

**Jihočeská univerzita v Českých Budějovicích**

Pedagogická fakulta  
Katedra fyziky

Vývoj a modernizace softwarové aplikace DN-Portál  
pro zákazníky používající ERP systém i/2.

Bakalářská práce

Vedoucí práce: Prof. RNDr. Petr Špatenka, CSc.

Autor: Marek Víta

## **Anotace**

Vývoj a modernizace softwarové aplikace DN-Portál pro zákazníky používající ERP systém i/2.

Cílem této bakalářské práce je vytvořit novou verzi objednávkové aplikace DN-Portál pro zákazníky používající ERP systém i/2. Nová aplikace nabízí koncovým zákazníkům komfortnější uživatelské rozhraní, rychlejší aktualizace, umožní výpočet stupnicových cen a individuálních slev ke každému produktu. Součástí nového řešení je varianta pro obchodní zástupce, která umožní realizaci objednávek obchodním zástupcem za koncového zákazníka. K realizaci aplikace byl použit programovací jazyk C# s využitím technologie Windows Presentation Foundation, knihovny Microsoft .NET Framework 4.0 a databáze SQLite. Přenos dat mezi klientskou aplikací a serverem je realizován prostřednictvím technologie SOAP.

## **Annotation**

Development and Modernization of the DN-Portal Software Application for Customers Using the System i/2 ERP.

The goal of my Bachelor's Thesis is to create a new version of the DN-Portal orders application for customers using the i/2 ERP system. The new application provides a more comfortable user interface and faster updates, and supports calculation of scale prices as well as individual product discounts. A version for sales representatives is included, which enables representatives to place orders on behalf of the end customer. The application was programmed in C#, utilising Windows Presentation Foundation technology, Microsoft .NET Framework 4.0 and an SQLite database. SOAP technology is used to transmit data between the client application and server.

## **Klíčová slova**

ERP i/2, DN-Portál, objednávková aplikace, .NET, WPF, SOAP, WSDL, XML

## **Prohlášení**

Prohlašuji, že jsem bakalářskou práci na téma *Vývoj a modernizace softwarové aplikace DN-Portál pro zákazníky používající ERP systém i/2* vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v úpravě vzniklé vypuštěním vyznačených částí archivovaných pedagogickou fakultou JU elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce.

Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledky obhajoby kvalifikační práce.

Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Českých Budějovicích.....

.....  
Marek Víta

## **Poděkování**

Rád bych poděkoval vedoucímu mé bakalářské práce Prof. RNDr. Petru Špatenkovi, Csc., a konzultantovi Ing. Ursi Gyselovi, za poskytnutí cenných rad a odborné vedení během mé práce.

# Obsah

<b>ANOTACE</b>	<b>2</b>
<b>ANNOTATION</b>	<b>2</b>
<b>KLÍČOVÁ SLOVA</b>	<b>2</b>
<b>PROHLÁŠENÍ</b>	<b>3</b>
<b>PODĚKOVÁNÍ</b>	<b>4</b>
<b>OBSAH</b>	<b>5</b>
<b>1. ÚVOD</b>	<b>7</b>
<b>2. PROJEKT A PROJEKTOVÉ ŘÍZENÍ</b>	<b>8</b>
2.1. Analýza	8
2.2. Návrh	9
2.3. Konstrukce	9
2.4. Testování	9
2.5. Instalace	9
<b>3. ANALÝZA SOUČASNÉHO ŘEŠENÍ</b>	<b>10</b>
3.1. Použité technologie	10
3.1.1. .Net Framework 2.0	10
3.1.2. Databáze Microsoft Access	10
3.2. Popis rozhraní	11
3.2.1. Role zákazník	11
3.2.2. Role obchodní zástupce	16
3.3. Komunikace	18
3.3.1. Definice zpráv	18
3.3.2. Aktualizace	20
3.4. Nedostatky současné verze	21
3.4.1. Aktualizace obrázků	21
3.4.2. Databáze	21
3.4.3. Verze pro obchodní zástupce	21
3.4.4. Izolace	21
<b>4. POUŽITÉ TECHNOLOGIE PRO VÝVOJ NOVÉ APLIKACE</b>	<b>22</b>
4.1. Microsoft. NET Framework	22
4.1.1. Jazyk C# 4.0	23
4.1.2. WPF	23
4.2. Visual Studio 2010	24
4.3. Databáze SQLite	25
4.4. Webové služby	27
4.4.1. XML	28
4.4.2. SOAP	29
4.4.3. WSDL	30
<b>5. NÁVRH APLIKACE</b>	<b>32</b>
5.1. Návrh grafické podoby aplikace	32
5.2. Uživatele systému	32
5.2.1. Zákazník	32
5.2.2. Obchodní zástupce	32
5.3. Návrh databáze	33
5.3.1. Přehled tabulek	34
5.4. Návrh komunikace	35
5.4.1. Rozhraní služby	36
5.5. Návrh aktualizace	38
5.5.1. Aktualizace databáze	38
5.5.2. Aktualizace programu	38

5.5.3.	Aktualizace obrázků	38
<b>6.</b>	<b>KONSTRUKCE APLIKACE</b>	<b>39</b>
6.1.	Design rozhraní – použití stylů	39
6.2.	Popis rozhraní	39
6.2.1.	Role zákazník	39
6.2.2.	Role obchodní zástupce	40
6.2.3.	Hlavní okno aplikace	41
6.2.4.	Komponenty	46
6.3.	Komunikace aplikace	51
6.4.	Aktualizace	52
6.4.1.	Aktualizace databáze	52
6.4.2.	Aktualizace programu	52
6.4.3.	Aktualizace obrázků	52
6.5.	Kontroly a výpočty	53
6.5.1.	Kontrola objednávaného množství	53
6.5.2.	Cenový model a výpočet ceny	55
6.5.3.	Aplikace slev	56
<b>7.</b>	<b>TESTOVÁNÍ APLIKACE</b>	<b>58</b>
7.1.	Odhalená a následně odstraněná slabá místa v návrhu	58
<b>8.</b>	<b>ZÁVĚR</b>	<b>59</b>
<b>9.</b>	<b>LITERATURA</b>	<b>62</b>
<b>10.</b>	<b>SEZNAM OBRÁZKŮ</b>	<b>64</b>
<b>11.</b>	<b>SEZNAM ZDROJOVÝCH KÓDŮ</b>	<b>65</b>
<b>12.</b>	<b>PŘÍLOHY</b>	<b>66</b>
	<b>PŘÍLOHA A – VÝPOČET KATALOGOVÉ CENY</b>	<b>67</b>
	<b>PŘÍLOHA B – POUŽITÍ KONVERTORU</b>	<b>69</b>

## 1. Úvod

Internet je v dnešní době stále častěji využíván v oblasti elektronického obchodování. E-shopy maloobchodů, které jsou cílené na koncové spotřebitele, se objevují stále častěji. Velkoobchody v tomto směru nezůstávají pozadu, snaží se však svým zákazníkům (maloobchodům) nabídnout i něco navíc, než jen přehledné e-shopy. Tím navíc je objednávková aplikace.

Objednávková aplikace je samostatný spustitelný program, který si koncový zákazník nainstaluje do svého osobního počítače. Svými možnostmi, jednoduchostí obsluhy a rychlostí předčí klasické e-shopy. Aplikace je snadno použitelná, takže pomocí ní mohou objednávat i uživatelé se základními znalostmi ovládnání počítače. Aplikace je navržena tak, aby vytváření objednávek bylo maximálně rychlé a pohodlné. Velkou výhodou je, že zákazník nemusí být připojen k internetu po celou dobu vytváření objednávky, či procházení katalogu produktů. K internetu se musí připojit pouze pro odeslání připravené objednávky, či pro stažení aktualizace dat a programu.

Cílem této bakalářské práce je připravit novou funkční verzi objednávkové aplikace, která bude základem pro další vývoj dle požadavků zákazníka. Současná verze DN-Portál je software vyvinutý firmou DATA-NORMS s.r.o.<sup>1</sup> jako doplňková aplikace ERP<sup>2</sup> systému i/2, který je vyvíjen a dodáván švýcarskou společností Polynorm<sup>3</sup> Software AG se sídlem v Glattbrugg/Zurich. Současná aplikace DN-Portál v úpravě pro velkoobchod ZC s.r.o.<sup>4</sup> Jindřichův Hradec přinesla významný nárůst objednávek a rychlost jejich zpracování a to nejen díky plné integraci do ERP i/2, ale i díky odstranění chybovosti způsobené lidskou obsluhou při přepisování došlých objednávek.

---

<sup>1</sup> <http://www.data-norms.cz/>

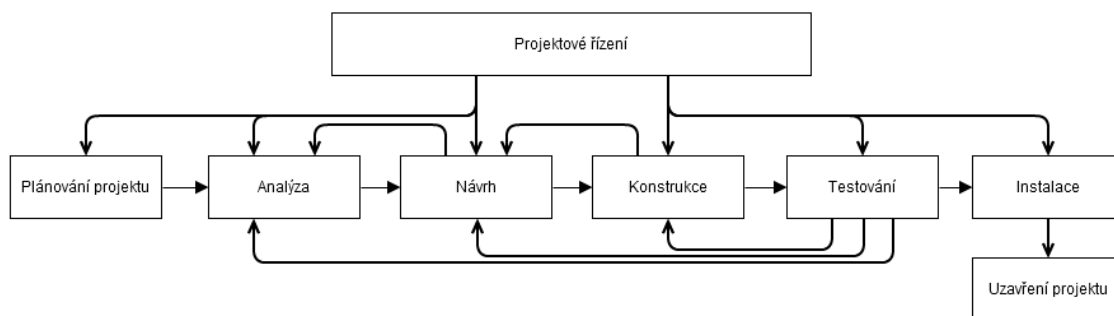
<sup>2</sup> Enterprise Resource Planning je informační systém, který integruje a automatizuje velké množství procesů souvisejících s produkčními činnostmi podniku. Typicky se jedná o výrobu, logistiku, distribuci, správu majetku, prodej, fakturaci, a účetnictví.

<sup>3</sup> <http://www.polynorm.ch/>

<sup>4</sup> <http://www.zc.cz>

## 2. Projekt a projektové řízení

Projekt je činnost, která má jasně daný cíl, začátek a konec. Cílem této práce je vytvoření nové funkční verze aplikace DN-Portál s využitím nových technologií. Projektové řízení je proces, ve kterém jednotlivci nebo organizace využívají své zdroje k realizaci projektů. Tato kapitola čerpá z [3] a [4].



Obrázek 1: Schéma projektového řízení

### 2.1. Analýza

Samotný projekt začíná sběrem požadavků. Cílem analýzy je co nejpodrobnější zjištění informací o požadavcích na budoucí projekt. Čím víc dat získáme, tím lepší analýzu budeme schopni udělat. Požadavky na modernizaci DN-Portálu byly zjištěny v rámci obchodních jednání firmy DATA-NORMS a jsou součástí zadání této bakalářské práce. Ostatní formální požadavky vyplývají z vlastností a funkčnosti současné verze, jejíž podrobnější analýza a popis je proveden v kapitole: 3.

#### Požadavky na novou verzi aplikace

##### Základní požadavky:

- Použít novější technologie – zajistí lepší konkurenceschopnost;
- Při vývoji aplikace zachovat jednoduchost a intuitivnost ovládání programu;
- Snížit zátěž ERP i/2 při generování číselníků - přesun výpočtu individuálních cen z ERP na DN-Portál;

##### Požadavky na aplikaci

- Připravit aplikaci na možnost rozšíření mimo hranice ČR - podpora více jazyků;
- Rozložení hlavních ovládacích prvků a komponent podobné stávající verzi;
- Verze pro obchodní zástupce;
- Automatická aktualizace obrázků s kontrolou platnosti;

##### Požadavky na komunikaci

- Umožnit propojení DN-Portál i do jiných ERP systémů;
- Podpora otevřených standardů - změna druhu komunikace mezi komponenty;
- Používat standardní protokoly a porty;

##### Požadavky na databázi

- Základní zabezpečení pro práci s databází;
- Jazykové mutace hlavních tabulek;



- Katalogové ceny (základní a stupnicové);
- Akční ceny (základní a stupnicové);
- Individuální ceny (základní a stupnicové);
- Slevy na zboží (dle skupiny zboží, dle zákazníka);
- Adresy zákazníka s možností přidání nové adresy;
- Objednávka jen jedna na zákazníka;
- Verze pro obchodní zástupce (přihlášení a správa vlastních zákazníků);

### **Požadavky na použité technologie**

- Operační systém Windows;
- Použití pouze standardních součástí Windows či volně šiřitelných knihoven;
- Použití volně šiřitelnou „lehkou“ databázi. Ověřit více možností a vybrat nejvhodnější;

### **Předpokládaná investice do vývoje**

Protože je projekt realizován v rámci BP, není nutné sledovat náklady. V případě komerčního projektu se náklady na vývoj této aplikace pohybují v rozsahu odhadovaném na 100-150 člověko dní. Při vývoji je obvyklé pravidelně porovnávat stav projektu s rozpočtem a s termínem. Při odchylkách je potřeba eskalovat problém a nechat schválit další postup (změny termínu, rozšíření rozpočtu, navýšení zdrojů).

### **2.2. Návrh**

Návrh aplikace vychází ze základních a rozšířených požadavků. Mohu ji rozdělit do dvou základních kroků. Tím prvním je výběr technologie, která bude použita při vývoji. Druhým krokem je návrh samotné aplikace, návrh databáze a definice základní komunikace aplikace.

Zvolené technologie použité pro vývoj nové aplikace jsou upřesněny v kapitole: 4.

Vlastnímu návrhu nové aplikace se budeme podrobněji věnovat v kapitole: 5.

### **2.3. Konstrukce**

Konstrukce aplikace vychází z požadavků návrhu. Grafické podobě, uživatelskému rozhraní a jednotlivým komponentám aplikace se budeme více věnovat v kapitole:6.

### **2.4. Testování**

Testování aplikace je nedílnou součástí vývoje. Po dokončení jednotlivých komponent či funkcí vždy následuje ověření, zda výsledek odpovídá očekávanému výstupu a zda splňuje všechny stanovené požadavky. Více se testování budeme věnovat v samostatné kapitole: 7.

### **2.5. Instalace**

Je konečnou fází projektu, popisuje kroky a procesy, které vedou k zprovoznění aplikace na počítači koncového zákazníka. Tuto fázi si dovoluji vypustit, a to z důvodu, že výstupem bakalářské práce je verze programu, která není konečná, ale je základem pro její další vývoj.

### **3. Analýza současného řešení**

#### **3.1. Použité technologie**

Vývoj současné verze DN-Portálu byl započat roku 2007 a tak technologie použité při sestavování aplikace korespondují s tehdejšími možnostmi a především s požadavky klienta.

##### **3.1.1. .Net Framework 2.0**

Je prostředí potřebné pro běh aplikací a nabízející jak spouštěcí rozhraní, tak potřebné knihovny. Je spojena s vývojovým prostředím Visual Studio 2005.

##### **3.1.2. Databáze Microsoft Access**

Databáze je uložena v jednom souboru \*.mdb s omezeným přístupem. Obsahuje informace o aktivním zákazníkovi, jako jsou adresy a uživatelé, dále číselník zboží, skupin a dodatečné informace o verzi databáze, aktualizacích, a přístupových právech.

##### **3.1.2.1. E-R model databáze**

Následující pasáž o rozsahu 2 stran obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

## **3.2. Popis rozhraní**

### **3.2.1. Role zákazník**

Zákazník je koncový uživatel systému DN-Portál, který má přístup do aplikace po zadání přihlašovacích údajů.

#### **3.2.1.1. Přihlášení zákazníka**

Po spuštění programu je zobrazena obrazovka s požadavkem na zadání přihlašovacích údajů. Pro úspěšné spuštění aplikace je nutné zadat správnou kombinaci uživatelského jména a hesla. Systém umožňuje několik způsobu přihlášení.

##### **První přihlášení - Inicializace zákazníka**

Po instalaci programu u zákazníka nejsou v databázi obsažené žádné údaje, které by umožnily přihlášení zákazníka. Pro první přihlášení musí být klientský počítač připojen do sítě internet a musí být dostupný cílový server. Identifikace uživatele pro první přihlášení je specifická, skládá se z čísla zákazníka, uživatelského jména a vygenerovaného uživatelského hesla. Například pro zákazníka s číslem 3333 a uživatelským jménem MVita je přihlašovacím výrazem „3333MVita“.



