

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Informační systém pro pizzerii



2016

Vedoucí práce: RNDr. Arnošt Ve-
čerka

Luděk Zavadil

Studijní obor: Aplikovaná informatika,
prezenční forma

Bibliografické údaje

Autor: Luděk Zavadil
Název práce: Informační systém pro pizzerii
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2016
Studijní obor: Aplikovaná informatika, prezenční forma
Vedoucí práce: RNDr. Arnošt Večerka
Počet stran: 33
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Luděk Zavadil
Title: Information system for pizza
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2016
Study field: Applied Computer Science, full-time form
Supervisor: RNDr. Arnošt Večerka
Page count: 33
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Hlavní cíle pro aplikaci bylo snadné ovládání, přehlednost - a to pro všechny skupiny uživatelů. V sekci pro administraci jsou vedené podrobné statistiky spolu s fakturami, které usnadňují navrhování obchodní strategie. K dispozici jsou nástroje pro správu produktů i uživatelů. V uživatelské sekci je dbáno na jednoduchost a stereotypy. Uživatel je neustále navigován celým nákupním procesem s možností případně upravit objednávku a to až do jejího konečného potvrzení.

Synopsis

The main objectives of the application were easy control and clarity across all user groups. In the administration section there are detailed statistics together with invoices which make the proposal of business strategy easier. There are tools for the management of products and users. In the consumer section, attention is paid to simplicity and stereotypes. An unsure consumer is navigated through the complete buying process via the possibility of modify until order completion.

Klíčová slova: internetový obchod, Nette, objektové relační mapování

Keywords: e-shop, Nette, object-relational mapping

Děkuji RNDr. Arnoštu Večerkovi za pomoc a odborné vedení bakalářské práce.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

1	Popis aplikace	8
1.1	Cíle aplikace	8
1.2	Průzkum a zjištění požadavků	8
1.3	Návrh vzhledu aplikace	8
1.4	Obsah Frontendu aplikace	8
1.4.1	Hlavní stránka	8
1.4.2	Stránka Pizza	9
1.4.3	Detail produktu	10
1.4.4	Stránka Objednávka	10
1.4.5	Info	10
1.4.6	Napište Nám	10
1.4.7	Kontakty	10
1.5	Obsah aplikace Backend	10
1.5.1	Objednávky	11
1.5.2	Statistiky	11
1.5.3	Produkty	11
1.5.4	Kategorie	11
1.5.5	Uživatelé	11
1.5.6	Nastavení	11
1.5.7	Další možnosti	12
1.6	Bodový systém	12
1.7	Výběr platformy	12
2	Použité technologie	13
2.1	Nette Framework	13
2.1.1	Adresářová struktura	13
2.1.2	MVP (Model-View-Presenter)	14
2.1.3	Model	14
2.1.4	View	14
2.1.5	Presenter	15
2.1.6	Zabezpečení	15
2.2	Nextas ORM	16
2.2.1	Entity	16
2.2.2	Mapper	17
2.2.3	Repozitář	17
2.3	Bootstrap	18
2.4	JQuery	19
2.5	mPDF	20
2.6	Eciovni	20
2.7	Paypal	20
2.8	Flot	20
2.9	Použité diagramy pro návrh aplikace	21
2.9.1	UC diagram pro zákazníky	22

2.9.2	UC diagram pro administrátora	23
2.9.3	ERM diagram	24
3	Uživatelská příručka	25
3.1	Vytvoření a správa uživatelského účtu	25
3.1.1	Registrace	25
3.1.2	Přihlášení	25
3.1.3	Změna osobních údajů	25
3.1.4	Zapomenuté heslo	25
3.2	Výběr produktů	25
3.3	Vytvoření objednávky	26
3.3.1	Přehled objednávky	26
3.3.2	Vyplnění údajů	26
3.3.3	Kontrola objednávky	26
3.3.4	Platba online	26
3.4	Napsat recenzi	26
3.5	Zpracování objednávky	27
3.6	Vytvoření produktu a jejich správa	27
3.7	Vytvoření kategorie jejich správa	27
3.8	Uživatelé a jejich správa	27
3.9	Nastavení prodejny	28
3.9.1	Všeobecné	28
3.9.2	Otevírací doba	28
	Závěr	29
	Conclusions	30
A	Příloha: Návod ke spuštění aplikace	31
A.1	Instalace aplikace	31
A.1.1	Databáze	31
A.1.2	Webový server	31
A.2	Spuštění aplikace	31
B	Obsah přiloženého CD	32
	Literatura	33

