

PŘÍRODOVĚDECKÁ FAKULTA UNIVERZITY PALACKÉHO
KATEDRA INFORMATIKY

BAKALÁŘSKÁ PRÁCE

Ekonomický systém pro řízení sítě prodejen



2013

David Staněk

Anotace

Cílem projektu bylo vytvořit aplikaci pro výrobní společnost ZD Vícov, která bude evidovat pohyb zboží a tržeb jednotlivých provozoven patřících této společnosti, ze kterých následně provede ekonomické vyhodnocení hospodaření každé provozovny. Výstupem programu bude export konsolidovaných údajů do účetního systému.

Děkuji panu Ing. Jiřímu Hronkovi za náměty, připomínky a čas, který mi věnoval v souvislosti s dokončením mé bakalářské práce.

Obsah

1. Úvod	7
1.1. Současný stav	7
1.2. Cíle projektu	7
1.3. Požadavky na aplikaci	8
2. Technické řešení	10
2.1. Entity Framework	10
2.1.1. Historie a vývoj	10
2.1.2. Architektura EF	11
2.2. Datový model	13
2.3. Struktura aplikace	16
3. Uživatelská část	16
3.1. Systémové požadavky	16
3.2. Instalace programu	17
3.3. Ovládání programu	21
3.4. Práce s programem	22
3.4.1. Nastavení	22
3.4.2. Provozovny	24
3.5. Výstupy	27
Závěr	28
Reference	29
4. Obsah příloženého CD	30

Seznam obrázků

1.	Model architektury s použitím ADO.NET Entity Frameworku . . .	11
2.	Entity Framework a EDMX soubor	13
3.	ER Diagram Aplikace Prodejny	15
4.	Rozhraní programu Prodejny	21

Seznam tabulek

1. Tabulka verzí Entity Framework.	12
--	----

1. Úvod

Aplikace Prodejny - Ekonomický systém pro řízení sítě prodejen vznikla jako bakalářská práce při studiu na Univerzitě Palackého v Olomouci, obor aplikovaná informatika. Druhým důvodem vzniku této aplikace byla potřeba nahradit současnou nevyhovující agendu sběru a vyhodnocování údajů z podnikových prodejen výrobního podniku ZD Vícov.

Text této bakalářské práce je rozdělen do tří stěžejních částí. V první části je popis řešené problematiky a současný stav. Druhá část se věnuje technologiím použitým k vývoji aplikace a její struktuře. Třetí část je uživatelská, která obsahuje popis uživatelského rozhraní a způsob práce s aplikací.

1.1. Současný stav

Společnost ZD Vícov je zemědělský podnik, který se zabývá rostlinou i živočišnou výrobou. Vlastní jatečný závod v Lešanech, který zpracovává maso jak z vlastního chovu, tak i maso pocházející z chovů zemědělců z nejbližšího okolí. Společnost vlastní v současnosti 11 prodejen, ve kterých je nabízeno jak masné zboží z vlastní jatky, tak i nakupované zboží od externích dodavatelů včetně doplňkového sortimentu jako je pečivo, koření a různé potravinové přísady.

Od počátku bylo nutné sledovat a vyhodnocovat pohyb zboží a tržeb na jednotlivých prodejních a následně zpracování výsledných dat centrálním účetním systémem. K tomuto byl až doposud využíván tabulkový procesor Microsoft Excel, ve kterém byly předdefinovány formuláře, do kterých obsluha vkládala údaje za jednotlivé prodejny. Toto řešení se se vzrůstajícím počtem prodejen, objemem zboží a rozšiřováním prodejního sortimentu stalo velmi nepraktické a do budoucna neudržitelné. Jakákoliv změna metodiky vyhodnocování znamenala provést potřebné změny na několika místech v excelovském sešitu, což nevyhnutelně často vedlo k chybám, které byly většinou těžko odhalitelné. Například přidání nebo zrušení prodejny byl úkol na několik desítek minut. Z toho důvodu byla s povděkem přijata moje nabídka zpracovat tuto problematiku jako svou bakalářskou práci.

1.2. Cíle projektu

Cílem tohoto projektu bylo vytvořit aplikaci, která bude přizpůsobena potřebám uvedené společnosti a co možná nejvíce minimalizuje úkony potřebné k evidenci zboží a hotovosti od příjmu zboží na prodejnu až po výsledné zaevidování všech transakcí v účetním systému. Zároveň bude snadno spravovatelná a umožní sledovat legislativní i vnitropodnikové metodické změny. Původní myšlenka byla taková, že každá prodejna bude mít svůj počítač, na kterém bude denně evidovat nákupy a tržby. Tyto údaje by se následně přenášely do centrály, kde by potom

docházelo k jejich zpracování. Při hlubší analýze se ovšem tato myšlenka ukázala jako nereálná. Dovybavení prodejen osobními počítači by vyžadovalo nepřiměřené náklady a jako nemalý problém se ukázalo i vyškolení obsluhy. Toto řešení by navíc nepřineslo zvýšení efektivity celého procesu, protože většina těchto podkladů se nejdříve připravuje na papíře a až poté se může převést do elektronické podoby. Z toho důvodu konečné řešení počítá pouze s jednou stanicí s lokální databází, na které bude obsluha provádět všechny činnosti od přepisování vstupních údajů jednotlivých prodejen až po konečný export souhrnných údajů do účetního systému.

Vzhledem k tomu, že nejčastěji prováděnou činností v aplikaci bude rutinní vstupování dílčích údajů z jednotlivých prodejen, byl při návrhu uživatelského rozhraní kladen důraz na co největší ergonomii těchto činností, což může někdy vést ke snížení jeho intuitivity pro nové uživatele. Výstupní sestavy by měly zachovávat rozložení sestav používaných v současnosti.

1.3. Požadavky na aplikaci

Personál prodejny bude v pravidelných cyklech předávat do centrály údaje, které následně obsluha aplikace vloží do databáze. Denně se budou předávat tyto údaje:

- denní hodnota převedeného zboží z centrály včetně čísla dokladu a výše DPH
- denní hodnota nakoupeného zboží od externích dodavatelů včetně čísla nákupního dokladu a výše DPH
- denní hodnota zakoupeného režijního materiálu včetně čísla dokladu a výše DPH
- výše denní tržby
- výše odvodu hotovosti do banky.