Obrázek 2: První přihlášení do aplikace - Zákazník

Po vyplnění požadovaných údajů je uskutečněn dotaz na webový server a je-li kombinace přihlašovacích údajů úspěšně ověřena, je uživatel požádán o změnu výchozího hesla. Po změně jsou systému navráceny základní číselníky identifikující zákazníka (EXKUNSTA, EXANSPRE, EXADRESS). Ty jsou následně importovány do databáze a je provedeno spuštění hlavního okna programu.

#### **Každé další přihlášení zákazníka**

Jakékoli další přihlášení je již možné jen ve formátu tzv. „krátkého“ uživatelského jména a zvoleného hesla. Po ověření přihlašovacích údajů v lokální databázi je v případě správné kombinace provedeno spuštění hlavního okna programu. Pokud přihlašovací údaje nesouhlasí, je uživatel vyzván k opětovnému zadání.

K přihlášení do programu není nutné být připojen k internetu, veškeré potřebné informace jsou dostupné v lokální databázi.

#### **3.2.1.2. Hlavní okno aplikace**

Aplikace je koncipována tak, aby všechny informace byly co nejsnadněji dostupné z hlavního okna bez nutnosti procházet množství dalších oken s dodatečnými informacemi.

The screenshot displays the main interface of the DN-Portal application. At the top, there is a header with the company logo 'ZC s.r.o. - ZC Portál', contact information (Hotline: +420 384 370 900, Příští závoz: 10.11.2010), and a navigation bar with buttons for 'Firma', 'Akce', 'Katalog', 'Objednávka', 'MO ceny', and 'Služby'. The main content area is divided into several sections:

- Search and Navigation:** A search bar with the text 'Vyhledávání' and a 'Hledat' button. Below it, a 'Katalog zboží' section shows a list of products with columns for 'Kat.číslo', 'Zboží', 'Balení', 'Katalogová cena', 'Nákupní cena', 'Obj.pocet', and 'Celková cena'.
- Product List:** A table listing various products such as 'Cocochips - kokosové kousky 500 g', 'Cocopress - kokosové vlákno 650 g', and various types of mulch ('Keramzít', 'Kůra mulčovací', 'Kůra piniová').
- Product Detail View:** A detailed view for 'Cocochips - kokosové kousky 500 g' (Kat.číslo: 180098). It includes an image of the product, a description ('Přírodní, organický, kokosový substrát...'), and pricing information: 'ZC Katalogová cena bez DPH: 32,00 Kč', 'Moje nákupní cena bez DPH: 26,24 Kč', and 'Cena celkem: 0,00 Kč'.
- Statistika:** A small table showing monthly statistics for the current year.

Obrázek 3: Hlavní formulář aplikace DN-Portál (v modifikaci pro Velkoobchod ZC s.r.o.)

Hlavní okno je rozděleno do třech základních sekcí:

### Informační sekce

Informační sekce obsahuje informace o příštím závozu zboží k zákazníkovi, aktuální ceně zboží vložené do objednávky a informaci o aktuálním a celkovém počtu bendit bodů.

### Hlavní sekce

Hlavní sekce je dělena na jednotlivé záložky, jejich popisu se budeme věnovat v následujících bodech.

#### Firma

Obsahuje informace o aktivním zákazníkovi a právě přihlášeném uživateli.

Informace o firmě (Název, IČO, DIČ), dodací a fakturační adresa, aktivní uživatel (Jméno, příjmení, mail). Je zde umožněno změnit zásilací emailovou adresu. Ostatní změny nejsou v systému povoleny. Změna je možná pouze při komunikaci s velkoobchodem přímo v ERP systému i/2.

#### Akce

Obsahuje základní přehled všech aktuálních produktů, které jsou v systému vedené jako akční položky; tzn., že při nákupu zboží v akci, při dodržení minimálního odebraného množství zákazník ušetří oproti své standardní individuální ceně. Záložka "akce" je dělena do třech sekcí:

- Vyhledávání, ušetříte

- Přehled produktů v akci
- Detailní informace o právě vybraném produktu

### **Katalog**

Tabulkový výpis zboží dle vybrané kategorie zboží. Záložka "katalog" je dělena do čtyř sekcí:

- Kategorie zboží
- Vyhledávání
- Přehled produktů dle zvolené kategorie
- Detailní informace o právě vybraném produktu

### **Objednávka**

Tabulkový výpis položek vložených do objednávky s podrobnějším rozpisem konečných cen. Záložka "objednávka" je dělena do čtyř sekcí:

- Přehled objednaných produktů
- Rychlá objednávka
- Detailní informace o právě vybraném produktu
- Odeslání objednávky, Export, Import

### **Maloobchodní ceny**

Tabulkový výpis zboží dle vybrané kategorie zboží. Zobrazení je z velké části shodné se zobrazením katalogu. Odlišností je, že se zbožím se v této sekci pracuje jen v režimu úpravy maloobchodních cen. Záložka "maloobchodní ceny" je dělena do čtyř sekcí:

- Kategorie zboží
- Vyhledávání
- Přehled produktů dle zvolené kategorie
- Detailní informace o právě vybraném produktu s možností zadání maloobchodní ceny, marže, či koeficientu

### **Webové služby**

Je komponenta zobrazující webovou stránku s proměnným obsahem závislým na přihlášeném zákazníkovi a zvolené sekci. Uvedu jen základní strany/funkce které se mohou zobrazit přihlášenému uživateli bez popisu dalších funkcí.

Zboží v rezervaci, přehled zaplacených faktur, maloobchodní faktury ke stažení a uživatelská příručka.

### **Status**

Informační lišta, ve které jsou zobrazovány informace o aktivním spojení s cílovým serverem a informace o průběhu aktualizace.

### 3.2.1.3. Hlavní komponenty

#### Kategorie zboží

Komponenta s výpisem hlavních kategorií a jejich podkategorií. Kategorie jsou přehledně zobrazeny v tabulkách. Aktuální vybraná kategorie či podkategorie je indikována podbarvením. V závislosti na výběru kategorie, jsou zobrazeny produkty v komponentě katalog.

#### Vyhledávání

Komponenta, která filtruje dle zadaného hledaného výrazu seznam produktů na aktuální vybrané záložce (akce, katalog, MOCeny<sup>5</sup>), je fulltextově vyhledáváno podle čísla zboží, jména a popisu zboží. Hledání je automaticky započato při změně hledaného výrazu, či při stisku tlačítka hledej.

Na záložce "katalog" a "MOCeny" je zobrazen checkbox (na záložce "akce" je skryt). Slouží k výběru vyhledávání:

- false – vyhledáváme jen nad vybranou skupinou a podskupinou zboží
- true – vyhledáváme nad celým katalogem zboží bez ohledu na aktuální vybranou kategorii či podkategorii

#### Katalog zboží

Je přehledný tabulkový výpis produktů dle kritérií, kterými mohou být např.: vybraná kategorie zboží, vyhledání, objednané produkty, akční produkty, atd.

Základními zobrazenými informacemi jsou: číslo zboží, název zboží, balení, katalogová cena, nákupní cena, objednaný počet, cena celkem. Produkty zobrazené v této komponentě jsou řazeny prioritně dle třídícího znaku (určuje se v ERP i/2). Zákazník má však možnost řazení přeskupit např. dle čísla či názvu zboží, a to jak vzestupně tak sestupně. Ostatní zobrazené sloupce není možné řadit.

Pohyb v katalogu zboží je možný standardními klávesami, objednání zboží je možné několika způsoby.

- Zadáním požadovaného množství kusů na klávesnici.
- Stiskem kláves +, -
- Použitím myši kliknutím na +, -

Při snížení nebo zvýšení objednaného počtu dojde k přepočtu ceny dané položky a ceny celkem. O tom že zboží je vloženo do košíku informuje podbarvení zboží a nenulová cena celkem.

#### Detailní informace o zboží

Obrázek zboží je zobrazen v případě, že existuje miniatura v lokálním adresáři obrázků k aktuálně vybranému produktu. Detailnější obrázek je možné zobrazit kliknutím na detail obrázku, či stisknutím mezerníku.

Detailnější informace o právě vybraném produktu jsou zobrazeny ve středu komponenty a obsahují např.: popis produktu, informace o cenách včetně zobrazení aktuální maloobchodní ceny, marže a koeficientu. Poslední důležitou informací je statistika produktu. Informuje zákazníka o množstvích, které u dané položky objednával v předchozích měsících.

#### Rychlá objednávka

Je komponenta, která umožňuje při znalosti čísla zboží rychle vložit toto zboží do objednávky. Před vložením je ověřeno, zda zadané číslo zboží existuje.

---

<sup>5</sup> Maloobchodní ceny

### MOCeny - Maloobchodní ceny

Komponenta, která umožňuje upravovat vlastní maloobchodní ceny. Při změně MOCeny, marže či koeficientu, je automaticky dopočteno zbývající. Např. měním-li MOCenu, je při změně automaticky dopočten koeficient a marže.

#### 3.2.2. Role obchodní zástupce

Obchodní zástupce je osoba, která má v obchodním procesu na starosti koncového zákazníka, V aplikaci DN-Portál má rozšířený přístup do rozhraní, ve kterém může přidat či upravit uživatele, kteří smí pracovat se systémem na lokálním počítači.

##### 3.2.2.1. Přihlášení obchodního zástupce

Přihlášení je možné jen v případě, že je dostupné připojení k internetu a slouží pouze k možnosti vytvoření nového uživatele. Manažer nemá možnost se samostatně přihlásit do systému a pracovat s daty uživatele.

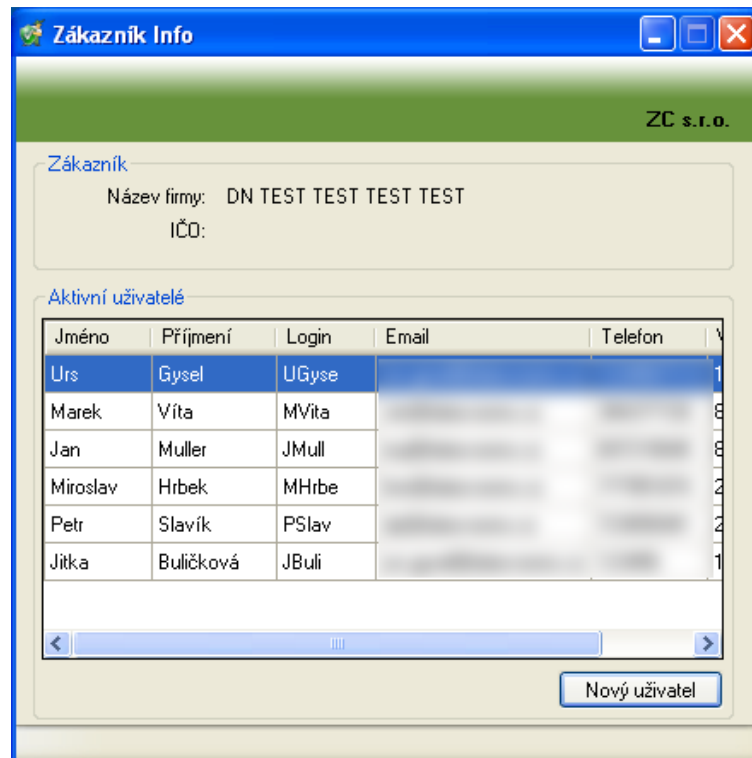
Přihlášení manažera je možné jen v případě zadání „dlouhého“ přihlašovacího jména rozšířeného o identifikaci manažera. Pro zákazníka 3333 a přihlašovací login manage01 je přihlašovacím výrazem „3333@manage01“ a hesla. Kombinace hesla a jména je ověřována pouze přes webové rozhraní (touto kombinací se nelze do systému připojit při nedostupnosti serveru)



Obrázek 4: První přihlášení do aplikace – Obchodní zástupce

Po vyplnění požadovaných údajů je uskutečněn dotaz na webový server a je-li kombinace přihlašovacích údajů úspěšně ověřena, zobrazen výpis aktuálních aktivních uživatelských jmen registrovaných pod zvolenou firmou.





Obrázek 5: Aktivní uživatelé DN-Portálu

Obchodní zástupce má možnost v systému založit nového uživatele zákazníka, k tomu slouží vlastní formulář, kde se vyplní požadované údaje. Po odeslání, je nový uživatel založen v lokální databázi a zároveň na straně serveru v ERP i/2.

The screenshot shows a window titled 'Zákazník Info' with a green header bar containing 'ZC s.r.o.'. Below the header, there is a section for 'Nový uživatel' with the following form fields:

Jméno:   
Příjmení:   
Login:   
Heslo:   
 zobrazit heslo  
Telefon:   
Email:

At the bottom of the form are two buttons: 'Zpět' and 'Založit uživatele'.

Obrázek 6: Formulář založení nového uživatele.

**3.3. Komunikace**

Následující pasáž o rozsahu 3 stran obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.





**3.4. Nedostatky současné verze**

Následující pasáž o rozsahu 1/2 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

## 4. Použité technologie pro vývoj nové aplikace

V následující kapitole se budeme věnovat technologiím použitých při vývoji nové aplikace a zdůvodnění, jejich výběru.

### 4.1. Microsoft .NET Framework

Platforma .NET byla oficiálně představena firmou Microsoft v roce 2000 jako klíčový produkt, jehož rozvoj a propagace je součástí dlouhodobé strategie firmy. První dostupná verze .NET Framework 1.0 byla uvedena na trh v průběhu roku 2002. Microsoft .NET znamená novou generaci systému vývoje aplikací pro operační systémy Windows založeném na řízeném běhovém prostředí, obohaceném o sadu základních tříd. Tato kapitola čerpá z [5], [6] a [7].

Hlavními důvody vývoje platformy .NET:

- odstranění nekompatibility jednotlivých programátorských jazyků
- omezení vysoké chybovosti aplikací
- usnadnění a urychlení vývoje aplikací

Všechny tyto problémy efektivně řeší platforma .NET – a to použitím běhového prostředí, systémem assemblies<sup>6</sup>, což jsou základní stavební prvky aplikací. V současné době je platforma .NET dostupná v těchto verzích. 1.0, 1.1, 2.0, 3.0, 3.5 a 4.0. Poslední uvedená verze 4.0 je dostupná od počátku roku 2010 a je základem, na kterém bude postavena aplikace DN-Portál.

Rozhraní .NET Framework je komplexní a konzistentní programovací model společnosti Microsoft určený pro sestavování aplikací, které se vyznačují vizuálně působivým uživatelským rozhraním, bezproblémovou a zabezpečenou komunikací a schopností modelovat řadu obchodních procesů. Je spojen s vývojovým prostředím Visual Studio 2010.

Rozhraní .NET Framework je nedílnou součástí systému Windows, který podporuje vytváření a spouštění aplikací nové generace. Rozhraní .NET Framework je navrženo tak, aby splňovalo tyto cíle:

- Poskytnout konzistentní objektivně orientované programovací prostředí, kde je kód objektu uložen a spuštěn lokálně.
- Poskytnout prostředí pro zpracování kódu, které minimalizuje konflikty nasazení a správy verzí softwaru.
- Poskytnout prostředí pro zpracování kódu, které propaguje bezpečné zpracování kódu včetně kódu, který je vytvářen neznámou nebo částečně důvěryhodnou třetí stranou.
- Poskytnout prostředí pro zpracování kódu, které eliminuje výkonnostní problémy skriptovacích nebo interpretovaných prostředí.
- Činit vývojářské zkušenosti konzistentními napříč nejrůznějšími typy aplikací, jako jsou například aplikace určené pro systém Windows nebo webové aplikace.
- Vytvářet veškerou komunikaci na průmyslových standardech, aby se zajistilo to, že kód založený na rozhraní .NET Framework lze integrovat s jakýmkoliv jiným kódem.

---

<sup>6</sup> Assembly (česky sestavení) je balíček obsahující metody, pole, vlastnosti, interfaci, obrázky a mnoho dalšího [37].

