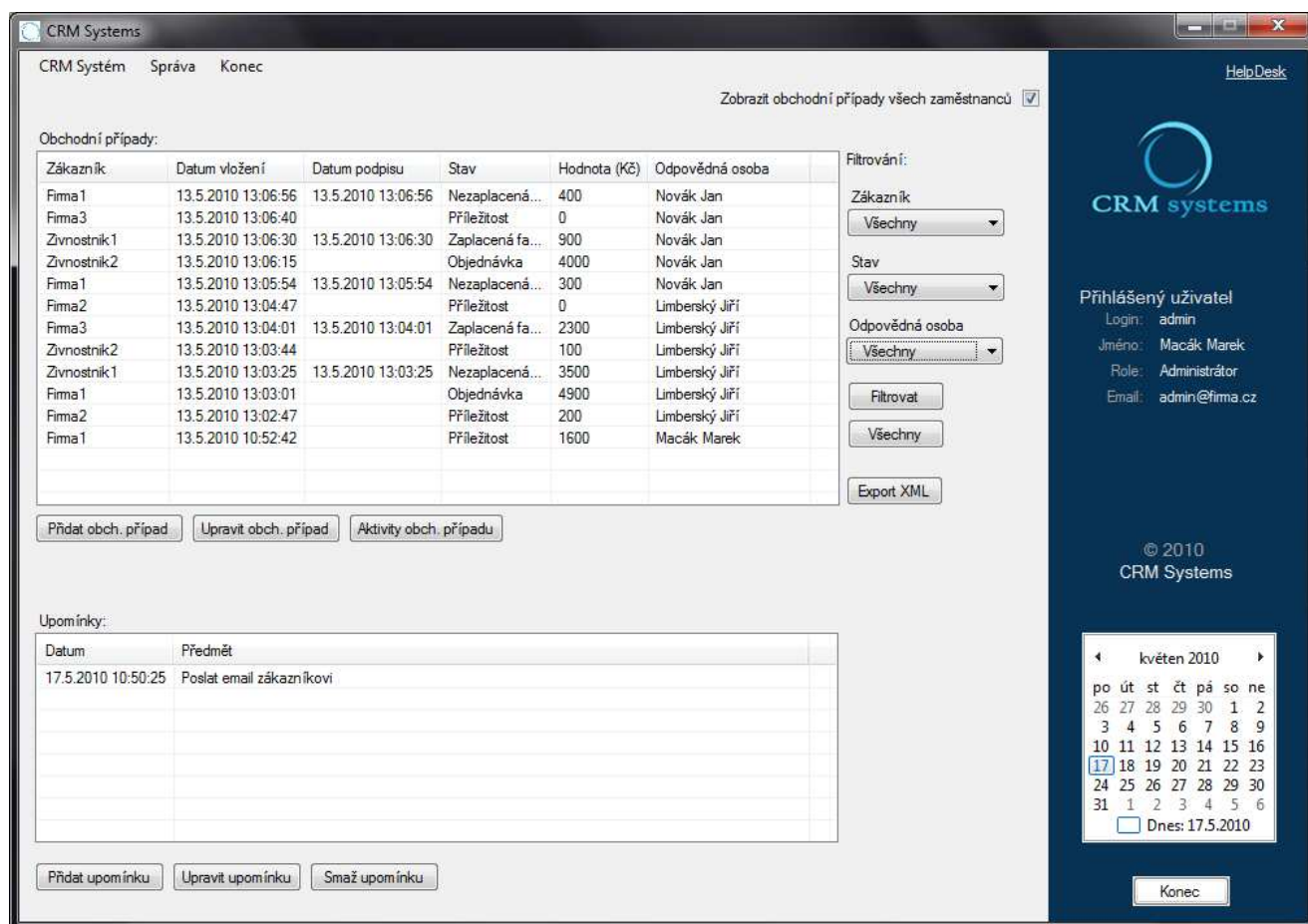


### 3.2.5 Postup implementace

V tuto chvíli se projekt nachází ve stavu ukončení analýzy. Všechny základní požadavky již byly stanoveny a nyní budou přibývat jen drobná doladění. Během předchozích kroků došlo k vytvoření analytických tříd a návrhu databáze, který byl použit k vygenerování kódu pro vytvoření databázových objektů.