Na konci každého měsíčního období a v případě požadavku i v průběhu tohoto období bude zpracována inventura každé prodejny, na jejíž základě budou do centrály předány údaje:

- stav hotovosti v pokladně
- stav zboží na prodejně ve snížené sazbě DPH
- stav zboží na prodejně ve zvýšené sazbě DPH

K přenosu údajů se budou používat předtištěné formuláře, které se používají i v současnosti.

Pro výpočet prodejních cen se bude používat pevný koeficient, který bude různý pro zboží z vlastní jatky a pro zboží od externích dodavatelů. Tyto koeficienty bude možné aktualizovat vždy ke konci účetního období. Na základě těchto koeficientů a údajů, které byly předány prodejny se v aplikaci vytvoří výkaz hospodaření každé prodejny, který bude možné následně vytisknout. Výkaz bude obsahovat:

- souhrn příjmů zboží na prodejnu
- předpis předpokládaných tržeb v daném období
- skutečné tržby včetně výše daně z přidané hodnoty
- celkový výsledek hospodaření

Dále bude aplikace umožňovat zobrazit přehled výsledků inventur za dané období. Konsolidované účetní údaje za všechny prodejny budou na konci každého období exportovány do textového souboru, který bude použit jako vstupní soubor pro import do účetního systému. Struktura věty tohoto textového souboru bude upřesněna s dodavatelem účetního systému.

2. Technické řešení

Vzhledem k charakteru požadovaného řešení, byla zvolena desktopová aplikace s MDI formulářem. Vývojovým prostředím bylo Microsoft Visual Studio 2012 a programovací jazyk C#, který využívá běhové prostředí .NET Framework. Jako databáze byla zvolena Microsoft SQL Server Compact Edition ve verzi 4.0. Pro přístup k této databázi byla použita technologie objektově relačního mapování (ORM) ADO.NET Entity Framework 5.0 (viz kap.2.1.) společně s dotazovacím jazykem Language-Integrated Query (LINQ).

Souhrn použitých technologií:

- Microsoft Visual Studio 2012
- programovací jazyk C# 5.0
- .NET Framework 4.5
- Entity Framework 5.0
- Linq
- Microsoft SQL Server CE 4.0

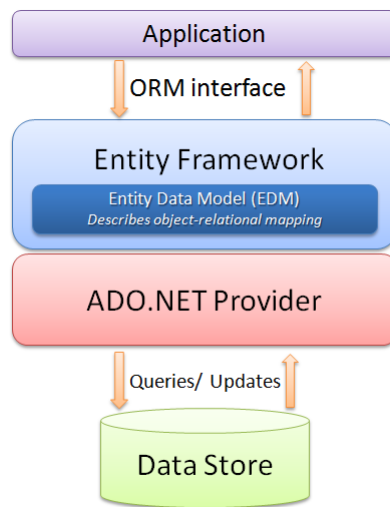
2.1. Entity Framework

Objektově relační mapování je technologie, která řeší rozpor mezi objektově orientovaným programováním a ukládáním dat do relačních databází, které vychází z databázového schématu. Z databáze pomocí ORM tedy místo pole hodnot dostáváme rovnou objekty, které obsahují svoje metody a vlastnosti. V jazyce SQL tedy vůbec nekomunikujeme, tabulky v databázi vidíme jako kolekce objektů, se kterými můžeme pracovat běžnými prostředky programovacího jazyka. Pracujeme tedy na vyšší úrovni abstrakce.

2.1.1. Historie a vývoj

Entity Framework není prvním ani jediným ORM nástrojem pro .NET Framework a Visual Studio, ale je to první nástroj, který je přímo do Visual Studia a .NET Frameworku integrován. Dalšími ORM nástroji používané pro vývoj aplikací a které je možné integrovat do Visual Studia jsou například NHibernate nebo OpenAccess.

První vydání Entity Framework EFv3.5 přišlo v roce 2008 jako součást .NET Framework 3.5 společně se Service Pack 1 pro Visual Studio 2008. Tato verze se ovšem setkala pro její zásadní nedostatky s hlasitou kritikou vývojářů. Druhé vydání přišlo v roce 2010 a bylo součástí .NET Frameworku 4.0. Microsoft v ní



Obrázek 1. Model architektury s použitím ADO.NET Entity Frameworku

zohlednil připomínky k prvnímu vydání a označil verzí EFv4.0. Tato verze již byla plně připravena k produkčnímu nasazení, další verze potom kromě oprav chyb a zvýšení výkonu postupně přidávaly další funkce. Poslední stabilní verzí, je EFv5.0, která je součástí .NET Frameworku 4.5. Aplikace, napsané pod touto verzí mohou běžet i pod .NET Frameworkem 4.0, ale nevyužijí výhod a nových funkcí obsažených ve verzi 4.5. U následující verze EFv6.0 se Microsoft rozhodl otevřít zdrojový kód a zveřejní jej pod licenci Apache v2.