#### 4.1.1. Jazyk C# 4.0

C# (C Sharp) je objektově orientovaný programovací jazyk vyvinutý firmou Microsoft zároveň s platformou .NET Framework. Microsoft založil C# na jazycích C++ a Java (a je tedy nepřímým potomkem jazyku C, ze kterého čerpá syntaxi). Tato kapitola čerpá z následujících zdrojů [8], [9] a [10].

V jazyce C# je realizováno 80% základních knihoven .NET Frameworku. I přesto, že je koncipován hlavně pro psaní řízeného kódu, na jehož užití je platforma .NET postavena, lze jej v případě potřeby využít i pro tvorbu kódu neřízeného (bloky unsafe). Použití neřízeného kódu znamená, že běhové prostředí CLR<sup>7</sup> neověřuje zda-li je napsaný kód bezpečný (například se neověřuje jinak vyžadovaná typová bezpečnost).

C# je case-sensitive<sup>8</sup>, neobsahuje a ani nepotřebuje dopřednou deklaraci, tzn., že není důležité pořadí deklarace metod. V jazyce C# neexistuje vícenásobná dědičnost, tzn., že každá třída může být potomkem pouze jedné třídy. Toto rozhodnutí bylo přijato, aby se předešlo komplikacím a přílišné složitosti, která je spojena s vícenásobnou dědičností. Třída však může implementovat libovolný počet rozhraní. Dále neexistují žádné globální proměnné a metody. Všechny funkce a metody musí být deklarovány uvnitř tříd. Náhradou za ně jsou statické metody a proměnné veřejných tříd.

Základní prvky jazyka C#:

- Třídy – základní stavební prvek při tvorbě objektově orientovaných aplikací obsahující akce (metody) a atributy
- Struktury – lze je chápat jako zjednodušené třídy, jejich užitím jsou nejčastěji popisovány vlastní datové struktury.
- Výčtové typy
- Vlastnosti – někdy označované jako chytré proměnné
- Pole a jejich „chytrá“ verze nazývaná indexery
- Zástupci – typově bezpečné ukazatele na funkce
- Události – druh zástupců sloužící ke zpracování asynchronních operací

#### 4.1.2. WPF

Předchůdcem Windows Presentation Foundation (WPF) jsou Windows Forms (WinForms). WPF ale není technologie nahrazující WinForms, jedná se spíše o další, druhý způsob psaní Windows aplikací. Oproti WinForms ale nabízí nové možnosti hlavně v oblasti grafického zpracování aplikace, které ve WinForms nejsou možné. Stále více se začíná prosazovat koncept bohatšího uživatelského rozhraní a hlavně jeho grafická podoba. Tato kapitola čerpá z následujících zdrojů [11], [12], [13], [14] a [15].

##### 4.1.2.1. Model WPF

WPF je zaměřené na uživatelsky zajímavé aplikace, díky vektorové grafice umožňuje bezetrátovou změnu velikosti prvků, podporuje manipulaci s multimédií, dovoluje animovat prvky, či snadno provázat data v aplikaci pomocí DataBinding<sup>9</sup>. Použitím stylů nebo šablon dává vývojářům a designérům možnosti měnit vzhled ovládacích prvků a objektů, případně je možné z geometrických útvarů vytvářet prvky vlastní.

<sup>7</sup> Common Language Runtime neboli společné běhové prostředí

<sup>8</sup> Case-sensitive - znamená, že významově odlišuje velká a malá písmena ve výrazech (customer a Customer jsou brány jako dva rozdílné pojmy)

<sup>9</sup> DataBinding poskytuje flexibilní mechanismus pro synchronizaci dat a uživatelského rozhraní.

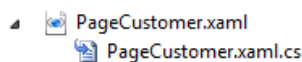
#### 4.1.2.2. XAML

WPF přináší vývojářům jednu z výrazných změn oproti WinForms, a to použití značkovacího jazyka XAML<sup>10</sup>. Jde o deklarativní programovací jazyk, který přináší možnost oddělit aplikační logiku od designu (v jednom souboru je XAML kód a v druhém funkční logika). Mezi hlavní výhody XAML patří čistota, jednoduchost a rychlost psaní kódu.

```
<Grid
    Grid.Row="1"
    Grid.Column="0">
    <Grid.ColumnDefinitions>
        <ColumnDefinition Width="Auto"/>
        <ColumnDefinition Width="*/>
    </Grid.ColumnDefinitions>
    <!-- Category -->
    <UControl:UCCategory
        Grid.Column="0"
        MinWidth="223"
        x:Name="_Category"
        DN_CategorieChanged="_Category_CategorieChanged"
        Focusable="False"/>
    <!-- Catalogue -->
    <DockPanel
        Grid.Column="1">
        <!-- Search -->
        <UControl:UCSearch
            DockPanel.Dock="Top"
            x:Name="_Search"
            Search=""
            SearchChanged="_Search_SearchChanged"
            SearchTypChanged="_Search_SearchTypChanged"
            SearchView="Katalog"
            Margin="2,0,0,0"
            Focusable="False"/>
        <UControl:UCCatalogue
            Grid.Row="0"
            Margin="2,2,0,0"
            x:Name="_Catalogue"
            SelectedRowChanged="_Catalogue_SelectedRowChanged"
            KatalogView="Katalog"
            Focusable="False"/>
    </DockPanel>
</Grid>
```

Zdrojový kód 1: XAML zápis

WPF se skládá ze dvou souborů: první s příponou ".cs", ve kterém je programová část aplikace a druhý s příponou ".xaml", ve kterém je definováno uživatelské rozhraní.



Obrázek 7: Struktura WPF – oddělení programové a grafické části kódu

Ve větších projektech tak díky tomuto přístupu již není nutné, aby se programátor zabýval vzhledem uživatelského rozhraní, to je přenechána grafikovi.

## 4.2. Visual Studio 2010

Microsoft Visual Studio 2010 je vývojové prostředí (IDE – Integrated Development Environment) od firmy Microsoft, které zjednodušuje tvorbu, ladění a nasazování aplikací na platformách Microsoft Windows, Windows Mobile, Windows CE, .NET, .NET Compact Framework a Microsoft Silverlight. Kapitola čerpá z [16] a [17].

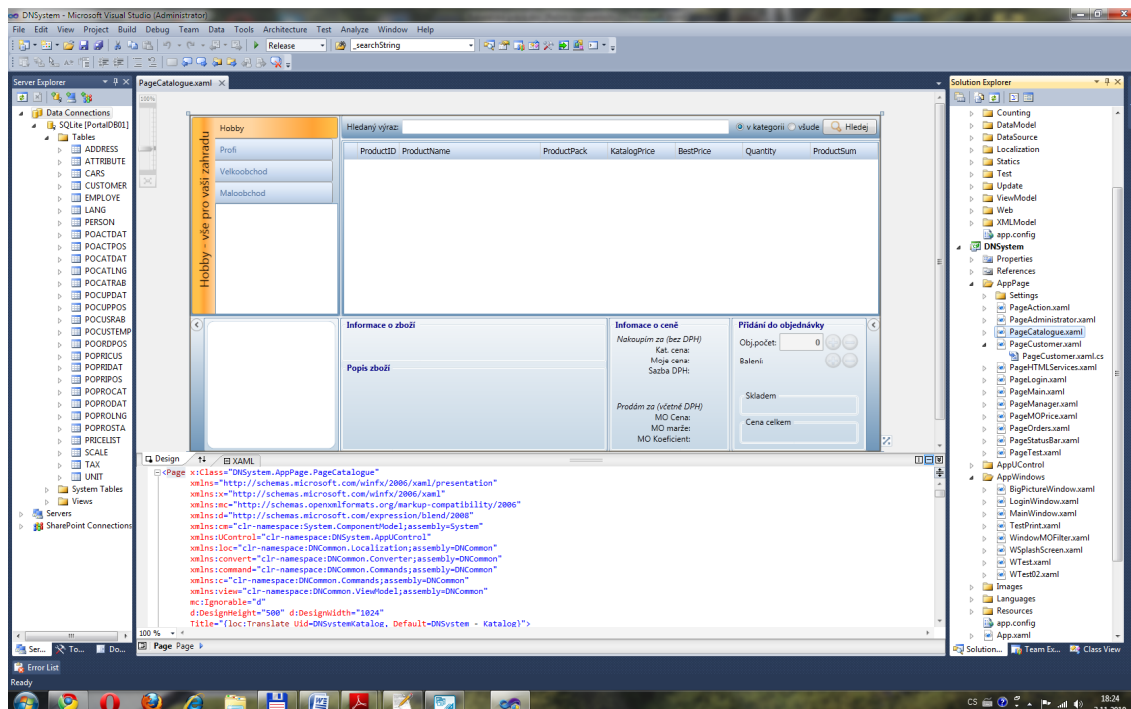
Visual Studio obsahuje editor kódu podporující technologii IntelliSense<sup>11</sup> a

<sup>10</sup> XAML - eXtensible Application Markup Language – čti „zammel“

<sup>11</sup> IntelliSense je funkce provádějící automatické dokončování psaného výrazu. Nabízí se funkce, metody, varaibilní jména atd. Návrhy automatického dokončování se zobrazí ve vyskakovacím seznamu.



refaktorování<sup>12</sup>, díky tomu je rychlost psaní kódu aplikace mnohem snazší a rychlejší. Podporuje jazyky prostřednictvím jazykových služeb, což umožňuje, aby editor kódu a debugger podporoval jakýkoliv programovací jazyk. Mezi vestavěné jazyky patří C/C++, VB.NET a C#. Další vestavěné nástroje zahrnují designer formulářů pro tvorbu grafického rozhraní aplikací (GUI), designer tříd a databázových schémat.



Obrázek 8: Vývojové prostředí Microsoft Visual Studio 2010.

### 4.3. Databáze SQLite

Jako databázový systém byl zvolen SQLite. SQLite je pouze malá knihovna, která se připojí k aplikaci a pomocí jednoduchého rozhraní ji lze začít využívat. Každá databáze je uložena v samostatném souboru „\*.db3“. Formát databázových souborů je přitom nezávislý na operačním systému a SQLite tak představuje i zajímavý a jednoduchý nástroj pro přenos strukturovaných dat. Kapitola čerpá z [18], [19], [20], [21], [22] a [23].

SQLite splňuje podmínky ACID a plně podporuje cizí klíče, operace JOIN. V SQLite je implementován téměř celý standard SQL-92. Databázi SQLite je možné používat na operačních systémech Linux, Mac OS a Windows, kde bude provozován v našem případě.

Velkou výhodou této databáze je, že je šířena pod public domain licencí. Public domain je ekvivalentem volného díla ve smyslu ustanovení § 28 odst. 1 autorského zákona, tedy díla, u kterého uplynula doba trvání majetkových práv a jež tak může každý bez dalšího volně užít a to včetně provádění změn tohoto počítačového programu.

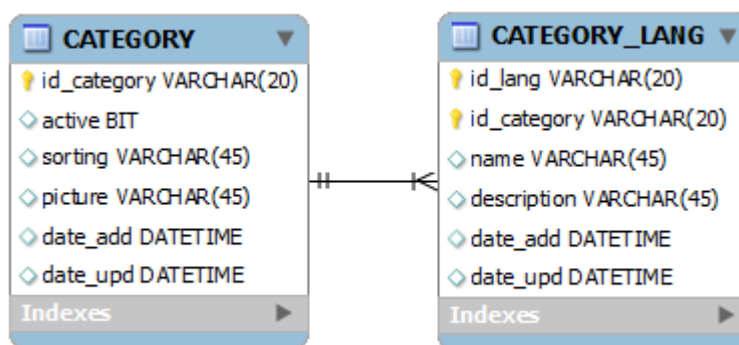
<sup>12</sup> Refaktorování je disciplinovaný proces provádění změn v softwarovém systému takovým způsobem, že nemají vliv na vnější chování kódu, ale vylepšují jeho vnitřní strukturu s minimálním rizikem vnášení chyb. [38]

Následuje výčet faktů, které byly rozhodující pro zvolení SQLite:

- Velmi výhodná public domain licence, která zajišťuje bezplatné šíření a užití databáze
- Vynikající výkonnostní parametry.
- Podpora různých operačních systémů.
- Databáze uložena jen v jednom samostatném souboru.
- Kvalitní a dostupná technická dokumentace.
- Velikost databáze je omezena typem použitého souborového systému.

### Parametry testování databáze

Samotnému výběru databáze předcházelo porovnání parametrů zvolených lehkých databází MS SQL CE a SQLite. Hlavní porovnání se týkalo rychlosti zápisu v transakci a rychlosti čtení z databáze, vzhledem k tomu, že se předpokládá vyšší vytížení databáze v průběhu výpočtu konečné ceny produktu.



Obrázek 9: E-R schéma databáze použité k testování (SQLite, MS SQL CE)

### Insert v transakci

Do prázdné databáze bylo vloženo 4000 záznamů do dvou tabulek v poměru 1:3. To znamená, že ke každé kategorii byly vloženy 3 odpovídající záznamy do tabulky jazyků. Celá operace insert probíhala v transakci.

#### Vložení 4000 záznamů do databáze

měření	SQLite [ms]	SQLite [ms] - syncoff	MS SQL CE [ms]
1	565,032	366,258	1997,109
2	572,015	325,290	2025,821
3	558,153	380,153	1965,021
4	562,524	355,853	2005,958
5	569,003	352,235	2043,508
6	525,815	341,952	1955,150
7	548,265	352,081	1945,807
8	551,856	398,120	1985,120
9	563,951	382,210	1992,861
10	556,508	375,958	1962,971
<b>průměr</b>	<b>557,312</b>	<b>363,011</b>	<b>1987,933</b>

Tabulka 1: Výsledky měření - Insert

Z naměřených výsledků a při spočtení průměrné doby vložení požadovaného množství záznamů do tabulky je databáze SQLite o poznání rychlejší. Navíc tato databáze

umožňuje ještě zvýšit rychlost vložení a to při použití parametru syncff. Rychlost vložení se při použití tohoto parametru téměř zdvojnásobí.

### Select with join

Bylo provedeno 1000 dotazů na zvolenou databázi. Každé volání samostatně otevíralo a uzavíralo spojení s databází. Z výsledků je patrné, že v případě příkazu SELECT s LEFT OUTER JOIN byla databáze SQLite mnohonásobně rychlejší oproti databázi MS SQL CE.

#### 1000 x Select with left join

měření	SQLite [ms]	MS SQL CE [ms]
1	3496,192	27352,230
2	3582,124	27339,236
3	3398,237	27854,230
4	3635,259	27153,854
5	3458,102	27359,804
6	3512,982	27985,159
7	3581,238	27320,837
8	3405,208	26992,128
9	3692,125	27485,845
10	3345,908	27397,052
<b>průměr</b>	<b>3510,737</b>	<b>27424,038</b>

Tabulka 2: Výsledky měření - Select

Z naměřených výsledků je patrné, že databáze SQLite si mnohem efektivněji dokázala poradit i při opakovaných dotazech. Byla téměř 8x rychlejší než databáze MS SQL CE.

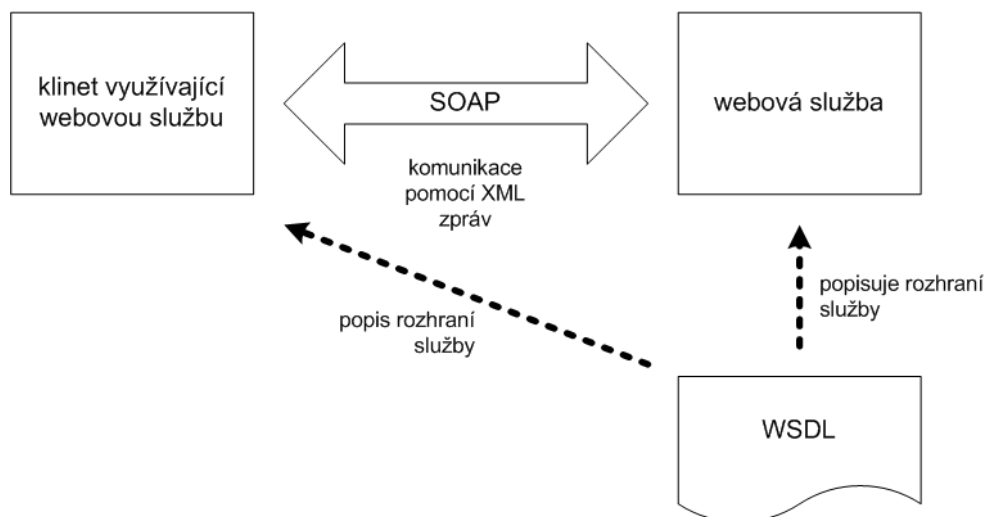
## 4.4. Webové služby

Specifikace W3C popisuje webové služby jako systém navržený pro podporu interoperabilní<sup>13</sup> interakce mezi stroji na síti s rozhraním popsáním ve strojově zpracovatelném formátu, konkrétně WSDL. Kapitola čerpá z [24], [25] a [26].