Postup implementace probíhal v následujících krocích:

- vytvoření datových tříd podle analýzy návrhu,
- naprogramování kontrolních tříd, které se starají o aplikační logiku a obstarávají funkci rozhraní mezi daty (vrstva model) a jejich grafickou prezentací (vrstva view),
- navržení hlavního formulářového okna aplikace (obrázek číslo 14),
- navržení přihlašovacího okna a dialogů, ošetření vstupů.



Obrázek 14: Hlavní formulářové okno programu.

### 3.2.6 Řešené problémy při implementaci

Prvním zajímavým řešeným problémem byl způsob ošetření situace, kdy se uživatel pokusí spustit aplikaci v situaci, že nebude mít k dispozici databázi umístěnou na centrálním serveru. Ať už tato situace nastane vinou uživatele (odpojí počítač od sítě či si odnese notebook domů) nebo vinou selhání serveru, měla by tato neočekávaná situace být ošetřena, jinak dojde k fatálnímu selhání a program se ukončí.

```
using (SqlConnection conn = new SqlConnection(connStr)) {  
    SqlCommand command = new SqlCommand(queryString, conn);  
    try {  
        command.Connection.Open();  
        command.ExecuteNonQuery();  
    }  
    catch (SqlException ex) {  
        int chyba = ex.ErrorCode;  
        return false;  
    }  
}  
return true;
```

Obrázek 15: Ošetření výjimky při práci s databází.

Na obrázku číslo 15 je vložena část funkce, která se stará o ošetření stavu, kdy je databáze nedostupná. Jedná se o kód, kdy se vytvoří připojení do databáze a v bloku try se provede pokus o získání hodnot. Pokud databáze nebude z jakéhokoli důvodu dostupná, nastane zachycení výjimky v bloku catch a funkce vrátí hodnotu false.

Takto je ošetřeno například spuštění aplikace. Ještě před samotným přihlášením se provede test spojení s databází, pokud není úspěšný, aplikace se ukončí.

Další problém vyvstal při ověřování uživatelů. Heslo se totiž v databázi ukládá v podobě otisku původního hesla a nastal problém, jakou hashovací funkci zvolit. Nakonec byla vybrána asi nejpoužívanější funkce md5 (funkce vytvářející otisk původního řetězce vždy 32 znaků dlouhý), která je přímo implementována v PHP (ve webové aplikaci stačilo tuto funkci pouze zavolat), ale v C# se musí použít trochu složitější konstrukce, aby byl vytvořen stejný otisk. Na obrázku číslo 16 je možné vidět vytvořenou funkci, která vrací stejný řetězec jako funkce md5 v PHP.

```
public static string HashMD5(string heslo) {  
    using (MD5CryptoServiceProvider md5 = new  
        MD5CryptoServiceProvider()) {  
        byte[] data = Encoding.ASCII.GetBytes(heslo);  
        data = md5.ComputeHash(data);  
        string hesloHash = "";  
        for (int i = 0; i < data.Length; i++)  
            hesloHash += data[i].ToString("x2").ToLower();  
        return hesloHash;  
    }  
}
```

Obrázek 16: Funkce vracějící stejný otisk hesla jako funkce md5 v PHP, zdroj: Igor Kulman [14]

Při implementaci systému došlo také na ošetřování vstupních hodnot ve formulářových polích. Všechny zadávané hodnoty jsou kontrolovány na maximální povolenou délku a správnost datového typu. U většiny vstupů se tedy jedná jen o kontrolu, zda vstupuje číslo či řetězec znaků, ale u emailové adresy bylo požadováno ověřovat její validitu.

Na obrázku číslo 17 je vložena funkce *JeEmail*, která vrací hodnotu true, když je vstupním parametrem emailová adresa odpovídající regulárnímu výrazu.

```
public static bool JeEmail(string retezec) {  
    string vzor = @"^[-a-zA-Z0-9][-a-zA-Z0-9]*@[-a-zA-Z0-9](\.[-a-zA-Z0-9]+)*\.(com|info|name|aero|[a-zA-Z]{2,3})$";  
  
    Regex reg = new Regex(vzor, RegexOptions.IgnorePatternWhitespace);  
  
    if (string.IsNullOrEmpty(retezec)) return false;  
    return reg.IsMatch(retezec);  
}
```

Obrázek 17: Funkce k ověření správnosti emailové adresy.