2.1.2. Architektura EF

Základním stavebním kamenem Entity Frameworku je Entity Data Model (EDM)(obr.2.). Je to množina konceptů, která popisuje spojení business modelu s databází, pomocí které lze načítat, upravovat, zapisovat a mazat záznamy v různých úložištích, ke kterým existuje provider s podporou Entity Frameworku. Existuje několik způsobů, jak je možné při vytváření EDM postupovat:

1. **Database First** - Tento přístup předpokládá již vytvořenou databázi, ze které Entity Framework automaticky vygeneruje EDM obsahující třídy a vlastnosti, které odpovídají tabulkám a sloupcům v mapované databázi.
2. **Model First** - Pomocí tohoto přístupu můžeme nejdříve namodelovat datový model EDM pomocí Entity Data Model Designeru, ze kterého lze následně vygenerovat prázdnou databázi s odpovídajícím schématem.
3. **Code First** - Tento způsob umožňuje definovat vlastní model pomocí CLR tříd, ze kterých lze potom vygenerovat databázové schéma, nebo namapovat

Verze	Popis
EF 5.0	Současná verze, která přináší výrazné zvýšení výkonu a větší integraci do Visual Studio 2012.
EF 4.3.1	Vylepšená podpora pro entity typu LocalDB pro uživatele Visual Studio 2012.
EF 4.3	Nový způsob přístupu Code First Migrations.
EF 4.1.1	Opravné vydání, přináší nové komponenty pro zjednodušení práce s Code First přístupem.
EF 4.1	První verze dostupná přes NuGet, zjednodušené DbContext API a nový způsob mapování pomocí Code First přístupu.
EF 4.0	Součást .NET Framework 4, nové funkce jsou podpora POCO, lazy loading a Model First přístup.
EF 3.5	První verze přišla s .NET Framework 3.5, poskytuje základní podporu pro ORM mapování pomocí Database First přístupu.

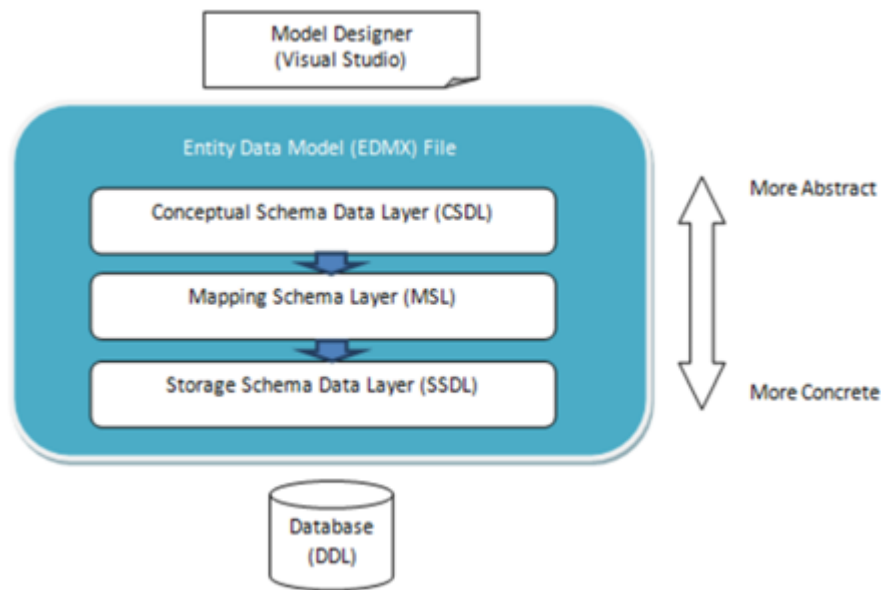
Tabulka 1. Tabulka verzí Entity Framework.

na již existující databázi.

4. **Code First Migrations** - Rozšiřuje předchozí přístup Code First o možnosti přizpůsobení při generování nového databázového schématu, nebo změnách EDM.

Entity Data Model je popsán pomocí metadat uložených v EDMX souboru (obr. 2.). Tento soubor je rozdělen do tří částí, z nichž každá pomocí jazyka XML popisuje jinou datovou vrstvu.

1. **Conceptual Schema Definition Language CSDL** - Tato část popisuje entity, vztahy mezi entitami a funkce které tvoří konceptuální model datové základny aplikace. Metadata popsána pomocí CSDL jazyka jsou využívána Entity Frameworkem k mapování entit a jejich vztahů na své datové protějšky v databázi.
2. **Store Schema Definition Language SSDL** - V této části jsou popsány informace o struktuře databáze, do které jsou ukládána data. Databázový model může být generován z již vytvořené databáze, nebo může být vygenerován na základě konceptuálního modelu.



Obrázek 2. Entity Framework a EDMX soubor

3. Mapping Specification Language - Poslední část udržuje informace mapování konceptuálního modelu na databázový model.

Tyto části EDMX souboru jsou generovány automaticky pomocí Entity Data Model Designeru, takže vývojář nemusí vůbec do jeho obsahu zasahovat. Jsou však situace, kdy jsou ruční zásahy do tohoto souboru nezbytné. Při kompilaci programu ve Visual Studiu jsou tyto jednotlivé části zkopírovány do třech oddělených XML souborů, které využívá Entity Framework pro běh aplikace. Prvním je csdl soubor, do kterého je uložen konceptuální model EDM, druhým je ssdl soubor, který obsahuje databázový model a posledním je soubor msl, který popisuje mapování konceptuálního modelu do databázového.

2.2. Datový model

Aplikace Prodejny využívá pro svůj běh databázi Microsoft SQL Server Compact Edition ve verzi 4.0. Tato databáze je určena pro použití v jednodušších aplikacích, kde se nepředpokládá, že se počet záznamů vyšplhá do milionových hodnot. Podstatný rozdíl oproti ostatním edicím MS SQL serveru je ten, že tato databáze neběží jako služba, ale je představována pouze jedním souborem SDF. To umožňuje velice snadnou distribuci databáze společně s aplikací. Stačí pouze zajistit instalaci příslušného runtime, nebo zajistit distribuci sedmi dll knihoven společně s aplikací.

Datový model aplikace Prodejny (obr. 3.) obsahuje 8 tabulek, z nichž 7 je

možné udržovat pomocí uživatelského rozhraní aplikace. Poslední tabulka VyslInventur slouží pouze k uchování vypočítaných výsledků a zároveň slouží jako zdroj pro tisk výsledných sestav.



Obrázek 3. ER Diagram Aplikace Prodejny

2.3. Struktura aplikace

Aplikace Prodejny byla navržena jako desktopová aplikace využívající třívrstvou architekturu. Aplikace je tedy rozdělena do několika vrstev, kde spolu sousedící vrstvy navzájem komunikují na úrovni objektů a jejich veřejných metod.