V základě se jedná o rozhraní, které umožňuje vzdálené volání procedur (RPC). Pro samotnou komunikaci tedy není nutné, aby obě komunikující strany znaly detaily, jakými jsou získávána data, zajímá nás pouze výsledek. Díky tomu, že komunikace probíhá ve standardizované formě, zpravidla v XML, je zaručena vzájemná kompatibilita mezi různými komunikujícími systémy, či platformami.

Komunikace s webovou službou probíhá způsobem předepsaným v popisu služby SOAP pomocí již zavedených protokolů, nejčastěji http, https. Důvodem využití toho druhu komunikace je snadnější průchod firewally.

<sup>13</sup> Interoperabilita je schopnost různých systémů vzájemně spolupracovat, poskytovat si služby, dosáhnout vzájemné součinnosti.



Obrázek 10: Vztah základních technologií (SOAP, WSDL) webových služeb

#### 4.4.1. XML

Extensible Markup Language (zkráceně XML, česky rozšiřitelný značkovací jazyk) je obecný značkovací jazyk, který byl vyvinut a standardizován konsorciem W3C. XML je založen na jednoduchém textu a je zpracovatelný (v případě potřeby) libovolným textovým editorem. Kapitola čerpá z [27] a [28].

Specifikace XML konsorcia W3C je zdarma přístupná všem. Každý tak může bez problémů do svých aplikací implementovat podporu XML. Pomocí XML značek (tagů) vyznačujeme v dokumentu význam jednotlivých částí textu. Dokumenty tak obsahují více informací, než kdyby se používalo značkování zaměřené na prezentaci (vzhled) – definice písma, odsazení a podobně. XML dokumenty jsou informačně bohatší.

XML neobsahuje předdefinované značky (tagy), je třeba definovat vlastní značky, které budeme používat. Tyto značky je možné (nepovinně) definovat v souboru DTD<sup>14</sup>. Potom je možné automaticky kontrolovat, zda vytvářený XML dokument odpovídá této definici. Program, který tyto kontroly provádí, se nazývá parser, ten za nás detekuje většinu chyb v datech. DTD není jediný definiční jazyk pro XML. Neobsahuje možnost kontrolovat typy dat (čísla, měnové údaje, údaje o datu a čase). To je vlastnost, která chybí při zpracování dat databázového charakteru.

<sup>14</sup> DTD (Document Type Definition, česky Definice typu dokumentu)

### 4.4.2. SOAP

SOAP (celým názvem Simple Object Access Protocol) je protokolem pro výměnu zpráv založených na XML přes síť, hlavně pomocí HTTP. Kapitola čerpá z [29] a [30].

Formát SOAP tvoří základní vrstvu komunikace mezi webovými službami a poskytuje prostředí pro tvorbu složitější komunikace. Existuje několik různých druhů šablon pro komunikaci na protokolu SOAP. Nejznámější z nich je RPC šablona, kde jeden z účastníků komunikace je klient a na druhé straně je server. Server ihned odpovídá na požadavky klienta.

Následující pasáž o rozsahu 1 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

#### 4.4.3. WSDL

WSDL (Web Services Description Language) je jazyk, který definuje webovou službu. Popisuje, jaké funkce webová služba nabízí, jaké mají tyto funkce parametry a jaké vrací hodnoty. Pro zápis WSDL je využíván standart XML.

WSDL je často používáno v kombinaci se SOAP, aby poskytovalo webové služby po internetu. Klientský program připojující se k webové službě umí číst WSDL, aby zjistil, jaké funkce jsou na serveru dostupné. Kapitola čerpá z [26], [31] a [32].

##### 4.4.3.1. Základní popis WSDL

WSDL soubor se skládá zejména z následujících elementů, které tvoří základní části každého WSDL popisu.

###### **types**

Obsahuje definici datových struktur používaných ve zprávách.

###### **message**

Abstraktní definice formátu zpráv pomocí definovaných datových typů.

###### **operation**

Abstraktní definice operací, které jsou službou podporovány. U operace se definuje jaké má vstupy a výstupy. Vstup a výstup je popsán již existující zprávou (message).

###### **portType**

Abstraktní seskupení operací. Sdružuje dohromady několik operací.

###### **binding**

Slouží pro navázání určitého typu portu na konkrétní protokol a formát přenosu zpráv.

###### **port**

Jeden koncový bod služby definovaný jako kombinace síťové adresy a dříve definované vazby.

###### **service**

Sdružuje několik koncových bodů do jedné služby.

Pro představu uvádím jednoduchou ukázkou WSDL souboru, který definuje rozhraní služby pro přihlášení aplikace DN-Portál (získání token).

Následující pasáž o rozsahu 1 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.



## 5. Návrh aplikace

### 5.1. Návrh grafické podoby aplikace

Samotný návrh grafické podoby aplikace vychází z požadavku zadání na zachování maximální podobnosti se současným řešením. Tzn., že umístění hlavních komponent aplikace se bude shodovat. Důvod, proč neměníme zásadním způsobem vzhled aplikace, je kladná odezva od koncových uživatelů na současné rozložení informací. Proto v novém návrhu budeme minimálním způsobem zasahovat do rozložení hlavních prvků, využijeme však možností, které nám dají nové technologie.

#### Navrhované změny komponent:

##### Kategorie produktů

K zobrazení kategorie produktů bude použita přehlednější struktura. Jako vhodné se jeví zobrazení formou stromové struktury. Nejsme tak vázáni na maximální hloubku vnoření jednotlivých kategorií.

##### Informace o produktu

V informační části okna umožnit skrytí detailu obrázku produktu a statistiky. Tímto krokem se uvolní místo pro zobrazení. Bude tak možné použít i na počítačích s nižším rozlišením obrazovky.

### 5.2. Uživatele systému

Na základě analýzy a definovaných požadavků můžeme nyní definovat uživatele aplikace.

#### 5.2.1. Zákazník

Zákazník, je uživatel systému, který po autentizaci může pracovat s aplikací DN-Portál. Po přihlášení bude zákazníkovi umožněno:

- Zobrazit informace o firmě, přidat, či upravit adresy, upravit vlastní kontaktní informace (telefon, mail) a změnit své přístupové heslo;
- Zobrazit produkty (v akci a dle kategorie – pro akční produkty vlastní zobrazení);
- Přidat produkty do objednávky;
- Zobrazit produkty v objednávce;
- Odeslat objednávku;
- Exportovat produkty v objednávce do souboru csv;
- Importovat produkty do objednávky ze souboru csv;
- Změnit MOCeny produktů;

#### 5.2.2. Obchodní zástupce

Obchodní zástupce je uživatel systému, který po autentizaci může pracovat s aplikací DN-Portál, v obchodním procesu stará o koncového zákazníka, proto bude mít při práci s aplikací DN-Portál rozšířena práva. Po přihlášení je zobrazeno hlavní okno aplikace, kde může pracovat s těmito moduly:

Po přihlášení bude obchodnímu zástupci umožněno:

- Zobrazit informace o aktivním zákazníkovi (aktuálně zvolený), přidat, či upravit adresy;



- Zobrazit produkty (v akci a dle kategorie – pro akční produkty vlastní zobrazení);
- Přidat produkty do objednávky aktivního zákazníka;
- Zobrazit produkty v objednávce aktivního zákazníka;
- Odeslat objednávku za zákazníka;
- Zobrazit seznam zákazníků a možností výběru nového aktivního zákazníka, aktualizace zákazníka a zařazení zákazníka mezi oblíbené;

### 5.3. Návrh databáze

Cílem je navrhnout kvalitní datovou strukturu pro konkrétní aplikaci a databázový systém, který bude tato aplikace využívat k uložení dat. Při datovém modelování obvykle vytváříme nejprve konceptuální<sup>15</sup> datový model [33]. Konceptuální datový model představuje určité zobecnění oproti konkrétní implementaci datové struktury v relační, případně objektové databázi. Zobecněním získáme nezávislost modelu na konkrétním databázovém systému, ale zároveň jsme schopni tento model kdykoliv převést do konkrétního implementačního prostředí. V našem případě zvolené SQLite databáze.

Následující pasáž o rozsahu 2 stran obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

---

<sup>15</sup> Koncept je první předběžný návrh.



#### **5.4. Návrh komunikace**

Návrh komunikace spočívá v definici základních požadavků (dotazů), které budeme při komunikaci se serverem využívat. Z důvodů úpory přenosové cesty a urychlení přenosu dat budeme veškerou komunikaci se serverem komprimovat. V budoucnu se navíc počítá s převedením komunikace ze standardního http na https, abychom maximálním způsobem zabezpečili přenášená data. V současné verzi si vystačíme s unikátním identifikátorem (tzv. token), který obdrží každá úspěšně přihlášená aplikace. Při jakémkoli požadavku na server bude token striktně vyžadován. Serverová strana bude ověřovat jeho platnost, v případě úspěchu poskytne aplikaci požadovaná data, v opačném případě bude požadovat nové přihlášení.

Následující pasáž o rozsahu 3 stran obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.





### **5.5. Návrh aktualizace**

Abychom aplikaci udržovali aktuální, je třeba všechny její součásti pravidelně aktualizovat, pokud je aktualizace k dispozici. Samotnou aktualizaci jsem rozdělil do třech částí. Aktualizaci databáze, programu a obrázků.

Následující pasáž o rozsahu 1/2 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

## 6. Konstrukce aplikace

### 6.1. Design rozhraní – použití stylů

Při sestavování grafické podoby nové aplikace, jsem se snažil co nejvíce zachovat základní rozložení hlavních ovládacích prvků, zároveň však bylo cílem využít nových vlastností WPF, které nám grafický návrh značně zjednoduší. Typickým příkladem je použití styl. Když vytváříme uživatelské rozhraní a chceme zachovat jednotný vzhled, jsme často nuceni psát některé vlastnosti stále znovu. Při použití stylů toto není nutné. Vzhled prvku je možné deklarovat jen jednou a na tuto deklaraci se v místě potřeby jen odkazovat. Tato kapitola čerpá z [34] a [35].

Styly je vhodné deklarovat na co nejvyšší úrovni, ať už je to Application, Window/Page nebo v samostatné ResourceDictionary. V aplikaci tedy využijeme již zmiňovanou ResourceDictionary jako hlavní knihovnu stylů, kterou pak pomocí property MergedDictionaries připojíme k Window, Page či UserControl.

Následující příklad názorně ukazuje, jak je možné definovat styl tlačítka a jak jej aplikovat.

```
<Window>
  <Window.Resources>
    <Style x:Key="newButton" TargetType="Button">
      <Setter Property="Background" Value="Orange" />
      <Setter Property="FontStyle" Value="Italic" />
      <Setter Property="Padding" Value="8,4" />
      <Setter Property="Margin" Value="4" />
    </Style>
  </Window.Resources>

  <StackPanel Orientation="Horizontal" VerticalAlignment="Top">
    <Button Style="{StaticResource newButton}">Button01</Button>
    <Button Style="{StaticResource newButton}">Button02</Button>
    <Button Style="{StaticResource newButton}">Button03</Button>
  </StackPanel>
</Window>
```

Zdrojový kód 2: Ukázka definice a použití stylu v aplikaci

Použitím stylů je možné jednoduše změnit vzhled aplikace a přizpůsobit tak vzhled požadavkům zákazníka během krátké doby.

### 6.2. Popis rozhraní

#### 6.2.1. Role zákazník

Zákazník je koncový uživatel systému DN-Portál, který má přístup do aplikace po zadání správných přihlašovacích údajů.

##### 6.2.1.1. Přihlášení zákazníka

Po spuštění programu je zobrazena obrazovka s požadavkem na zadání přihlašovacích údajů. Pro úspěšné spuštění aplikace, je nutné zadat správnou kombinaci uživatelského jména a hesla. Systém umožňuje několik způsobů přihlášení.

Následující pasáž o rozsahu 1/2 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

### **6.2.2. Role obchodní zástupce**

Obchodní zástupce je osoba, která má v obchodním procesu na starosti koncového zákazníka. Do aplikace DN-Portál má obchodní zástupce rozšířený přístup, ve kterém může pracovat s přiřazenými zákazníky. Za vybraného zákazníka může obchodní zástupce zpracovat a odeslat objednávku.

#### **6.2.2.1. Přihlášení obchodního zástupce**

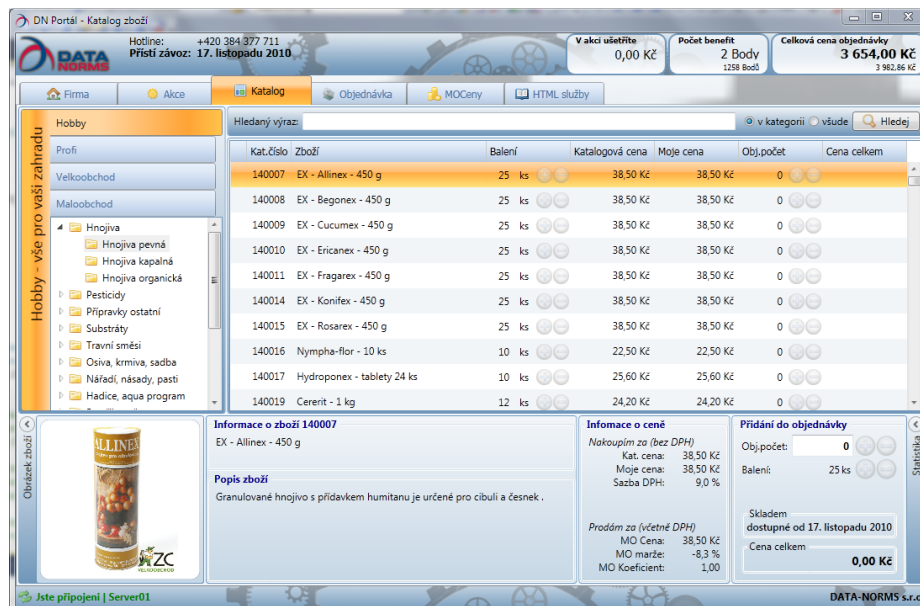
Po spuštění programu je zobrazena obrazovka s požadavkem na zadání přihlašovacích údajů. Pro úspěšné přihlášení je nutné zadat správnou kombinaci uživatelského jména a hesla.

Následující pasáž o rozsahu 1/2 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.



### 6.2.3. Hlavní okno aplikace

Jedním z hlavních požadavků bylo zachování základního konceptu stávající aplikace. Důvodem zachování je kladná odezva od uživatelů současné verze aplikace na zavedenou grafickou podobu, její přehlednost, jednoduchost a intuitivnost. Výsledná podoba aplikace DN-Portál je zobrazena na následujícím obrázku.



Obrázek 11: Hlavní okno nové verze aplikace DN-Portál

Hlavní okno nové verze aplikace je rozděleno do třech základních sekcí:

#### 6.2.3.1. Hlavní informační sekce

Hlavní informační sekce obsahuje informace o příštím závozu zboží k zákazníkovi, aktuální celkové ceně zboží vložené do objednávky, celkové ušetřené ceně při nákupu zboží v akci a informaci o aktuálním a celkovém počtu benefit bodů. V modifikaci

aplikace pro obchodní zástupce je zobrazena ještě informace o aktuálním zvoleném zákazníkovi, se kterým obchodní zástupce právě pracuje.