Obrázek číslo 18 popisuje část kódu, který byl využit pro vykreslování koláčových grafů v marketingu a reportingových funkcích.

Proměnná *hodnoty* představuje pole vstupních hodnot pro vykreslení, proměnná *suma* pak obsahuje jejich součet. Proměnné *poziceX* a *poziceY* udávají levý horní roh vykreslovaného grafu a proměnné *vyska* a *sirka* udávají rozměry výsledného koláčového grafu. Položka *barvy* představuje pole barev použitých pro odlišení jednotlivých položek v grafu. Obsahem proměnné *posun* je velikost středového úhlu, pomocí kterého je docíleno rotací kruhových výsečí v koláčovém grafu.

```

Graphics graph = CreateGraphics();
for (int i = 0; i < hodnoty.Length; i++) {
    suma += hodnoty[i];
}
float posun = 0, jednotka = 360 / suma;

for (int i = 0; i < hodnoty.Length; i++) {
    graph.FillPie(new SolidBrush(barvy[i]), poziceX, poziceY, sirka,
        vyska, posun, hodnoty[i] * jednotka);
    posun += jednotka * hodnoty[i];
    if (i == hodnoty.Length - 1) {
        graph.FillPie(new SolidBrush(barvy[i]), poziceX, poziceY, sirka,
            vyska, posun, 360 - posun);
    }
}
graph.Dispose();

```

Obrázek 18: Kód použitý k vykreslení kruhového grafu (zdroj: Amadeo Mareš [3])

### 3.3 Zprovoznění aplikace

Pro bezchybný provoz programu je nutné:

- nainstalovat Microsoft SQL Server,
- nainstalovat rozhraní pro správu SQL serveru (doporučen je Microsoft SQL Server Management Studio Express),
- povolit ověřování oproti SQL účtům (v Management studiu zvolit vlastnosti dané databázové instance, záložka *security*),
- vytvořit databázi *Diplomka*,

- vytvořit uživatele *diplomka* s heslem *9orfk4* a přidělit mu příslušná oprávnění na databázi (pokud není systémový správce),
- obnovit databázi z příloženého souboru *databaze.bak* (některé tabulky již obsahují data a je jednodušší databázi obnovit než vytvářet znovu),
- spustit soubor *CrmSystems.exe* (nebo ve Visual Studiu otevřít projekt a vytvořit si vlastní sestavení) a přihlásit se do aplikace.

Pro testovací účely byly v systému vytvořeny zkušební účty, které se obnovily společně se zbytkem databáze ze zálohy. Přihlašovací jména a hesla těchto uživatelů shrnuje tabulka číslo 1.

Přihlašovací jméno	Heslo	Role
admin	heslo	administrátor
manager	heslo	manažer
obchodnik	heslo	obchodník

*Tabulka 1: Seznam výchozích zkušebních účtů.*

### 3.3.1 Administrace CRM systému

Pouze je-li přihlášen uživatel s oprávněním administrátor, má k dispozici nabídku *Správa > Uživatelé*, odkud má možnost přidávat a editovat uživatelské účty. Je-li označen nějaký účet a je vybrána možnost *upravit uživatele* nebo možnost *přidat*, otevře se dialogové okno (obrázek číslo 19), které nabízí správu konkrétního uživatele.

Všechna přihlašovací jména musí být pro celý systém jedinečná. Uživateli je přidělena systémová role (administrátor, manažer, obchodník), vždy právě jedna.

Druhou nabídkou, kterou vidí pouze administrátor je zálohování. Tato volba vytvoří soubor s kompletním exportem celé databáze.

Obrázek 19: Dialogové okno pro přidávání (editaci) uživatele.