Architektura aplikace Prodejny:

- **Prezentační vrstva** - Je tvořena WinForm projektem Prodejny.exe, ve kterém jsou definice jednotlivých formulářů, které zároveň řeší základní validaci dat do těchto formulářů vkládaných.
- **Aplikační vrstva** - Je tvořena knihovnou Business.dll. Třídy v této knihovně zajišťují výpočty a operace prováděné mezi vstupně-výstupními požadavky prezentační a datové vrstvy.
- **Datová vrstva** - Datová vrstva je tvořena databází SQL Server CE a právě Entity Frameworkem. Protože aplikační vrstva nekomunikuje přímo s databází, ale s kontextem Entity Frameworku, můžeme někdy v tomto spojení hovořit o další vrstvě architektury.

3. Uživatelská část

Program Prodejny slouží výrobnímu podniku, který vlastní svou síť prodejen, k evidenci pohybu zboží a tržeb těchto provozoven, ze kterých následně provede ekonomické vyhodnocení hospodaření každé provozovny. Výsledná ekonomická data je možné exportovat do textového souboru, který může sloužit jako importní soubor do účetního systému.

3.1. Systémové požadavky

Program Prodejny využívá pro svou funkčnost běhové prostředí Microsoft .NET Framework 4.5 a databázi Microsoft SQL Server 4.0 Compact Edition. Požadavky na systém tedy vychází z požadavků na tento software:

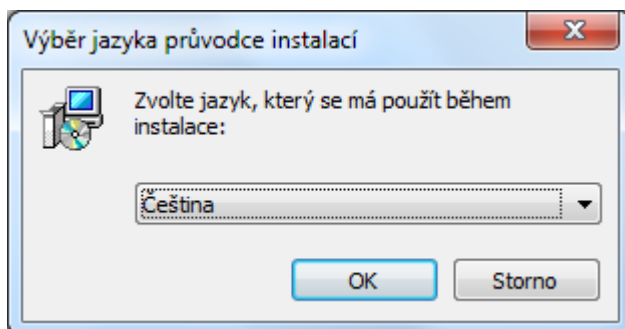
- Požadavky na operační systém
 - Windows Vista SP2 (x86 a x64)
 - Windows 7 SP1 (x86 a x64)
 - Windows Server 2008 R2 SP1 (x64)
 - Windows Server 2008 SP2 (x86 a x64)

- Požadavky na software
 - Microsoft .NET Framework 4.5
 - Microsoft SQL Server 4.0 Compact Edition
- Požadavky na hardware
 - Procesor s frekvencí 1 GHz nebo vyšší
 - 512 MB paměti RAM
 - 850 MB volného místa na pevném disku (x86)
 - 2 GB volného místa na pevném disku (x64)

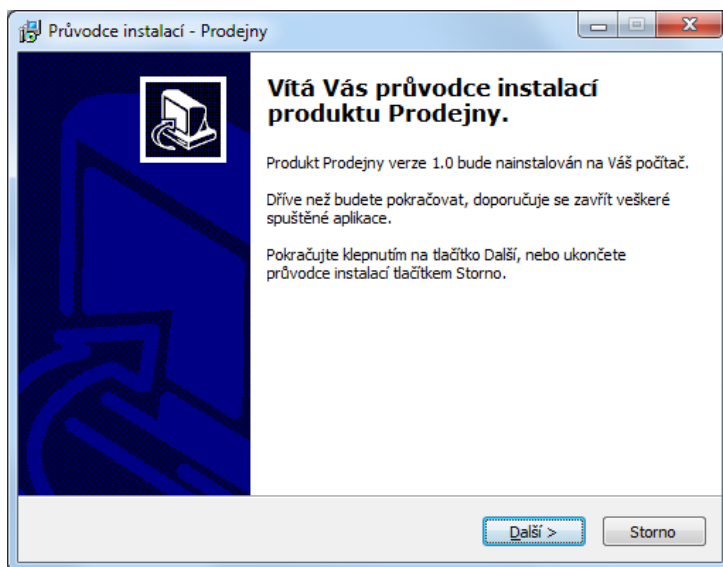
3.2. Instalace programu

Instalace programu se provádí pomocí instalátoru `setup.exe`, který po spuštění provede kontrolu na přítomnost .NET Frameworku 4.5 a runtime Microsoft SQL serveru 4.0 CE. V případě, že některá z těchto komponent v systému není nainstalovaná, instalátor nabídne možnost instalace.

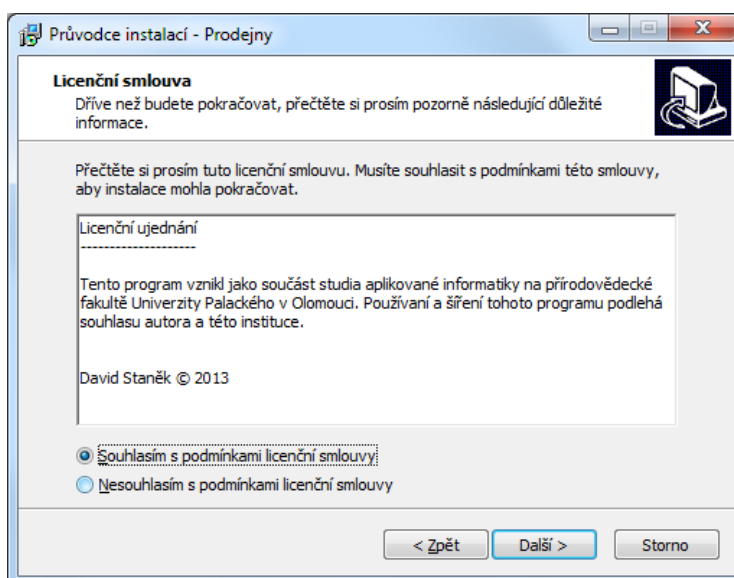
1. Zvolte jazyk, který bude použit během instalace programu a stiskněte tlačítko OK.