 Hotline: +420 384 377 711 Příští závoz: 7. prosince 2010	V akci ušetříte 0,00 Kč	Počet benefit 0 Bodů <small>1258 Bodů</small>	Celková cena objednávky <b>0,00 Kč</b> <small>0,00 Kč</small>
	Aktivní zákazník 1002 Zahradnictví Šenk	V akci ušetříte 0,00 Kč	Počet benefit 0 Bodů <small>1258 Bodů</small>

Obrázek 12: Hlavní informace, porovnání verze pro zákazníka a obchodního zástupce

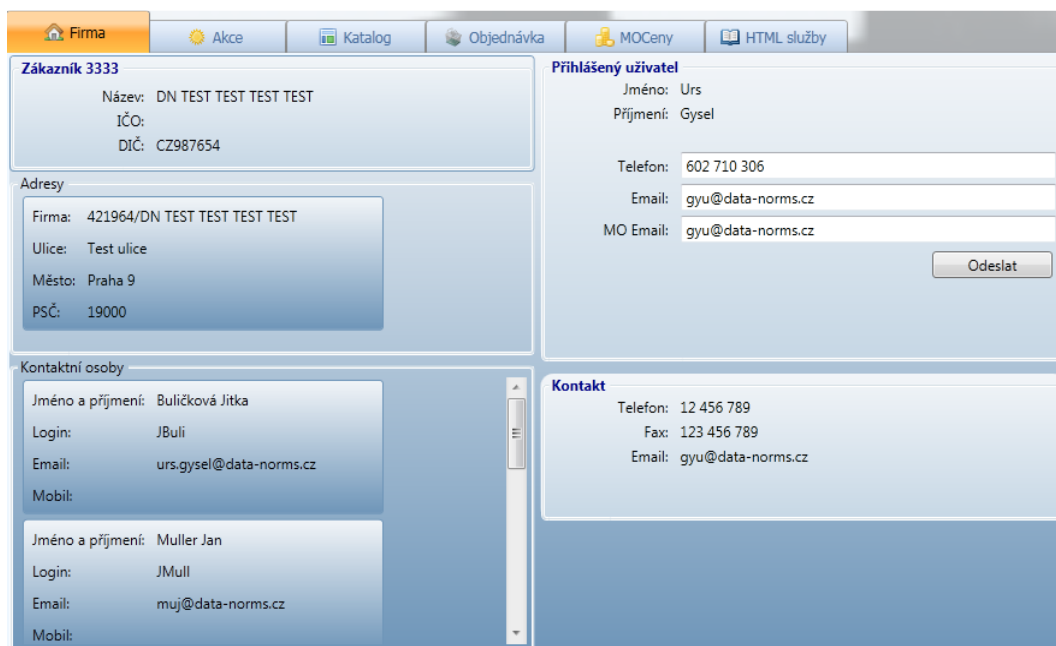
### 6.2.3.2. Hlavní sekce

Hlavní sekce je dělena na jednotlivé záložky (moduly), jejich popisu se budeme věnovat v následujících bodech.

#### Modul Firma

Obsahuje informace o aktivním zákazníkovi a právě přihlášeném uživateli.

Informace o firmě (Název, IČO, DIČ), adresy (dodací a fakturační), Aktivní uživatel (Jméno, příjmení, mail). Je zde umožněno změnit zasílací emailovou adresu. Ostatní změny nejsou v aplikaci povoleny. Ostatní změny jsou možná pouze přímo v ERP systému i/2.



Obrázek 13: Modul Firma

#### Modul Akce

Obsahuje základní přehled všech aktuálních produktů, které jsou v systému vedené jako akční položky, tzn., že při nákupu zboží v akci, při dodržení minimálního odebraného množství zákazník ušetří oproti své standardní individuální ceně. Záložka akce je dělena do třech sekcí.

- Vyhledávání;
- Přehled produktů – jen produkty v akci;
- Detailní informace o právě vybraném produktu;

Kat.číslo	Zboží	Balení	Katalogová cena	Akční cena	Moje cena	Ušetříte	Obj.pocet	Cena celkem
010020	Travní směs - Univerzální 1 kg	20 ks	81,00 Kč	58,70 Kč	64,57 Kč	5,87 Kč	1	58,70 Kč
100004	Aliette 80 WG - 2x5 g	35 ks	35,50 Kč	27,44 Kč	30,18 Kč		0	
100019	Dithane DG Neotec - 2x10 g	35 ks	29,00 Kč	22,05 Kč	24,26 Kč		0	
100028	Kocide 2000 - 2x17,5 g	35 ks	41,50 Kč	32,83 Kč	36,11 Kč		0	
100099	Acrobat MZ - 2x10 g	35 ks	39,90 Kč	30,38 Kč	33,42 Kč		0	
100100	Baycor 25 WP - 2x5 g	35 ks	39,90 Kč	30,87 Kč	33,65 Kč	3,15 Kč	1	30,50 Kč
100185	Kuprikol 50 - 2x10 g	35 ks	27,20 Kč	21,07 Kč	23,18 Kč		0	
100208	Ortiva - 10 ml	50 ks	53,00 Kč	41,65 Kč	45,75 Kč	4,25 Kč	1	41,50 Kč
110003	Agrion Delta - 500 ml rozprašovač	10 ks	62,90 Kč	48,51 Kč	53,36 Kč		0	
110004	Agrion Delta - 500 ml náplň	10 ks	51,30 Kč	38,71 Kč	42,58 Kč		0	

	<b>Informace o zboží 010020</b> Travní směs - Univerzální 1 kg zboží v akci  <b>Popis zboží</b> AKČNÍ CENA 59,90 Kč/ks, platí od 20 ks, v období od 04.06.10 do 02.07.10. Travní směs vhodná k rychlému ozelenění ploch, na které neklademe zvláštní požadavky. Doba výsevu III-X. Výsevka cca 10-12g na m2. Složení: jilek mnohokvětý 30%, jilek jednolet	<b>Informace o ceně</b> Nakoupím za (bez DPH) Kat. cena: 81,00 Kč Moje cena: 58,70 Kč Sazba DPH: 9,0 %  Prodám za (včetně DPH) MO Cena: 81,00 Kč MO marže: 26,6 % MO Koefficient: 1,38	<b>Přidání do objednávky</b> Obj.pocet: 1 Balení: 20 ks  Skladem dostupné od 10. prosince 2010 Cena celkem <b>58,70 Kč</b>
--	---	---	---

Obrázek 14: Modul Akce

### Modul Katalog

Tabulkový výpis zboží dle vybrané kategorie zboží. Je děleno do čtyř sekcí.

- Kategorie zboží;
- Vyhledávání;
- Přehled produktů – dle zvolené kategorie;
- Detailní informace o právě vybraném produktu;

Kat.číslo	Zboží	Balení	Katalogová cena	Moje cena	Obj.pocet	Cena celkem
140037	Hořtický hnůj - 3 kg s rohovinou	6 ks	72,00 Kč	63,65 Kč	0	
140171	Koňský hnůj ZC produkt - 6 kg kbelik s roř	1 ks	165,00 Kč	145,85 Kč	0	
140172	Koňský hnůj ZC produkt - 10 kg s rohovin	1 ks	199,50 Kč	176,35 Kč	0	
140173	Koňský hnůj ZC produkt - 3 kg s rohovinou	6 ks	66,00 Kč	58,34 Kč	0	
140174	Týčinky hořtické 30 ks	10 ks	26,90 Kč	23,78 Kč	0	
140178	Kravný hnůj - 3 kg	6 ks	54,20 Kč	47,91 Kč	0	
140179	Kravný hnůj - 10 kg	1 ks	165,00 Kč	145,85 Kč	0	
140181	Slepičince tekuté - 500 ml	12 ks	48,30 Kč	42,70 Kč	0	
140189	Slepičince granulované - 2,5 kg	10 ks	75,90 Kč	67,09 Kč	0	
140213	NPK - 1 kg hořtické s guánem	12 ks	20,90 Kč	18,48 Kč	0	

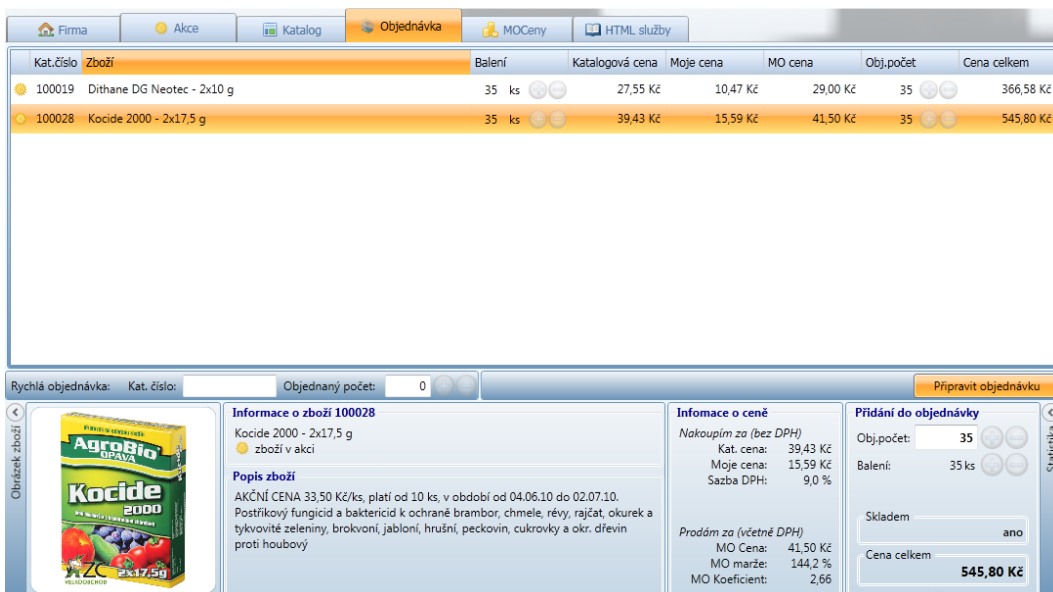
	<b>Informace o zboží 140181</b> Slepičince tekuté - 500 ml  <b>Popis zboží</b> Kapalně organicko-minerální hnojivo, jehož důležitou součástí je drůbeží trus. Rostlinám dodává potřebné živiny, výrazně aktivuje půdní mikroflóru. Vysoký obsah fosforu zaručuje bohatou násadu květů a u plodin velkou úrodu /15ml hnojiva na 3l vody/.	<b>Informace o ceně</b> Nakoupím za (bez DPH) Kat. cena: 48,30 Kč Moje cena: 42,70 Kč Sazba DPH: 9,0 %  Prodám za (včetně DPH) MO Cena: 48,30 Kč MO marže: 3,8 % MO Koefficient: 1,13	<b>Přidání do objednávky</b> Obj.pocet: 0 Balení: 12 ks  Skladem dostupné od 7. prosince 2010 Cena celkem <b>0,00 Kč</b>
--	--	--	---

Obrázek 15: Modul Katalog

### Modul Objednávka

Tabulkový výpis položek vložených do objednávky s podrobnějším rozpisem konečných cen. Je děleno do čtyř sekcí

- Přehled objednaných produktů;
- Rychlá objednávka;
- Detailní informace o právě vybraném produktu;
- Odeslání objednávky, Export, Import;

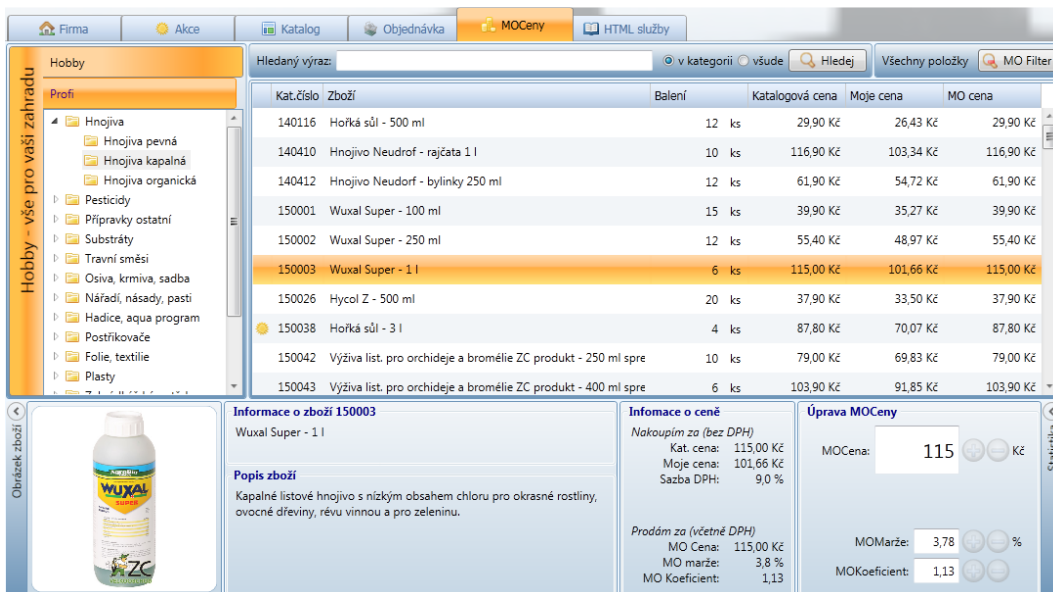


Obrázek 16: Modul Objednávka

### Modul Maloobchodní ceny

Tabulkový výpis zboží dle vybrané kategorie zboží. Zobrazení je z velké části shodné se zobrazím katalogu. Odlišností je, že se zbožím se v této sekci pracuje jen v režimu úpravy maloobchodních cen. Je děleno do čtyř sekcí.

- Kategorie zboží;
- Vyhledávání;
- Přehled produktů dle zvolené kategorie;
- Detailní informace o právě vybraném produktu s možností zadání maloobchodní ceny, marže, či koeficientu;



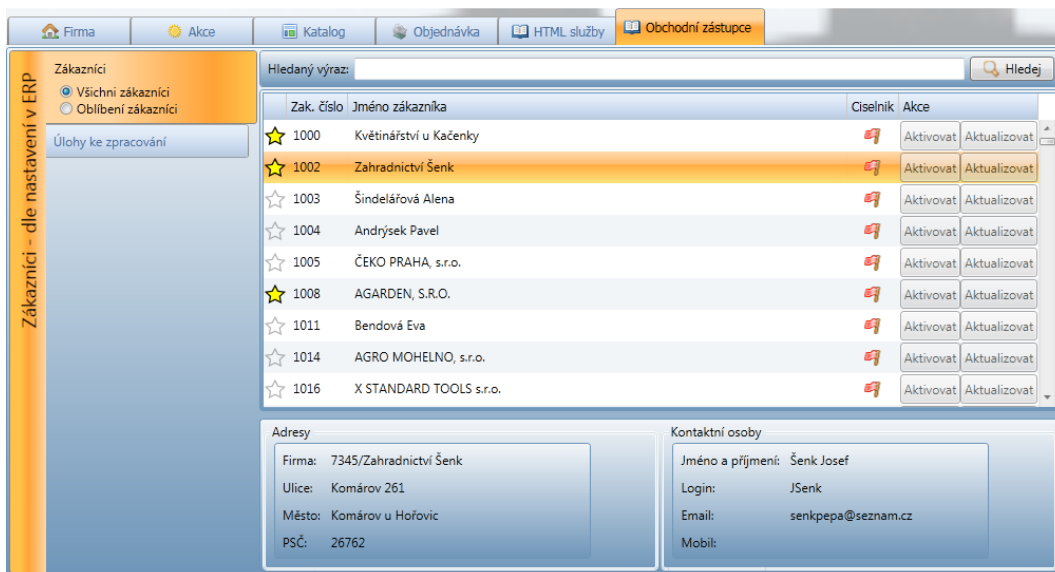
Obrázek 17: Modul MOCeny

### Modul Obchodní zástupce

Obsahuje přehled zákazníků, se kterými může obchodní zástupce pracovat. Samotné zobrazení je formou tabulkového výpisu. Při výběru zákazníka jsou ve spodním informačním pruhu zobrazeny informace o adresách zákazníka a o kontaktních osobách s dodatečnými údaji, jako jsou mail, telefon, přihlašovací jméno.