Seznam obrázků

1	Hlavní stránka (horní část)	9
2	Návrh vzhledu nabízených produktů	9
3	Názorná ukázka adresářové struktury	14
4	Rozvržení stránky na jednotlivé části	20
5	Přizpůsobená hlavní stránka na menší rozlišení displeje	21
6	Diagram případů užití pro zákazníka	22
7	Diagram případů užití pro administrátora	23
8	Databázový diagram	24

Seznam zdrojových kódů

1	Přehlednější kód za pomoci latte	15
2	Pro porovnání kód bez pomoci latte	15
3	Ukázka použití validačních pravidel	16
4	Zápis entity pomocí anotací	18
5	Mapper entity Orders, dotaz pomocí SQL	19
6	Mapper entity Products, dotaz nad Dbal	19
7	Vrátí entitu uživatele, který má zadanou přezdívku	20

1 Popis aplikace

1.1 Cíle aplikace

Cílem bylo vytvořit komplexní webovou aplikaci pro pizzerii (dále jen aplikace) obsahující sekci pro uživatele (frontend) ¹ a také sekci pro administraci sloužící jako informační systém (backend) ². Frontend aplikace slouží primárně jako internetový obchod pro nákup pizz a prezentaci samotné pizzerie. Backend aplikace obsahuje informace o uživateli, produktech i souhrnné statistiky.

1.2 Průzkum a zjištění požadavků

Základním cílem průzkumu bylo zjistit, jak fungují stávající webové stránky pro pizzerie a jak jsou s nimi spokojeni jak zákazníci tak i zaměstnanci. I přes značnou neochotu jak ze strany zaměstnanců tak i managementu pobočky, mi nakonec bylo odpovězeno na pár otázek.

Dotazovaní kladli důraz na přehlednost a jednoduchost. Důležité informace o prodejně či o nabízených produktech by měly být snadno dohledatelné. Stále více návštěvníků také přichází na stránky z mobilního telefonu, proto je nutnost jim stránku přizpůsobit. Dále jsem se dozvěděl, že zákazníci velmi často využívají různé akce. Občasné obměňování méně oblíbených produktů za nové.

Výše popsané návrhy byly v aplikaci zohledněny.

1.3 Návrh vzhledu aplikace

Při každém návrhu jakékoliv webové stránky, v našem případě internetového obchodu, je nutné brát ohled na zákazníka. Zákazník při výběru produktu vždy zvažuje možnosti, proto je nutné abychom mu nákup co nejvíce usnadnili a ničím ho nevyrušovali. Dále se snažíme zákazníkovi usnadnit výběr nabídkou nejprodávanějších a nejlépe hodnocených produktů. Z těchto důvodů byl zvolen jednoduchý a přehledný design s použitím principu AIDCAS³. Pro administraci webu byl použitý standardní vzhled bootstrapu 2.3, který je ověřený a přehledný pro administraci webu.