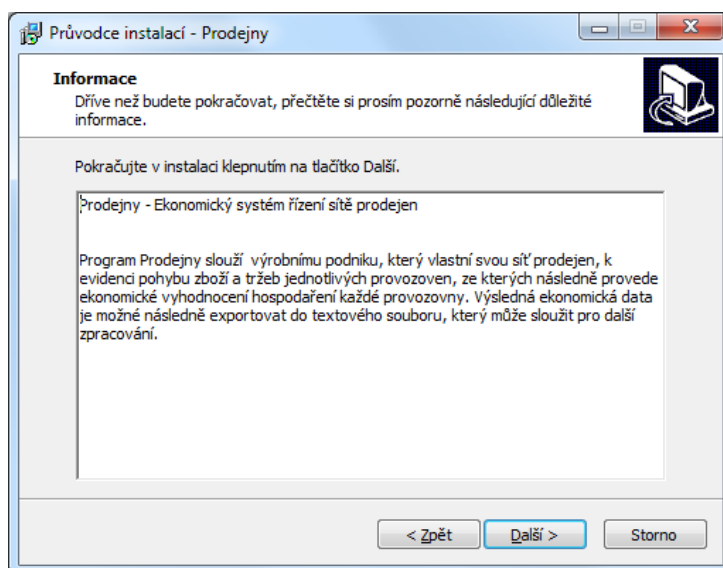
2. Spustí se průvodce instalačním procesem. Pokračujte dle pokynů na obrazovce.



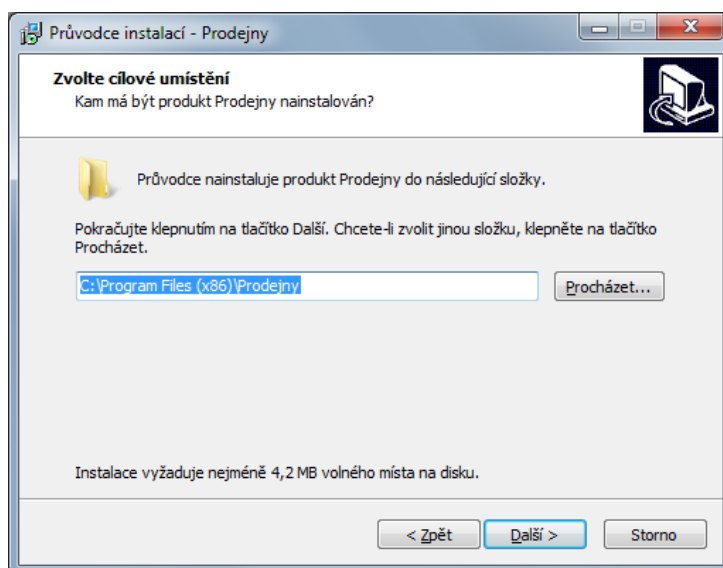
3. Přečtěte si podmínky použití programu a vyberte možnost **Souhlasím s podmínkami licenční smlouvy**.



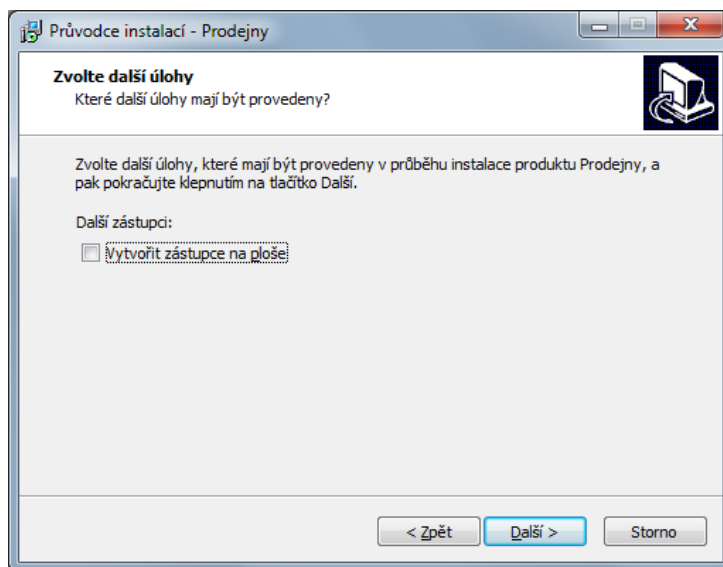
4. Přečtete si informace o programu a pokračujte v instalaci.



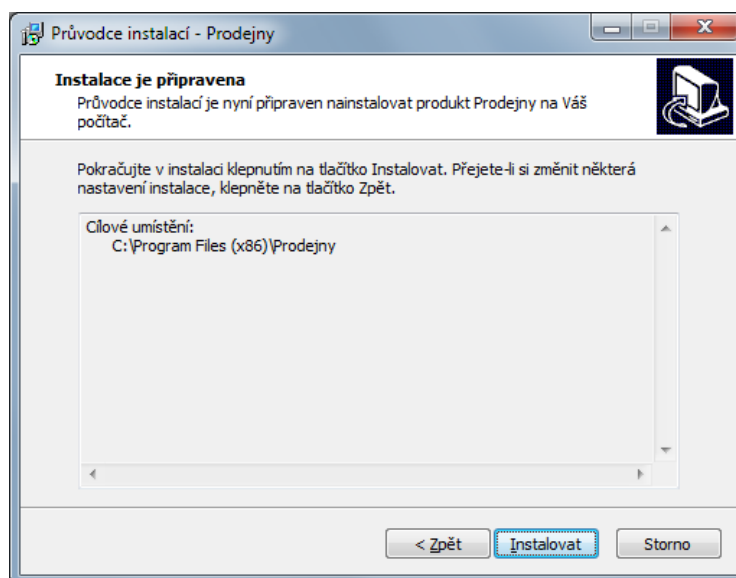
5. Zvolte adresář pro instalaci programu.