Základní dělení:

- Vyhledávání;
- Přehled zákazníků;
- Detailní informace o právě vybraném zákazníkovi (adresy a kontaktní osoby);

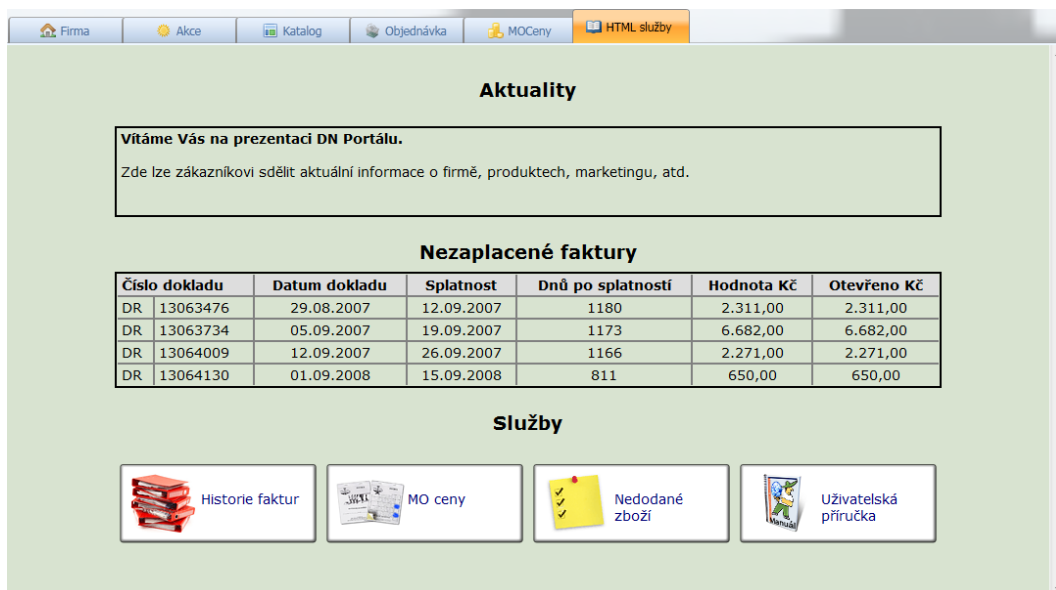


Obrázek 18: Modul Obchodní zástupce

### Modul HTML služby

Je komponenta zobrazující webovou stránku s proměnným obsahem závislým na přihlášeném zákazníkovi a zvolené sekci. Uvedu jen základní strany/funkce, které se mohou zobrazit přihlášenému uživateli bez dalšího popisu.

Zboží v rezervaci, přehled zaplacených faktur, faktury ke stažení a uživatelská příručka.



Obrázek 19: Modul HTML Služby

### 6.2.3.3. Status

Status bar, ve kterém jsou zobrazovány informace o aktivním spojení s cílovým serverem a informace o průběhu aktualizace.

## 6.2.4. Komponenty

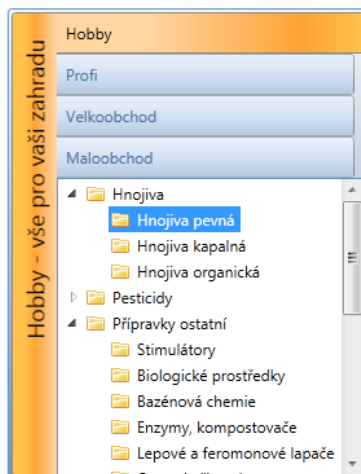
### 6.2.4.1. Kategorie zboží

Vzhled této komponenty prošel asi největší změnou oproti původní verzi. Od jednoduchého zobrazení kategorie a podkategorie použitím datagrid, přecházíme na elegantnější řešení zobrazení kategorií ve stromové struktuře s využitím ovládacího prvku TreeView. Tento prvek nabízí velice pohodlný způsob plnění, a to pomocí hierarchických datových šablon ve spojení s data bindingem.

```
<TreeView
  Name="viewsTreeView"
  ItemsSource="{Binding Path=CategoryViewModels}"
  SelectedItemChanged="viewsTreeView_SelectedItemChanged"
  BorderThickness="0">
  <TreeView.Resources>
    <HierarchicalDataTemplate
      DataType="{x:Type view:CategoryVM}"
      ItemsSource="{Binding Path=SubCategories}">
      <Grid Margin="0,0,5,0">
        <Grid.RowDefinitions>
          <RowDefinition Height="20" />
        </Grid.RowDefinitions>
        <Grid.ColumnDefinitions>
          <ColumnDefinition Width="Auto" />
          <ColumnDefinition Width="*" />
        </Grid.ColumnDefinitions>
        <Image Source="{Binding Path=CategoryPicture,
          Converter={StaticResource ConverterIcon}}"
          Width="16" Height="16" Margin="0,0,5,0"
          VerticalAlignment="Center" HorizontalAlignment="Left"/>
        <TextBlock
          Grid.Column="1"
          Text="{Binding Path=CategoryName}"
          VerticalAlignment="Center" HorizontalAlignment="Left"/>
      </Grid>
    </HierarchicalDataTemplate>
  </TreeView.Resources>
</TreeView>
```

Zdrojový kód 3: XAML TreeView hierarching data binding – Kategorie zboží

Hlavní kategorie zboží jsou prvky TabItem ovládacího prvku TabControl. Využitím stylů jsem zmíněné ovládací prvky upravil a výsledná podoba komponenty kategorie zboží je následující:



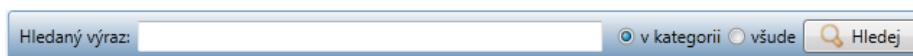
Obrázek 20: Komponenta – Kategorie zboží

#### 6.2.4.2. Vyhledávání

Komponenta, která filtruje dle zadaného hledaného výrazu seznam produktů na aktuální vybrané záložce „akce/katalog/MOCeny. Po zadání hledaného výrazu je automaticky započato fulltextové vyhledávání podle čísla produktu, jeho jména a popisu. Na záložce Katalog a MOCeny je navíc zobrazena dvojice přepínačů (radiobutton), která slouží k výběru typu vyhledávání:

- V kategorii – vyhledáváme jen nad vybranou skupinou či podskupinou zboží;
- Všude – vyhledáváme nad celým katalogem zboží bez ohledu na aktuální vybranou kategorii či podkategorii;

Ve verzi pro obchodní zástupce je stejná komponenta využita pro filtrování seznamu zákazníků na záložce „Obchodní zástupce“. Při vyhledávání v seznamu zákazníků je hledáno v čísle zákazníka, jeho jméně, v adresách i kontaktních osobách. Výběr typu vyhledávání je skryt.



Obrázek 21: Komponenta – Vyhledávání

#### 6.2.4.3. Katalog produktů

Zobrazuje základní informace o nabízených produktech, jako jsou blokační znak, katalogové číslo, balení produktu, katalogová cena, bestprice cena (konečná individuální cena), objednaný počet a cena celkem.

Kat.číslo	Zboží	Balení	Katalogová cena	Moje cena	Obj.počet	Cena celkem
140029	Ceririt - 10 kg	3 ks	206,00 Kč	182,10 Kč	0	
↓ 140031	Ledek vápenatý - 3 kg	10 ks	40,80 Kč	36,07 Kč	0	
↓ 140032	Ledek amonný s vápen	10 ks	47,90 Kč	42,34 Kč	0	
140035	NPK - 50 kg (15-15-15)	1 ks	852,00 Kč	918,46 Kč	0	
140036	Síran amonný - 3 kg	10 ks	39,90 Kč	35,27 Kč	0	
140040	Dusíkaté vápno - 1 kg	12 ks	58,90 Kč	52,07 Kč	0	
↓ 140043	Síran draselný - 1 kg	12 ks	39,70 Kč	35,09 Kč	0	

Obrázek 22: Komponenta - Katalog produktů

Produkty zobrazené v této komponentě jsou řazeny prioritně dle třídícího znaku (určuje se v ERP i/2). Zákazník má však možnost řazení přeskupit např. dle čísla či názvu zboží a to jak vzestupně tak sestupně. Ostatní zobrazené sloupce není možné řadit. Pohyb v komponentě je umožněn standardními klávesami.

#### Objednání zboží je možné několika způsoby:

- Stiskem kláves +(Add)/- (Substract);
- Použitím myši kliknutím na příslušné ovládací prvky (Button) +/-;

Zobrazované informace o cenách (katalogová a individuální cena) jsou ve výchozím stavu (u zboží, bez objednaného množství) zobrazeny jako ceny za 1 měrnou jednotku. Po změně objednaného počtu dojde k automatickému přepočtu celkové ceny produktu a zboží je vloženo do objednávky. Automatickým přepočtem se rozumí vypočtení individuální ceny za produkt pro daného zákazníka pro určené množství. Výpočtu ceny se budeme věnovat podrobněji věnovat v kapitole 0.

#### 6.2.4.4. Informace o produktu

##### Obrázek produktu

Komponenta je svázána s aktuálně vybraným zbožím v katalogu produktů, zobrazuje platný obrázek produktu, pokud je nalezen na lokálním disku v příslušném adresáři.



```
<Border
  CornerRadius="3"
  Background="{StaticResource mainBackgroundInfo}"
  Margin="0,0,0,0"
  Width="Auto"
  Height="Auto" BorderThickness="1"
  BorderBrush="{StaticResource BorderBrush}">
  <Border
    CornerRadius="10"
    Background="White"
    Width="190"
    Height="190" BorderThickness="0.5"
    BorderBrush="{StaticResource BorderBrush}">
    <Image
      Source="{Binding Path=DataRow.ProductID,
        Converter={StaticResource ConverterImage}}"
      MaxHeight="180"
      MaxWidth="180"
      MouseDown="Image_MouseDown"
    />
  </Border>
</Border>
```

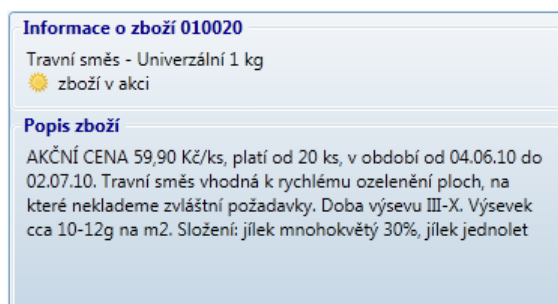
Obrázek 23: Komponenta – Obrázek produktu (vzhled + xaml)



Pro zobrazení obrázku využijeme s velkou výhodou konvertoru, definoval jsem totiž, že jméno obrázku je shodné s identifikačním číslem zboží. Ukázka použití konvertoru je v příloze B.

### Informace o produktu

Komponenta je svázána s aktuálně vybraným zbožím v katalogu produktů, zobrazuje dodatečné informace o produktu, kterými jsou: název, popis a informace o blokačním znaku.



Obrázek 24: komponenta – Rozšířené informace o produktu

Pro zobrazení ikony a popisu blokačního znaku využijeme opět konvertoru.

### Ceny produktu

Komponenta je svázána s aktuálně vybraným zbožím v katalogu produktů, zobrazuje informace o ceně produktu. Konečnou katalogovou cenu (základní nebo stupnicovou, bez aplikovaných slev) a konečnou zákaznickou cenu (bestprice). Navíc obsahuje informace o maloobchodní ceně s odpovídajícím koeficientem a marží.

Informace o ceně	
<i>Nakoupím za (bez DPH)</i>	
Kat. cena:	81,00 Kč
Moje cena:	64,57 Kč
Sazba DPH:	9,0 %
<i>Prodám za (včetně DPH)</i>	
MO Cena:	81,00 Kč
MO marže:	15,1 %
MO Koeficient:	1,25

Obrázek 25: Komponenta – Ceny produktu

### Objednání produktu

Komponenta je svázána s aktuálně vybraným zbožím v katalogu produktů, zobrazuje aktuální objednané množství, stav skladem a cenu produktu za objednané množství. Komponenta navíc umožňuje objednání zboží. Po změně objednaného počtu dojde k automatickému přepočtu celkové ceny produktu a zboží je vloženo do objednávky. Automatickým přepočtem se rozumí vypočtení individuální ceny za produkt pro daného zákazníka pro určené množství. Výpočtu ceny se budeme věnovat podrobněji v kapitole 0.

**Přidání do objednávky**

Obj.pocet:

Balení:

Skladem:

Cena celkem:

Obrázek 26: Komponenta – Manipulace s produktem

**Objednání zboží je možné několika způsoby.**

- Zapsáním požadovaného množství;
- Použitím myši kliknutím na příslušné ovládací prvky (Button) +/-;

**Statistika produktu**

Komponenta je svázána s aktuálně vybraným zbožím v katalogu produktů, zobrazuje informace o počtu zakoupených kusů vybraného produktu v předchozím období.

Měsíc	2009	2010
leden	0	0
únor	0	0
březen	0	0
duben	0	0
květen	0	0

Obrázek 27: Komponenta – Statistika produktu

**MOCena**

Komponenta je svázána s aktuálně vybraným zbožím v katalogu produktů, je přístupná pouze na záložce „MOCeny“. Umožňuje změnu uložené maloobchodní ceny, marže či koeficientu. Změna kterékoliv hodnoty vyvolá automatický přepočítání ostatních hodnot.

**Úprava MOCeny**

MOCena:    Kč

MOMarže:    %

MOKoeficient:

Obrázek 28: Komponenta – Maloobchodní ceny produktu

**6.3. Komunikace aplikace**

Následující pasáž o rozsahu 1 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

**6.4. Aktualizace**

Následující pasáž o rozsahu 1 strany obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.

**6.5.           Kontroly a výpočty**

Následující pasáž o rozsahu 5 stran obsahuje utajované skutečnosti a je obsažena pouze v originále bakalářské práce uložené na Pedagogické fakultě JU.











## 7. Testování aplikace

Testování aplikace je nedílnou součástí vývoje softwarových řešení. Testování probíhalo po celou dobu vývoje aplikace, po dokončení jednotlivých komponent či funkcí vždy následovalo ověření, zda výsledek odpovídá očekávanému výstupu a zda splňuje všechny stanovené požadavky. Výkon a stabilita aplikace jsou dvě hlavní kritéria úspěšnosti, pokud je aplikace rychlá, má šanci na úspěch, pokud je aplikace pomalá, je velmi pravděpodobné, že bude rychle zapomenuta. Proto byl při vývoji a testování kladen důraz na nalezení co nejvhodnějšího postupu a optimalizaci kódu a výkonných skriptů tak, aby aplikace splňovala i tyto náročné podmínky.

### 7.1. Odhalená a následně odstraněná slabá místa v návrhu

#### Rychlost načítání přehledu produktů

Při vývoji aplikace bylo odhaleno slabé místo v návrhu, které snižovalo komfort aplikace. Jednalo se o nižší rychlost načítání seznamu produktů při změně kategorie, či při využití vyhledávání. Důvodem zpomalení bylo přesunutí výpočtu konečné ceny ze serveru na klientský počítač. Výpočet konečné ceny pro jeden produkt je časově náročná operace, před zobrazením výsledného seznamu produktů bylo třeba pro každý produkt spočítat konečnou cenu za 1mj. Pro představu přehled obsahující 250 položek byl načten za cca 2,5s což odpovídá průměrnému času přepočtu konečné ceny na jeden produkt cca 10ms.

Tento problém byl odstraněn zavedením nové tabulky (POPRICUS), která obsahuje předem přepočtené konečné ceny produktů na 1mj. Načtení položek se tak zkrátilo na průměrnou dobu cca 50-100ms, což už je pro uživatele zanedbatelná doba, která odpovídá rychlosti načítání přehledu produktů v původní verzi. Nevýhodou toho přístupu je, že je nutné po aktualizaci základních dat pro výpočet ceny provést přepočtení všech cen produktů. Jedině tak zabezpečíme, že uživatel bude mít zobrazena vždy aktuální platná data a spočteny správné ceny. Přepočet cen v databázi s 5000 záznamy s odpovídajícím množstvím katalogových, akčních a individuální cen a při aplikaci platných slev je výpočet proveden za cca 1 min. Přepočet probíhá v odděleném procesu a neovlivňuje tak možnost práce s aplikací.

## 8. Závěr

Cílem bakalářské práce bylo vytvořit novou verzi aplikace DN-Portál, která bude kvalitním základem pro další vývoj této aplikace dle požadavků zákazníků.

Hlavním důvodem pro vývoj nové verze však byla skutečnost, že aktuální aplikace neumožňuje efektivní práci obchodních zástupců, dále že ve stavu offline nezobrazuje zákazníkovi jeho konečnou cenu po aplikaci všech souvisejících slev.