### 3.3.2 Návod k obsluze systému

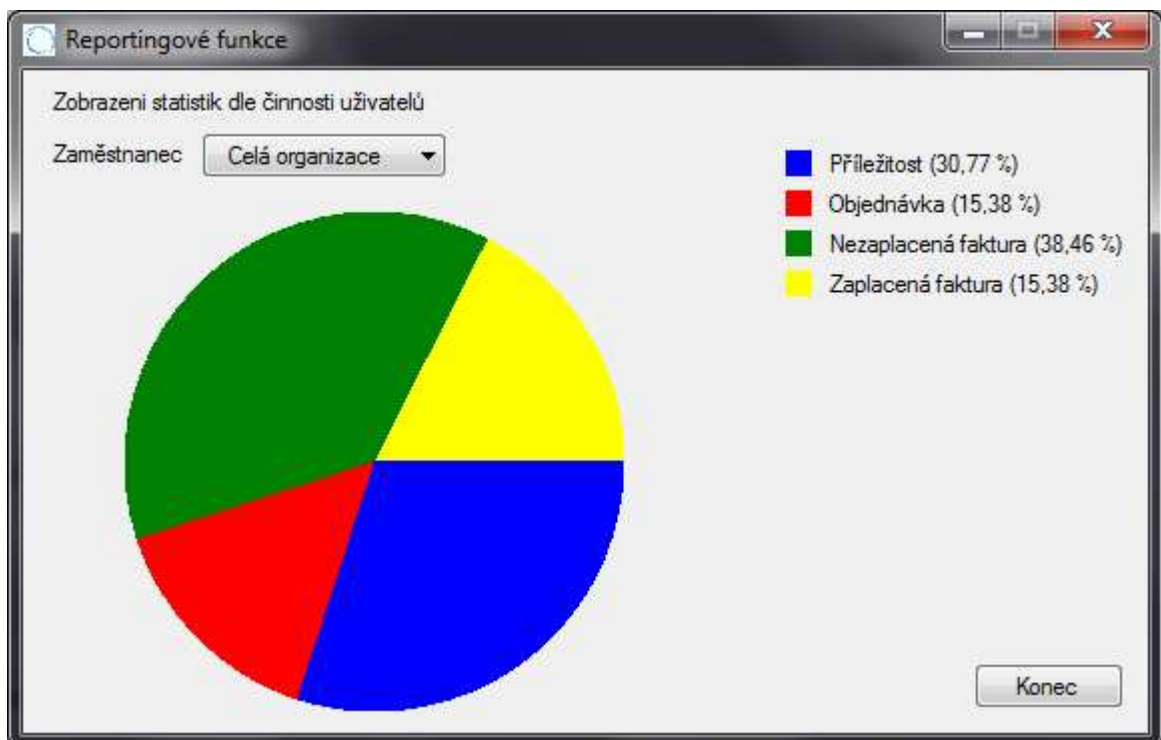
Po přihlášení uživatele do systému se otevře hlavní okno aplikace. Odtud uživatel může spravovat své obchodní případy, přidávat a editovat zákazníky (obrázek číslo 20) s jejich kontaktními osobami, využívat kalendáře s upomínkami a další možnosti CRM systému.

Obrázek 20: Dialogové okno pro přidávání (editaci) zákazníka.

V uživatelské roli manažer je navíc v pravém horním rohu zatrhávací políčko, jehož zaškrtnutím se manažerovi zobrazí obchodní případy ostatních zaměstnanců (pro prohlížení i úpravy). Takto zobrazené záznamy lze filtrovat a následně například exportovat do xml souboru.

Dalšími funkcemi, které se zobrazí jen v případě, že je přihlášen manažer nebo administrátor, jsou správa produktů, marketing a reporty.