6. Zvolte možnost, má-li instalátor vytvořit zástupce. pro spuštění programu.



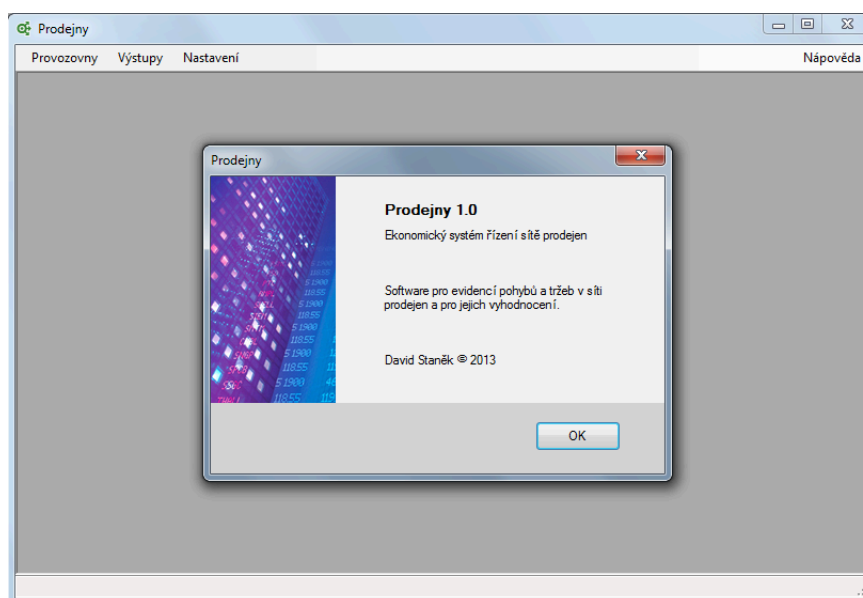
7. Stiskem tlačítka **Instalovat** instalátor provede instalaci programu Prodejny a všech součástí potřebných pro běh tohoto programu.



Odinstalace programu se provede pomocí tlačítka operačního systému Windows Start a dále **Všechny programy - Prodejny - Odinstalovat aplikaci Prodejny** nebo v Ovládací panelech systému Windows pomocí volby **Programy a funkce**. Zde v seznamu nainstalovaných programů vyberte položku **Prodejny** a stiskněte tlačítko **Odinstalovat**. Dále postupujte podle pokynů instalátoru.

3.3. Ovládání programu

Rámec aplikace tvoří hlavní okno, které je možné rozdělit na dvě části. V horní části je umístěné menu, pomocí kterého můžeme ovládat jednotlivé funkce programu. Spodní část okna slouží jako kontejner pro všechna nedialogová okna programu (obr. 4.).



Obrázek 4. Rozhraní programu Prodejny

Všechna okna pracující se seznamem položek jsou navržena se stejným rozhráním. V horní části je nástrojová lišta s ovládacími prvky pro přechod mezi jednotlivými položkami seznamu a v dolní části je potom samotný seznam. Pro přechod mezi jednotlivými položkami seznamu lze použít i kurzorové klávesy. Pro editaci záznamů lze použít klávesu F2. Nový záznam se vkládá vložením požadovaných údajů do posledního řádku seznamu. Provedené změny a nově přidané záznamy je nutné před zavřením okna uložit do databáze pomocí symbolu diskety v nástrojové liště v horní části okna.

Pro každou úlohu v aplikaci je k dispozici kontextová nápověda. Skiskem tlačítka F1 se v kterémkoliv okně programu otevře okno s příslušnou nápovědou k dané úloze.

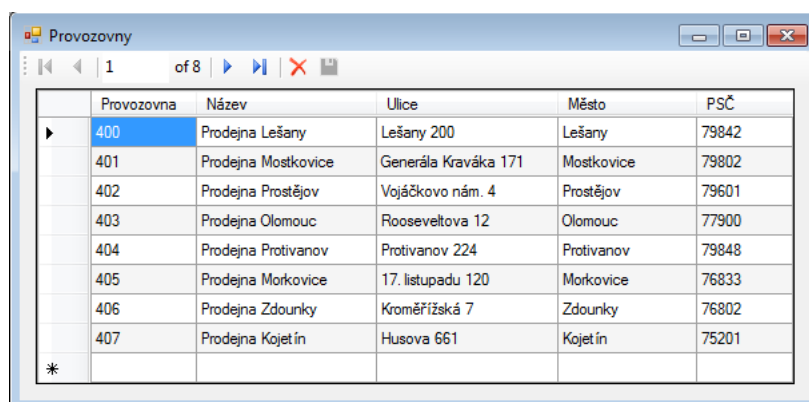
3.4. Práce s programem

Aby bylo možné do programu přidávat nové záznamy výkazů, tržeb a inventur jednotlivých prodejen, je nejdříve nutné naplnit číselníky jednotlivých prodejen a dodavatelů.

3.4.1. Nastavení

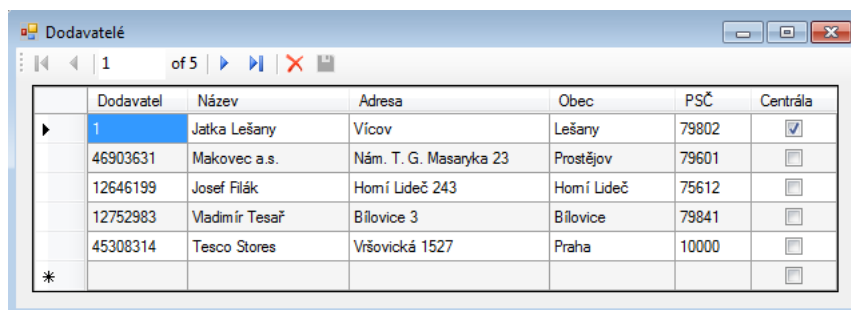
Pod položkou **Nastavení** hlavního menu najdeme číselníky systému, položky pro práci s databází a nastavení konstant používaných při výpočtech.

Provozovny - Slouží k editaci a vkládání nových prodejen do systému. Každá prodejna musí mít zadáno svoje unikátní ID a název.