Shrňme hlavní rozdíly mezi současnou a novou verzí aplikace:

Současná verze	Nová verze
<ul style="list-style-type: none"> <li>• Windows Forms</li> <li>• .NET 2.0</li> <li>• Databáze Microsoft Access</li> <li>• Komunikace probíhá pomocí HTTP GET</li> <li>• Neobsahuje verzi pro obchodní zástupce</li> <li>• Cenový model obsahuje jen základní ceny, skupinové ceny jsou zobrazeny jen v online stavu</li> <li>• Aktualizace databáze není optimální - rychlost</li> <li>• Aktualizace obrázků neřeší odstranění či poškození obrázku z disku</li> </ul>	<ul style="list-style-type: none"> <li>• WPF</li> <li>• .NET 4.0</li> <li>• Databáze SQLite</li> <li>• Komunikace probíhá pomocí SOAP</li> <li>• Obsahuje verzi pro obchodní zástupce</li> <li>• Cenový model obsahuje katalogové, akční a individuální ceny včetně stupnicových cen, obsahuje individuální a skupinové slevy</li> <li>• Rychlejší aktualizace databáze</li> <li>• Navržena aktualizace obrázků - ve vývoji</li> <li>• Navržena aktualizace programu - ve vývoji</li> </ul>

Při vývoji jsem kladl důraz na rozložení základních ovládacích prvků a zachování jednoduchosti a intuitivnosti ovládání aplikace.

Následující obrázek názorně zobrazuje rozdíl v grafické podobě současné a nové verze aplikace DN-Portál.

DN Portál - Katalog zboží

Hotline: +420 384 377 711  
Příští závoz: 17. listopadu 2010

V akci ušetřite: 0,00 Kč  
Počet benefit: 2 Body (1258 Bodů)  
Celková cena objednávky: 3 654,00 Kč (3 982,86 Kč)

Firma Akce Katalog Objednávka MOCeny HTML služby

Hledaný výraz:  v kategorii všude Hledej

Kat.číslo	Zboží	Balení	Katalogová cena	Moje cena	Obj.počet	Cena celkem
140007	EX - Allinex - 450 g	25 ks	38,50 Kč	38,50 Kč	0	
140008	EX - Begonex - 450 g	25 ks	38,50 Kč	38,50 Kč	0	
140009	EX - Cucumex - 450 g	25 ks	38,50 Kč	38,50 Kč	0	
140010	EX - Ericanex - 450 g	25 ks	38,50 Kč	38,50 Kč	0	
140011	EX - Fragarex - 450 g	25 ks	38,50 Kč	38,50 Kč	0	
140014	EX - Konifex - 450 g	25 ks	38,50 Kč	38,50 Kč	0	
140015	EX - Rosarex - 450 g	25 ks	38,50 Kč	38,50 Kč	0	
140016	Nympha-flor - 10 ks	10 ks	22,50 Kč	22,50 Kč	0	
140017	Hydroponex - tablety 24 ks	10 ks	25,60 Kč	25,60 Kč	0	
140019	Ceririt - 1 kg	12 ks	24,20 Kč	24,20 Kč	0	

**Informace o zboží 140007**  
EX - Allinex - 450 g

**Popis zboží**  
Granulované hnojivo s přísadkou humatanu je určené pro cibuli a česnek.

**Informace o ceně**  
Nakoupím za (bez DPH):  
Kat. cena: 38,50 Kč  
Moje cena: 38,50 Kč  
Sazba DPH: 9,0 %  
Prodám za (včetně DPH):  
MO Cena: 38,50 Kč  
MO marže: -8,3 %  
MO Koefficient: 1,00

**Přidání do objednávky**  
Obj.počet: 0  
Balení: 25 ks  
Skladem dostupné od 17. listopadu 2010  
Cena celkem: 0,00 Kč

Jste připojeni | Server01 DATA-NORMS s.r.o.

ZC s.r.o. - ZC Portál

Hotline: +420 384 370 900  
Příští závoz: 10.11.2010  
Benefit program celkem: 0 ZC Bonů

Za tuto objednávku: 0 ZC Bonů  
Celková cena objednávky: 504,89 Kč (422,45 Kč)

Firma Akce Katalog Objednávka MO ceny Služby

Zahradkářské potřeby Semena

Hlavní skupina zboží Novinky Akce Hnojiva Pesticidy Přípravky ostatní Substráty Travní směsi

Podskupina zboží Substráty Florcom Substráty Agro CS Substráty Rašelina Substráty Gramoflor Keramzity, dřev. perlit Substráty Floria Substráty žampiony a hlíva Substráty Forestina Substráty Pokon Všechno

Vyhledávání hledaný výraz:  Hledej

Katalog zboží

Kat.číslo	Zboží	Balení	Katalogová cena	Nákupní cena	Obj.počet	Celková cena
180098	Cocochips - kokosové kousky 500 g	24 ks	32,00	26,24	0	0,00
180099	Cocopress - kokosové vlákno 650 g	24 ks	29,00	23,76	0	0,00
180117	Dřv mramorová 5l - 4-7 mm	10 ks	48,00	39,36	0	0,00
180011	Dřv mramorová 5l - 7-14 mm	10 ks	48,00	39,36	0	0,00
180027	Keramzít - 20 l 8-16 mm	1 ks	114,50	93,99	0	0,00
180037	Keramzít - 50 l 8-16 mm (červený)	1 ks	205,60	168,59	0	0,00
180036	Keramzít - 50 l 8-16 mm (šedý)	1 ks	175,00	143,50	0	0,00
180009	Keramzít ZC produkt - 1 l 4-8 mm	12 ks	16,00	13,12	0	0,00
180010	Keramzít ZC produkt - 5 l 8-16 mm	1 ks	28,70	23,53	0	0,00
180203	Kůra dekorativní - 70 l Floria žlutá	3 ks	158,90	130,30	0	0,00
180025	Kůra mulčovací - 70 l Raš.	3 ks	90,00	73,80	0	0,00
189029	Kůra mulčovací Florcom - 70 l	3 ks	89,00	72,98	0	0,00
188030	Kůra piniová Florcom - 70 l /frakce 15-35/	3 ks	241,00	197,62	0	0,00
188031	Kůra piniová Florcom - 70 l /frakce 35-55/	3 ks	261,00	214,02	0	0,00
188032	Kůra piniová Florcom - 70 l /frakce 55-95/	3 ks	269,00	220,58	0	0,00
180013	Lávová dř - 5 l 8-16 mm	1 ks	48,00	39,36	0	0,00
180094	Mulč pro růže - 50 l Raš.	3 ks	89,50	73,39	0	0,00

Nalezeno 154 položek

**Informace o zboží**  
Katalogové číslo: 180098  
Název zboží: Cocochips - kokosové kousky 500 g

**Popis:** Přírodní, organický, kokosový substrát s hrubou strukturou pro pěstování orchidejí, bromélií a dalších epifytických rostlin. Má mimořádnou schopnost zadržovat vodu a rostliny v tomto substrátu

**Přidání do objednávky**  
Objednaný počet: 0  
Karton: 24 ks

**Informace o ceně**  
ZC Katalogová cena bez DPH: 32,00 Kč  
Moje nákupní cena bez DPH: 26,24 Kč  
Sazba DPH: 20 %  
Moje MO cena včetně DPH: 0,00 Kč  
MO Marže: -100,00 %  
MO Koefficient: 0,00

**Statistika**  
měsíc min.rok akt.rok  
červ...  
červ...  
srpen  
září  
říjen  
list...  
prosi...

Cena celkem: 0,00 Kč

ZCOnline DATA-NORMS s.r.o.

Obrázek 29: Porovnání hlavního okna nové a současné verze aplikace DN-Portál

V průběhu vývoje, v době, kdy byla dokončena základní část aplikace, byla nová verze představena potencionálním zákazníkům. Koncepce se setkala s velice příznivou odezvou. V současné době vývoj aplikace stále pokračuje, do systému jsou přidávány individualizace dle požadavků zákazníka. S nasazením do ostrého provozu se počítá v průběhu roku 2011.

## 9. Literatura

- [1] MACDONALD, Matthew, *Pro WPF in C# 2010: Windows Presentation Foundation in .NET 4.0*, Apress USA 2010, ISBN 978-1-4302-7205-2
- [2] NATHAN, Adam, *WPF 4 Unleashed*, Sams USA 2010, ISBN 978-0-672-33119-0
- [3] BEDNÁŘ, David; KRESLÍKOVÁ, Jitka. *Metodika projektového řízení vývoje softwarových aplikací* [online]. 2010-05-04  
Dostupné z: <[http://www.odbornecasopisy.cz/index.php?id\\_document=30574](http://www.odbornecasopisy.cz/index.php?id_document=30574)>
- [4] Wikipedia. *Metodologie vývoje softwaru* [online]. 2010-09-25.  
Dostupné z: <[http://cs.wikipedia.org/wiki/Metodologie\\_vývoje\\_softwaru](http://cs.wikipedia.org/wiki/Metodologie_vývoje_softwaru)>
- [5] Wikipedia. *.NET* [online] 2010-07-18.  
Dostupné z: <<http://cs.wikipedia.org/wiki/.NET>>
- [6] Microsoft Corporation. *.NET Framework 4 (webová instalační služba)* [online] 2010-10-05. Dostupné z:  
<<http://www.microsoft.com/downloads/details.aspx?FamilyID=9cfb2d51-5ff4-4491-b0e5-b386f32c0992&displaylang=cs>>
- [7] Živě.cz. *Poznáváme C# a Microsoft .NET* [online]. 2010-02-12. Dostupné z:  
<<http://www.zive.cz/clanky/poznavame-c-a-microsoft-net--1dil/sc-3-a-120978/default.aspx>>
- [8] Wikipedia. *C#* [online]. 2010-10-12. Dostupné z:  
<[http://cs.wikipedia.org/wiki/C\\_Sharp](http://cs.wikipedia.org/wiki/C_Sharp)>
- [9] Microsoft Corporation. *Modul CLR (Common Language Runtime)* [online]. 2010-10-05 Dostupné z: <<http://msdn.microsoft.com/cs-cz/library/8bs2ecf4.aspx>>.
- [10] Microsoft Corporation. *C# Language Specification 4.0* [online] 2010-10-05.  
Dostupné z:  
<<http://www.microsoft.com/downloads/en/details.aspx?displaylang=en&FamilyID=dfbf523c-f98c-4804-afbd-459e846b268e>>
- [11] Wikipedia. *Windows Presentation Foundation* [online] 2010-10-05. Dostupné z:  
<<http://cs.wikipedia.org/wiki/WPF>>
- [12] ŠTURALA, Aleš. *WPF - úvod* [online] 2010-10-05  
Dostupné z: <<http://www.vyvojar.cz/Articles/445-0-wpf-uvod.aspx>>
- [13] ŠTURALA, Aleš. *WPF - Hello 'XAML' World* [online] 2010-10-05  
Dostupné z: <<http://www.vyvojar.cz/Articles/447-1-wpf-hello-xaml-world.aspx>>
- [14] JÁNOŠÍK, Dušan. *Něco málo k technologiím WPF a Silverlight* [online]. 2010-10-05. Dostupné z: <<http://www.vyvojar.cz/Articles/473-neco-malo-k-technologiim-wpf-a-silverlight.aspx>>
- [15] 2010-10-02.  
Dostupné z: <<http://xaml.cz/wpf/databinding-a-datatemplate>>
- [16] Microsoft Corporation. *Microsoft Visual Studio* [online] 2010-10-02.  
Dostupné z: <<http://www.microsoft.com/cze/msdn/vstudio/2010>>
- [17] Wikipedia. *Microsoft Visual Studio* [online] 2010-10-02.  
Dostupné z: <[http://cs.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](http://cs.wikipedia.org/wiki/Microsoft_Visual_Studio)>
- [18] SQLite. *SQLite* [online] 2010-02-01.  
Dostupné z: <<http://www.sqlite.org>>
- [19] SQLite. *How SQLite Is Tested* [online] 2010-02-01.  
Dostupné z: <<http://www.sqlite.org/testing.html>>
- [20] SQLite. *SQL As Understood By SQLite* [online] 2010-02-01.  
Dostupné z: <<http://www.sqlite.org/lang.html>>
- [21] SQLite. *Limits In SQLite* [online] 2010-02-01.  
Dostupné z: <<http://www.sqlite.org/limits.html>>

- [22] SQLite. *SQLite Copyright* [online] 2010-02-01.  
Dostupné z: <<http://www.sqlite.org/copyright.html>>
- [23] Dwasden, *Light Database Comparison* [online] 2010-02-01.  
Dostupné z: <<http://zlog.zenware.com/?tag=microsoft-sql-server-ce>>
- [24] Wikipedia. *Webová služba* [online] 2010-02-01.  
Dostupné z: <[http://cs.wikipedia.org/wiki/Webová\\_služba](http://cs.wikipedia.org/wiki/Webová_služba)>
- [25] W3C. *Web Services Architecture* [online] 2010-02-15.  
Dostupné z: <<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>>
- [26] KOSEK, Jiří, *Využití webových služeb a protokolu SOAP při komunikaci* [online ] 2010-02-15 Dostupné z <<http://www.kosek.cz/diplomka/html/websluzby.html>>
- [27] Wikipedia. *Extensible Markup Language Architecture* [online] 2010-02-15.  
Dostupné z: <[http://cs.wikipedia.org/wiki/Extensible\\_Markup\\_Language](http://cs.wikipedia.org/wiki/Extensible_Markup_Language)>
- [28] W3C. *Extensible Markup Language (XML) 1.1 (Second Edition)* [online] 2010-02-15. Dostupné z: <<http://www.w3.org/XML>>
- [29] Wikipedia. *Simple Object Access Protocol* [online] 2010-02-15.  
Dostupné z: <<http://cs.wikipedia.org/wiki/SOAP>>
- [30] W3C. *SOAP Version 1.2 Part 0: Primer (Second Edition)* [online] 2010-02-15.  
Dostupné z: <<http://www.w3.org/TR/2007/REC-soap12-part0-20070427>>
- [31] Wikipedia. *Web Services Description Language* [online] 2010-02-15.  
Dostupné z: <<http://cs.wikipedia.org/wiki/WSDL>>
- [32] W3C. *Web Services Description Language (WSDL) 1.1* [online] 2010-02-15.  
Dostupné z: <<http://www.w3.org/TR/2001/NOTE-wsdl-20010315>>
- [33] KUŽALA, Radovan. *Konceptuální datový model* [online] 2010-08-21  
Dostupné z: <<http://owebu.bloger.cz/Databaze/Konceptualni-datovy-model>>
- [34] ŠTURALA, Aleš. *Začínáme s WPF* [online] 2010-07-12  
Dostupné z: <<http://www.vyvojar.cz/Series/3-zaciname-s-wpf.aspx>>
- [35] JIRAVA, Jaroslav, *WPF Aplikace – použití stylů a panelů* [online] 2010-09-08  
Dostupné z: <<http://xaml.cz/wpf/wpf-aplikace-pouziti-stylu-a-panelu>>
- [36] PASTOREK, Tomáš, *Konvertor - nejlepší přítel člověka 1* [online] 2010-05-12  
Dostupné z: <<http://xaml.cz/wpf/konverter-nejlepsi-pritel-cloveka-1>>
- [37] JECHA, Tomáš, *.NET Assemblies - pohled pod pokličku* [online] 2010-05-12  
Dostupné z:  
< [http://www.vbnet.cz/clanek--58-net\\_assemblies\\_pohled\\_pod\\_poklicku.aspx](http://www.vbnet.cz/clanek--58-net_assemblies_pohled_pod_poklicku.aspx) >
- [38] Wikipedia, *Refaktorování* [online] 2010-10-18.  
Dostupné z: < <http://cs.wikipedia.org/wiki/Refaktorování> >