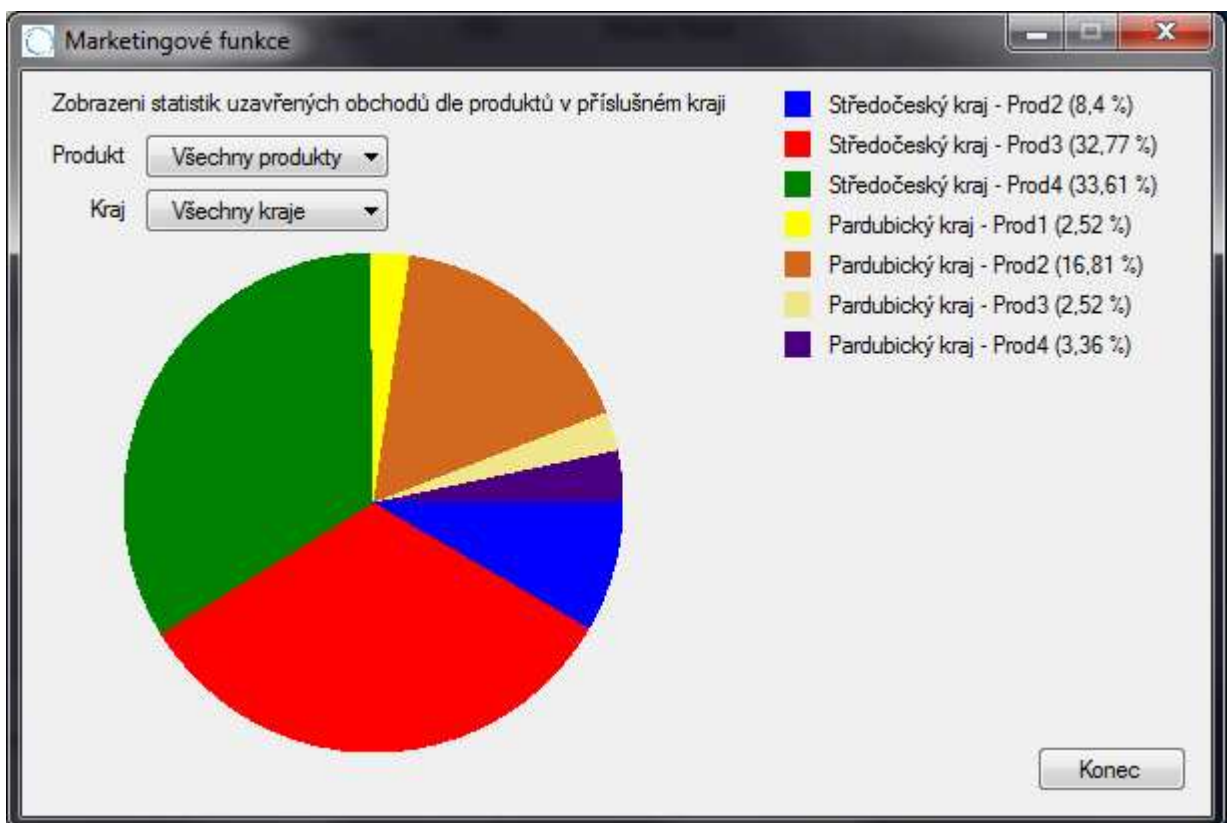
Po zvolení nabídky reporty se otevře nové dialogové okno, kde má manažer možnost sledovat úspěšnost, jak jednotlivých zaměstnanců, tak celé organizace. Úspěšností se v tomto případě myslí poměr příležitostí, objednávek a faktur. Více na obrázku číslo 21, kde je zobrazen kruhový graf s procentuálním vyjádřením jednotlivých položek.



Obrázek 21: Okno s kruhovým grafem reportingové funkce.



Podporou pro marketing jsou statistiky prodeje jednotlivých produktů v různých krajích. Manažer si může nechat zobrazit množství prodeje všech produktů ve všech krajích nebo si může zvolit vybraný produkt ve všech krajích nebo si zobrazit všechny prodávané produkty v jednom kraji, aby zjistil, který produkt má největší přínos pro firmu. Statistická funkce pro marketing je zobrazena na obrázku číslo 22.



Obrázek 22: Okno s kruhovým grafem marketingové funkce.

### 3.3.3 Zaškolení uživatelů

Před uvedením systému do ostrého provozu ve firmě je nutné nechat proškolit všechny zaměstnance, aby byli schopni využívat všech funkcí CRM systému. Zaměstnanec, který prošel praktickými ukázkami a měl možnost systém vyzkoušet ve fázi testování, je schopen si mnohem rychleji zvyknout na nové prostředí a snadno se naučí automatizovat jednotlivé úkony.

### **3.3.4 Migrace dat**

Dochází-li ve firmě k náhradě jednoho programového vybavení za druhé, bude třeba do nového systému přesunout i historická data z původního systému. Ve většině případů se bude struktura staré a nové databáze lišit, a proto bude potřeba data analyzovat, případně odstranit duplicity či nekonzistentní data a poté provést jednorázový (ruční či automatizovaný) import dat do nového systému.