Provozovna	Název	Ulice	Město	PSČ
400	Prodejna Lešany	Lešany 200	Lešany	79842
401	Prodejna Mostkovice	Generála Kraváka 171	Mostkovice	79802
402	Prodejna Prostějov	Vojáčkovo nám. 4	Prostějov	79601
403	Prodejna Olomouc	Rooseveltova 12	Olomouc	77900
404	Prodejna Protivanov	Protivanov 224	Protivanov	79848
405	Prodejna Morkovice	17. listopadu 120	Morkovice	76833
406	Prodejna Zdounky	Kroměřížská 7	Zdounky	76802
407	Prodejna Kojetín	Husova 661	Kojetín	75201
*				

Dodavatelé - Okno pro editaci a vkládání nových dodavatelů. Dodávky zboží na prodejny z vlastního výrobního závodu se řídí jinými pravidly a vyhodnocují se na zvláštním účtu. Z toho důvodu je nutné jednoho nebo více dodavatelů za tímto účelem označit. To provedeme u příslušného dodavatele zaškrtnutím políčka ve sloupci **Centrála**.



Dodavatel	Název	Adresa	Obec	PSČ	Centrála
1	Jatka Lešany	Vícov	Lešany	79802	<input checked="" type="checkbox"/>
46903631	Makovec a.s.	Nám. T. G. Masaryka 23	Prostějov	79601	<input type="checkbox"/>
12646199	Josef Filák	Homí Lideč 243	Homí Lideč	75612	<input type="checkbox"/>
12752983	Vladimír Tesař	Bílovice 3	Bílovice	79841	<input type="checkbox"/>
45308314	Tesco Stores	Višovická 1527	Praha	10000	<input type="checkbox"/>
*					<input type="checkbox"/>

Účty - Zde je možné změnit čísla účtů, které se používají ve výsledných sestavách a exportu do účetnictví.

Číslo účtu	Název účtu
132200	Zboží
501190	Spotřeba materiálu
504000	Spotřeba zboží
604100	Tržby prodejen
621000	Aktivace
582950	Manko/přebytek
321000	Zaplacené faktury v hotovosti
343015	DPH snížená
343021	DPH základní
395000	Převodový účet
211300	Pokladna

Záloha databáze - Tato volba zobrazí dialogové okno pro výběr adresáře, do kterého je možné uložit zálohu databáze. Soubor se zálohou má jméno Prodejny.sdf. Obnovit zpět databázi je možné kdykoliv pomocí funkce Obnova databáze.

Obnova databáze - Tato volba zobrazí dialogové okno pro výběr souboru, ze kterého je možné obnovit stav databáze do stavu, ve kterém byla při vytvoření zálohy pomocí funkce Záloha databáze.

Pozor! Tato funkce smaže všechna data, která byla přidána nebo změněna od poslední zálohy databáze.

Předvolby - V předvolbách je možné změnit aktuální výše sazeb DPH a výše rabatu pro zboží z centrály a od externích dodavatelů. Tento rabat se používá pro výpočet předpokládaných tržeb a pro vyhodnocení celkové bilance hospodaření prodejny.

Předvolby

DPH

Základní sazba: 21 %

Snižena sazba: 15 %

Rabat

Výše rabatu: 21 %

Rabat Centrály: 16 %

OK Zrušit

3.4.2. Provozovny

Zde najdeme úlohy pro zadávání denních i měsíčních výsledků jednotlivých provozoven.

Výkazy - Tato volba slouží k zadávání denních výsledků z jednotlivých provozoven. Zde se evidují všechny přijaté doklady na prodejnu v částkách včetně DPH.

Výkaz provozovny

Provozovna

ID Provozovny: 400 Název: Prodejna Lešany Období: květen 2013

Výkazy

	Datum	Dodavatel	Jméno	Doklad	Nákup	Režie	DPH %
▶	1.5.2013	1	Jatka Lešany		1000	100	15
	2.5.2013	46903631	Makovec a.s.	F20232123	5600		15
	3.5.2013	12646199	Josef Filák	F12332123	4500		21
	5.5.2013	45308314	Tesco Stores			450	21
	10.5.2013	1	Jatka Lešany		8000		15
*							

Počet záznamů : 5 Nákup celkem : 19100 Režie celkem : 550

- **Význam jednotlivý polí**

- ID Provozovny - Slouží k zadání ID prodejny, pro kterou budeme zadávat denní výkazy. Je možné vybírat pouze ze záznamů, které jsou nadefinovány v číselníku provozoven. Po výběru konkrétní prodejny se v tabulce zobrazí pouze záznamy odpovídající této prodejně.
- Období - Je-li zaškrťovací políčko označené, je možné pomocí šipek nahoru a dolů vybrat měsíc a rok odpovídající období, pro které chceme zobrazit záznamy v tabulce. Není-li toto políčko označené, zobrazí se k vybrané prodejně všechny záznamy bez ohledu na jejich datum.

- **Tabulka výkazy**

- Datum - Datum příjmu zboží / režijního materiálu.
- Dodavatel - Identifikační číslo dodavatele zboží. Je možné zadat pouze dodavatele definované v číselníku dodavatelů.
- Jméno - Jméno dodavatele. Toto pole nelze měnit, jméno dodavatele je možné změnit v číselníku dodavatelů.
- Doklad - Číslo faktury nebo dodacího listu, který byl vystaven dodavatelem.
- Nákup - Hodnota nakoupeného zboží včetně DPH, které je určeno k dalšímu prodeji.
- Režie - Hodnota režijního materiálu včetně DPH, který není určen k dalšímu prodeji.
- DPH - Výše DPH v procentech. Lze zadat pouze platné hodnoty výše sazeb definované v předvolbách.

Tržby - Tato volba slouží k zadávání denních tržeb a odvodů do banky.

- **Význam jednotlivý polí**

- Období - Je-li toto zaškrťovací políčko označené, je možné pomocí šipek nahoru a dolů vybrat měsíc a rok odpovídající období, pro které chceme zobrazit záznamy v tabulce. Není-li toto políčko označené, zobrazí se všechny záznamy bez ohledu na jejich datum.