## 10. Seznam obrázků

<i>Obrázek 1: Schéma projektového řízení</i>	8
<i>Obrázek 2: E-R model databáze současné verze DN-Portálu</i>	10
<i>Obrázek 3: První přihlášení do aplikace - Zákazník</i>	12
<i>Obrázek 4: Hlavní formulář aplikace DN-Portál (v modifikaci pro Velkoobchod ZC s.r.o.)</i>	13
<i>Obrázek 5: První přihlášení do aplikace – Obchodní zástupce</i>	16
<i>Obrázek 6: Aktivní uživatelé DN-Portálu</i>	17
<i>Obrázek 7: Formulář založení nového uživatele.</i>	17
<i>Obrázek 8: Schéma komunikace stávající aplikace DN-Portál</i>	18
<i>Obrázek 9: Struktura WPF – oddělení programové a grafické části kódu</i>	24
<i>Obrázek 10: Vývojové prostředí Microsoft Visual Studio 2010.</i>	25
<i>Obrázek 11: E-R schéma databáze použité k testování (SQLite, MS SQL CE)</i>	26
<i>Obrázek 12: Vztah základních technologií (SOAP, WSDL) webových služeb</i>	28
<i>Obrázek 13: E-R model databáze nové verze DN-Portálu</i>	33
<i>Obrázek 14: Schéma komunikace aplikace DN-Portál</i>	35
<i>Obrázek 15: Přihlášení do nové aplikace - Zákazník</i>	40
<i>Obrázek 16: Přihlášení do nové aplikace – Obchodní zástupce</i>	41
<i>Obrázek 17: Hlavní okno nové verze aplikace DN-Portál</i>	41
<i>Obrázek 18: Hlavní informace, porovnání verze pro zákazníka a obchodního zástupce</i>	42
<i>Obrázek 19: Modul Firma</i>	42
<i>Obrázek 20: Modul Akce</i>	43
<i>Obrázek 21: Modul Katalog</i>	43
<i>Obrázek 22: Modul Objednávka</i>	44
<i>Obrázek 23: Modul MOCeny</i>	45
<i>Obrázek 24: Modul Obchodní zástupce</i>	45
<i>Obrázek 25: Modul HTML Služby</i>	46
<i>Obrázek 26: Komponenta – Kategorie zboží</i>	47
<i>Obrázek 27: Komponenta – Vyhledávání</i>	47
<i>Obrázek 28: Komponenta - Katalog produktů</i>	48
<i>Obrázek 29: Komponenta – Obrázek produktu (vzhled + xaml)</i>	48
<i>Obrázek 30: komponenta – Rozšířené informace o produktu</i>	49
<i>Obrázek 31: Komponenta – Ceny produktu</i>	49
<i>Obrázek 32: Komponenta – Manipulace s produktem</i>	50
<i>Obrázek 33: Komponenta – Statistika produktu</i>	50
<i>Obrázek 34: Komponenta – Maloobchodní ceny produktu</i>	50
<i>Obrázek 35: Přidání reference na webovou službu ve Visual Studiu 2010</i>	51
<i>Obrázek 36: Zjednodušené schéma výpočtu cen – bez aplikace slev</i>	55
<i>Obrázek 37: Zjednodušené schéma aplikace slev na cenu produktu</i>	56
<i>Obrázek 38: Porovnání hlavního okna nové a současné verze aplikace DN-Portál</i>	60
<i>Obrázek 39: Princip konvertoru</i>	69



## 11. Seznam zdrojových kódů

<i>Zdrojový kód 1: Ukázka XML návratové zprávy</i>	19
<i>Zdrojový kód 2: Ukázka XML pro update skupin zboží</i>	20
<i>Zdrojový kód 3: XAML zápis</i>	24
<i>Zdrojový kód 4: SOAP request - getLogin</i>	29
<i>Zdrojový kód 5: SOAP response - getLogin</i>	29
<i>Zdrojový kód 6: WSDL ukázka - getLogin</i>	31
<i>Zdrojový kód 7: Ukázka definice a použití stylu v aplikaci</i>	39
<i>Zdrojový kód 8: XAML TreeView hierarching data binding – Kategorie zboží</i>	47
<i>Zdrojový kód 9: Přihlášení k serveru funkcí getLogin – získání token</i>	51
<i>Zdrojový kód 10: Ukázka aktualizace databázové tabulky LANG</i>	52
<i>Zdrojový kód 11: Kontrola množství – povinné minimální množství</i>	53
<i>Zdrojový kód 12: Kontrola množství – doporučené minimální množství</i>	54
<i>Zdrojový kód 13: Kontrola množství – povinné balení</i>	54
<i>Zdrojový kód 14: Kontrola množství – doporučené balení</i>	55
<i>Zdrojový kód 15: Výpočet katalogové ceny – bez aplikace slev</i>	68
<i>Zdrojový kód 16: Použití IValueConverter</i>	70
<i>Zdrojový kód 17: Použití IMultiValueConverter</i>	71

## **12. Přílohy**

Přiložené CD obsahuje:

- elektronickou podobu této bakalářské práce.
- videa s ukázkami původní a nové verze aplikace DN-Portál.

## Příloha A – Výpočet katalogové ceny

Následující část zdrojového kódu obsahuje ukázkou výpočtu katalogové ceny. Na výslednou cenu nejsou aplikovány žádné individuální slevy. V prvním kroku je nalezena aktuální platná katalogová cena (POPRIDAT), pokud je u ceny definována stupnice (id\_scale v POPRIDAT > 0) je nalezena a dohledána příslušná pozice stupnicové ceny (SCALE). Podle pozice je nalezena příslušná stupnicová cena (POPRIPOS).

```
#region Katalog Price
DataSet dataSet = new DataSet("PRICE");
SQLiteDataAdapter adapterPRIDAT = new SQLiteDataAdapter();
adapterPRIDAT.TableMappings.Add("Table", "POPRIDAT");

SQLiteCommand selectCommandPRIDAT = new SQLiteCommand(
    string.Format(@"SELECT
        *
        FROM
            POPRIDAT AS p
        WHERE
            ([id_pricelist] = @id_pricelist) AND
            ([id_product] = @id_product) AND
            ([id_currency] = @id_currency) AND
            ([date_of] <= @date_of)
        LIMIT 1").Trim(), connection);

SQLiteParameterCollection paramPRIDAT = selectCommandPRIDAT.Parameters;
paramPRIDAT.Add("@id_pricelist", DbType.Int32).Value = idPricelist;
paramPRIDAT.Add("@id_product", DbType.String).Value = _idProduct;
paramPRIDAT.Add("@id_currency", DbType.Int32).Value = idCurrency;
paramPRIDAT.Add("@date_of", DbType.DateTime).Value = String.Format("{0:yyy-MM-dd HH:mm:ss}", DateTime.Now);

selectCommandPRIDAT.CommandType = CommandType.Text;
adapterPRIDAT.SelectCommand = selectCommandPRIDAT;

try { adapterPRIDAT.Fill(dataSet); }
catch (Exception ex)
{
    Debug.WriteLine(String.Format("{0}", ex.Message), "CatalogPrice");
}
finally { adapterPRIDAT.Dispose(); }

foreach (DataRow priceRow in dataSet.Tables["POPRIDAT"].Rows)
{
    price = DNConvert.ConvertToDouble(priceRow["price"].ToString());
    rcat = DNConvert.ConvertToBool(priceRow["rcat"].ToString());
    ritem = DNConvert.ConvertToBool(priceRow["ritem"].ToString());

    Int32 idTax = DNConvert.ConvertToInt32(priceRow["id_tax"].ToString());
    Int32 iid_scale = DNConvert.ConvertToInt32(priceRow["id_scale"].ToString());
    Int32 iid_pscale = DNConvert.ConvertToInt32(priceRow["id_pscale"].ToString());

    _prices.CTaxID = idTax;
    _prices.CPrice01 = price;
    _prices.CRCat = rcat;
    _prices.CRItem = ritem;

    if (_count > 0 && iid_scale != 0 && iid_pscale != 0)
    {
        Int32 iposition = 1;

        SQLiteDataAdapter adapterSCALE = new SQLiteDataAdapter();
        adapterSCALE.TableMappings.Add("Table", "SCALE");
        SQLiteCommand selectCommandSCALE = new SQLiteCommand(
            string.Format(@"SELECT
                *
                FROM
                    SCALE AS s
                WHERE
                    ([id_scale] = @id_scale)
                LIMIT 1").Trim(), connection);

        SQLiteParameterCollection paramSCALE = selectCommandSCALE.Parameters;
        paramSCALE.Add("@id_scale", DbType.Single).Value = iid_scale;

        selectCommandSCALE.CommandType = CommandType.Text;
        adapterSCALE.SelectCommand = selectCommandSCALE;

        try { adapterSCALE.Fill(dataSet); }
        catch (Exception ex)
```

```

    {
        Debug.WriteLine(String.Format("{0}", ex.Message), "CatalogPrice");
    }
    finally { adapterSCALE.Dispose(); }

    foreach (DataRow scaleRow in dataSet.Tables["SCALE"].Rows)
    {
        for (int ii = 2; ii <= 20; ii++)
        {
            Int32 iLow = DNConvert.ConvertToInt32(scaleRow[String.Format("scale{0:00}", ii -
1)].ToString());
            Int32 iHigh = DNConvert.ConvertToInt32(scaleRow[String.Format("scale{0:00}",
ii)].ToString());

            if (_count >= iLow && _count < iHigh || iHigh == 0)
            {
                iposition = ii - 1;
                break;
            }
            else if (ii == 20)
            {
                iposition = ii;
                break;
            }
        }
    }

    // není default
    if (iposition > 1)
    {
        // Scale
        SQLiteDataAdapter adapterPRIPOS = new SQLiteDataAdapter();
        adapterPRIPOS.TableMappings.Add("Table", "POPRIPOS");
        SQLiteCommand selectCommandPRIPOS = new SQLiteCommand(
            string.Format(@"SELECT
                *
                FROM
                POPRIPOS AS p
                WHERE
                ([id_pscale] = @id_pscale) AND
                ([position] <= @position)").Trim(), connection);

        SQLiteParameterCollection paramPRIPOS = selectCommandPRIPOS.Parameters;
        paramPRIPOS.Add("@id_pscale", DbType.Int32).Value = iid_pscale;
        paramPRIPOS.Add("@position", DbType.Int32).Value = iposition;

        selectCommandPRIPOS.CommandType = CommandType.Text;
        adapterPRIPOS.SelectCommand = selectCommandPRIPOS;

        try { adapterPRIPOS.Fill(dataSet); }
        catch (Exception ex)
        {
            Debug.WriteLine(String.Format("{0}", ex.Message), "CatalogPrice");
        }
        finally { adapterPRIPOS.Dispose(); }

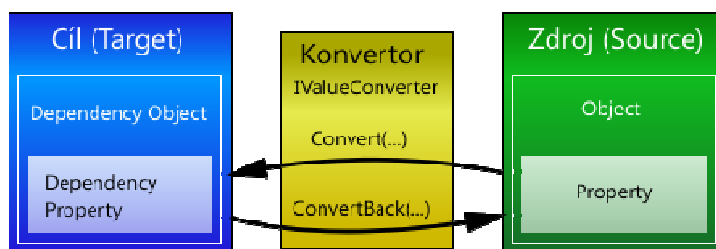
        foreach (DataRow rRow in dataSet.Tables["POPRIPOS"].Rows)
        {
            price = DNConvert.ConvertToDouble(rRow["price"].ToString());
            if (price != 0)
            {
                rcat = DNConvert.ConvertToBool(priceRow["rcat"].ToString());
                ritem = DNConvert.ConvertToBool(priceRow["ritem"].ToString());
                _prices.CPrice01 = price;
                _prices.CRCat = rcat;
                _prices.CRItem = ritem;
                break;
            }
        }
    }
}
dataSet.Clear();
dataSet.Dispose();
#endregion

```

Zdrojový kód 4: Výpočet katalogové ceny – bez aplikace slev

## Příloha B – Použití konvertoru

Tato příloha čerpá z [36]. V aplikaci DN-Portál jsem na mnoha místech použil konvertor, který usnadnil konverzi dat z jednoho typu do druhého elegantní cestou. Co to tedy je? Konvertor (Converter) je třída zděděná z interface `IValueConverter` pro jednoduchý konvertor nebo z interface `IMultiValueConverter` pro transformace vyžadující více zdrojů dat. Interface obsahuje pouze dvě metody: `Convert(...)` pro převod ze zdroje do cíle a `ConvertBack(...)` pro převod z cíle do zdroje. Činnost konvertoru je názorně vysvětlena následujícím obrázkem:



Obrázek 30: Princip konvertoru

Jako cíl DataBindingu můžeme použít pouze `Dependency Property`<sup>16</sup> (a tu může obsahovat pouze `Dependency object`). Zdroj dat pro DataBinding může být libovolná vlastnost (`Property`) jakéhokoli objektu. Mezi zdroj a cíl DataBindingu se vloží konvertor, který může data upravit.

Pro směr dat ze zdroje do cíle se používá metoda `Convert(...)`

Pro opačný směr se používá `ConvertBack(...)`

Pro názornou ukázkou jsem zvolil velmi jednoduchý příklad. Budeme chtít převést číslo produktu na obrázek patřící danému produktu. V analýze projektu jsem definoval, že každý produkt má maximálně jeden platný obrázek zboží, jehož název je shodný s identifikačním číslem zboží (`id_product`), Cílem je nalézt platný obrázek zboží a pokud obrázek existuje vrátit jej.

Logika konvertoru je tedy velmi jednoduchá:

```
[ValueConversion(typeof(BitmapImage), typeof(String))]
public sealed class ConverterImage : IValueConverter
{
    public object Convert(object value, Type targetType,
        object parameter, CultureInfo culture)
    {
        try
        {
            //break if unset
            if (value == DependencyProperty.UnsetValue)
                return null;

            FileInfo fInfo = DNFile.GetFile(ePath.PictureSource,
                String.Format("{0}.png", (string)value));

            if (fInfo.Exists)
                return new BitmapImage(new Uri(fInfo.FullName));
        }
    }
}
```

<sup>16</sup> Představuje vlastnost, která lze nastavit pomocí metody, například styl, data vazby, animace a dědičnost.

```

        else
        {
            fInfo = DNFile.GetFile(ePath.PictureSource,
                String.Format("nopicture.png"));
            if (fInfo.Exists)
                return new BitmapImage(new Uri(fInfo.FullName));
            return null;
        }
    }
    catch
    {
        return null;
    }
}

public object ConvertBack(object value, Type targetType,
    object parameter, CultureInfo culture)
{
    throw new NotImplementedException();
}
}

```

Zdrojový kód 5: Použití IValueConverter

Jako další ukázkou použití konvertoru jsem zvolil převod parametrů produktu (akce, novinka, a blokační znak) na ikonu informující uživatele o dostupnosti produktu.

```

[ValueConversion(typeof(BitmapImage), typeof(String))]
public sealed class ConverterBlockCode : IMultiValueConverter
{
    public object Convert(object[] values, Type targetType,
        object parameter, CultureInfo culture)
    {
        try
        {
            //break if unset
            for (int i = 0; i < values.Length; i++)
            {
                if (values[i] == DependencyProperty.UnsetValue)
                    return null;
            }

            int BlockCode = (int)values[0];
            bool Akce = (bool)values[1];
            bool Novinka = (bool)values[2];
            string fileName = string.Empty;
            switch (BlockCode)
            {
                case 10:
                    fileName = string.Format("{0}", "BlockCode10");
                    break;
                case 20:
                    fileName = string.Format("{0}", "BlockCode20");
                    break;
                case 30:
                    fileName = string.Format("{0}", "BlockCode30");
                    break;
                case 50:
                    fileName = string.Format("{0}", "BlockCode50");
                    break;
                case 60:
                    fileName = string.Format("{0}", "BlockCode60");
                    break;
                case 80:
                    fileName = string.Format("{0}", "BlockCode80");
                    break;
                case 85:
                    fileName = string.Format("{0}", "BlockCode85");
                    break;
                default:
                    if (Akce && Novinka)
                    {
                        fileName = string.Format("{0}", "BlockCode0_AN");
                    }
                    else if (Akce && !Novinka)
                    {
                        fileName = string.Format("{0}", "BlockCode0_A");
                    }
                    else if (!Akce && Novinka)
                    {
                        fileName = string.Format("{0}", "BlockCode0_N");
                    }
            }
        }
    }
}

```

```

        }
        else if (!Akce && !Novinka)
        {
            fileName = string.Format("{0}", "transparent");
        }
        break;
    }

    FileInfo fileInfo = DNFile.GetFile(ePath.IconSource,
        string.Format("{0}.png", fileName));
    if (fileInfo.Exists)
        return new BitmapImage(new Uri(fileInfo.FullName));
    else
        return new BitmapImage();
}
catch (Exception ex)
{
    return null;
}
}

public object[] ConvertBack(object value, Type[] targetType,
    object parameter, CultureInfo culture)
{
    throw new NotImplementedException();
}
}
}

```

Zdrojový kód 6: Použití IMultiValueConverter