## 4 Závěr

Cílem diplomové práce byl návrh a implementace intranetové aplikace typu CRM. Z teoretické části vyplynuly základní požadavky na funkce a vlastnosti systému tohoto typu. V praktické části práce je zachycen postup vývoje systému v jednotlivých krocích, sestavování požadavků, analýzy a vytváření případů užití. Na základě těchto artefaktů byla vybrána k implementaci technologie .NET (programovací jazyk C#) v prostředí Microsoft Visual Studio za podpory databázového serveru Microsoft SQL Server.

Byl vytvořen datový model a návrh analytických tříd, které se staly základem vyvíjené aplikace. Ta je dokončena a nyní se nachází ve stavu testování.

Již při vytváření programu bylo počítáno s možností kustomizace a lokalizace tvořené aplikace. Před jejím zavedením do ostrého provozu bude možné aplikaci upravit dle požadavků konkrétního zákazníka – aplikační logika by se však již měnit neměla.

## POUŽITÁ LITERATURA

- [1] GÁLA, Libor; POUR, Jan; TOMAN, Prokop. *Podniková informatika*. První vydání. Praha: Grada Publishing, 2006. 484 s. ISBN 80-247-1278-4.
- [2] CHLEBOVSKÝ, Vít. *CRM: Řízení vztahů se zákazníky*. Vydání první. Brno: Computer Press, 2005. 190 s. ISBN 80-251-0798-1.
- [3] MAREŠ, Amadeo. *1001 tipů a triků pro C#*. Vydání první. Brno : Computer Press, 2008. 360 s. ISBN 978-80-251-2125-2.
- [4] ŠMÍD, Vladimír. *Management informačního systému* [online]. 2002 [cit. 2010-05-15]. Pojem informačního systému. Dostupné z WWW: <<http://www.fi.muni.cz/~smid/mis-infsys.htm>>.
- [5] Customer relationship management. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, , last modified on 2010-05-11 [cit. 2010-05-15]. Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Customer\\_relationship\\_management](http://cs.wikipedia.org/wiki/Customer_relationship_management)>.
- [6] Enterprise resource planning. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, , last modified on 2010-05-15 [cit. 2010-05-15]. Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Enterprise\\_resource\\_planning](http://cs.wikipedia.org/wiki/Enterprise_resource_planning)>.
- [7] ŽEBRÁK, Miroslav. Integrace CRM se systémy třetích stran. *Business World* [online]. 2009-04-14, [cit. 2010-05-15]. Dostupný z WWW: <<http://businessworld.cz/rady-nazory-zkusenosti/Integrace-CRM-se-systemy-tretich-stran-4483>>.
- [8] Microsoft SQL Server. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, , last modified on 2010-05-12 [cit. 2010-05-15]. Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](http://cs.wikipedia.org/wiki/Microsoft_SQL_Server)>.

- [9] PATTON, Susannah; WAILGUM, Thomas; TABER, David. CRM aneb jak si udržet zákazníky. *Business World* [online]. 2010-03-03, [cit. 2010-05-15]. Dostupný z WWW: <<http://businessworld.cz/podnikove-is/crm-aneb-jak-si-udrzet-zakazniky-5675>>.
- [10] KODERA, Jiří. Kam se ubírají modely plateb za software v době SaaS. *Lupa: Server o českém internetu* [online]. 2008-04-02, [cit. 2010-05-15]. Dostupný z WWW: <<http://www.lupa.cz/clanky/kam-se-ubiraji-modely-plateb-za-software/>>.
- [11] Enterprise Architect. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, , last modified on 2010-05-08 [cit. 2010-05-15]. Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Enterprise\\_Architect](http://cs.wikipedia.org/wiki/Enterprise_Architect)>.
- [12] Toad Data Modeler. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, , last modified on 2010-04-16 [cit. 2010-05-15]. Dostupné z WWW: <[http://en.wikipedia.org/wiki/Toad\\_Data\\_Modeler](http://en.wikipedia.org/wiki/Toad_Data_Modeler)>.
- [13] Visual Studio. In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida): Wikipedia Foundation, , last modified on 2010-05-13 [cit. 2010-05-15]. Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Visual\\_Studio](http://cs.wikipedia.org/wiki/Visual_Studio)>.
- [14] KULMAN, Igor. *Osobný weblog* [online]. 2010-02-07 [cit. 2010-05-15]. Ako v C# vygynerovať MD5 hash. Dostupné z WWW: <<http://www.kulman.sk/sk/content/ako-v-csharp-vygynerovat-md5-hash>>.