- **Tabulka Tržby a odvody**

- Provozovna - ID prodejny, pro kterou budeme zadávat tržbu nebo odvedenou hotovost. Je možné vybírat pouze ze záznamů, které jsou nadefinovány v číselníku provozoven.
- Název - Název prodejny. Toto pole nelze měnit, název prodejny je možné změnit v číselníku provozoven.

Tržby a odvody

Období: květen 2013

	Provozovna	Název	Datum	Tržba	Odvod
	400	Prodejna Lešany	1.5.2013	100	100
	401	Prodejna Mostkovice	1.5.2013		130
	400	Prodejna Lešany	2.5.2013	150	
	400	Prodejna Lešany	3.5.2013	120	
▶*	400				

Počet záznamů : 4 Tržby celkem : 370 Odvody celkem : 230

- Datum - Datum odvedené hotovosti nebo tržby.
- Tržba - Hodnota denní tržby na konci směny.
- Odvod - Hodnota odvedené hotovosti do banky.

Inventury - V této tabulce se zadávají výsledky měsíčních inventur. Inventura by měla být provedena vždy na konci měsíce. V případě potřeby je však možné zadat a spočítat inventuru k libovolnému datu uprostřed měsíčního období.

Inventury

	Provozovna	Jméno	Datum	Pokladna	Zboží ve snížené sazbě	Zboží v základní sazbě
	400	Prodejna Lešany	30.4.2013	1000	5000	2000
	400	Prodejna Lešany	2.5.2013	1100	4000	3000
	400	Prodejna Lešany	12.5.2013	900	3000	1000
	400	Prodejna Lešany	30.3.2013	2000	4000	1500
	401	Prodejna Mostkovice	30.3.2013	2000	2000	3000
	401	Prodejna Mostkovice	30.4.2013	1000	2000	1500
	401	Prodejna Mostkovice	15.5.2013	1500	3000	1600
▶*	400					

- Význam jednotlivých polí

- **Provozovna** - ID prodejny, pro kterou budeme zadávat inventuru. Je možné vybírat pouze ze záznamů, které jsou nadefinovány v číselníku provozoven.
- **Název** - Název prodejny. Toto pole nelze měnit, název prodejny je možné změnit v číselníku provozoven.
- **Datum** - Datum provedené inventury.
- **Pokladna** - Hodnota hotovosti v pokladně.
- **Zboží ve snížené sazbě** - Hodnota zboží ve snížené sazbě DPH včetně této daně.
- **Zboží v základní sazbě** - Hodnota zboží v základní sazbě DPH včetně této daně.

3.5. Výstupy

Pod touto položkou najdeme výstupní a exportní sestavy.

Výkaz hospodaření - Tato volba slouží k zobrazení a tisku výkazů hospodaření každé prodejny, nebo souhrnné sestavy za všechny prodejny. Výkaz hospodaření lze zobrazit pouze za období, ve kterém jsou pro danou prodejnu vyplněné příslušné údaje a zároveň existuje záznam inventury z předchozího měsíce, aby bylo možné vypočítat počáteční stavy.

Výsledek inventury - Touto volbou můžeme zobrazit výsledky inventur jednotlivých prodejen, nebo souhrnného výsledku za všechny prodejny. Výsledky lze zobrazit pouze za období, ve kterém jsou pro danou prodejnu zadané příslušné údaje a zároveň existuje záznam inventury z předchozího měsíce, aby bylo možné vypočítat počáteční stavy.

Export do účetnictví - Touto volbou zobrazíme dialog s výběrem cesty a jména pro uložení textového souboru se souhrnnými výsledky hospodaření všech zpracovávaných prodejen společně s příslušnými účty. Tento soubor je určen pro import do účetního systému.

Výběr provozovny

Provozovna 400 Prodejna Lešany Období duben 2013

	Počáteční stav	Konečný stav	Rozdil
► Pořizovací ceny	2 351,44 Kč	2 990,95 Kč	639,51 Kč
Prodejní ceny	5 500,00 Kč	7 000,00 Kč	1 500,00 Kč
Pokladna	2 000,00 Kč	1 000,00 Kč	-1 000,00 Kč

Závěr

Cílem této bakalářské práce bylo vytvořit aplikaci, která bude přizpůsobena potřebám společnosti ZD Vícov a co možná nejvíce minimalizuje úkony spojené s potřebou evidovat pohyb zboží a tržeb jednotlivých provozoven patřících této společnosti, ze kterých následně provádí ekonomické vyhodnocení hospodaření každé této provozovny. Výstupem programu je exportní soubor konsolidovaných údajů do účetního systému. Aplikace je vytvořena tak, aby byla snadno spravovatelná a zároveň sledovala legislativní i vnitropodnikové změny.

Reference

- [1] SHARP, J.: *Microsoft Visual C# 2005 Krok za krokem*
Computer Press, a.s., 2006, ISBN 80-251-1156-3
- [2] Virius, Miroslav.: *C# Hotová řešení*
Computer Press, a.s., 2006, ISBN 80-251-1084-2
- [3] Lerman, Julie. *Programming Entity Framework*.
On-line publikace dostupná na <http://learnentityframework.com>
- [4] Microsoft. *Webový portál MSDN*.
On-line publikace dostupná na <http://msdn.microsoft.com>
- [5] Diskuzní fórum *StackOverflow*
On-line publikace dostupná na <http://stackoverflow.com>

4. Obsah přiloženého CD

`bin/`

Instalační soubor `setup.exe` aplikace PRODEJNY a kompletní podoba aplikace ve spustitelné formě přímo z CD. Vyžaduje instalaci .NET Framework 4.0 a SQL Ssrver 4.0 CE, které jsou součástí instalačního souboru.

`doc/`

Dokumentace bakalářské práce ve formátu PDF a ZIP archiv zdrojových souborů k dokumentaci.

`src/`

Zdrojové kódy aplikace pro Visual Studio 2012

`install/`

Instalační soubor aplikace a samostatné instalační soubory .NET Framework 4.0 a SQL Server 4.0 CE x86 a 64bit.

`readme.txt`

Instrukce pro instalaci a spuštění programu PRODEJNY, včetně požadavků pro jeho provoz.