1.4 Obsah Frontendu aplikace

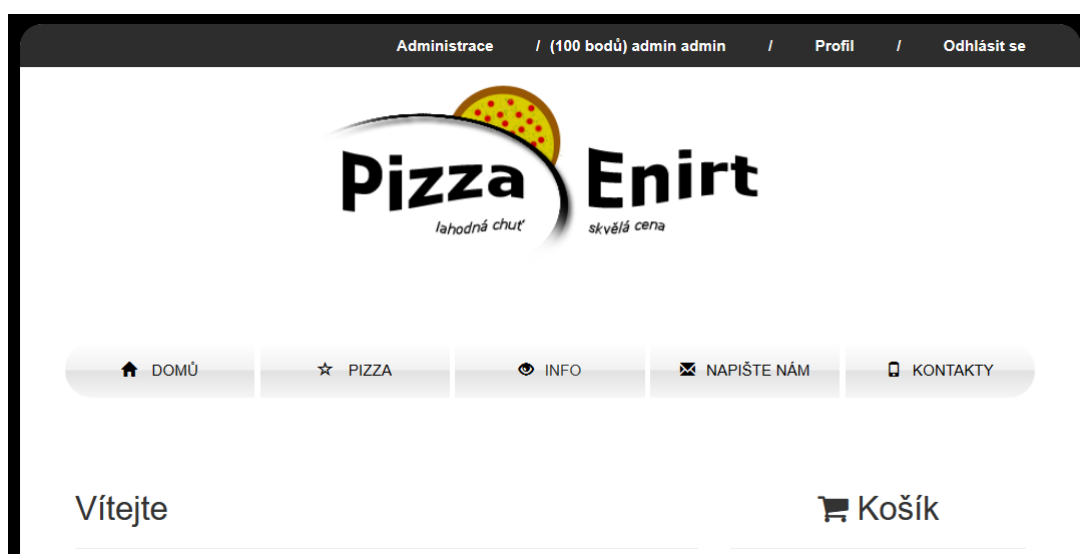
1.4.1 Hlavní stránka

Hlavní stránka je logicky uspořádaná na jednotlivé části pro snadnou orientaci. V horní liště se nacházejí tlačítka pro registraci, přihlášení a odhlášení uživatelů. Pod horní lištou se nachází logo a jméno prodejny. Následuje hlavní menu, kde

¹Frontend se označuje část webu viditelné běžným návštěvníkům.

²Backend se označuje část webové aplikace, která slouží k administraci webu a ke zpracování dat.

³zkratka z anglických slov Attention, Interest, Desire, Confidence, Action a Satisfaction, který se uplatňuje v reklamě pro získávání přízně zákazníků a motivaci nakupovat.



Obrázek 1: Hlavní stránka (horní část)

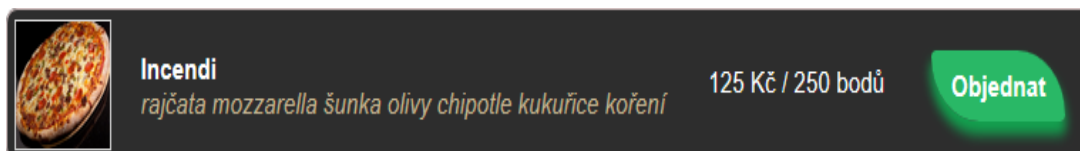
najdeme položky „Domů“, „Pizza“, „Info“, „Napište Nám“ a „Kontakty“. Výše popsané části jsou společné pro všechny podstránky.

Střední část stránky je rozdělena na dvě části. Na levé straně je uvítací text a pod ním je propagace nejlépe hodnocených produktů s možností zobrazení detailu. Pravá strana obsahuje nákupní košík s možností přejít do přehledu objednávek či přímo do nákupního procesu. Pod košíkem se nacházejí bannery ⁴. Celá pravá strana je fixní a roluje se až po patičku.

Patička stránky obsahuje dodatečné informace, ať už otevírací dobu, odkazy na sociální sítě a nejprodávanější produkty - tato část je pro každého přihlášeného uživatele individuální. V patičce se nachází také sekundární menu.

1.4.2 Stránka Pizza

Na stránce „Pizza“ se nacházejí nabízené produkty s možností filtrace či vyhledání produktu. Je zde možnost zobrazit detail produktu či zvětšeného náhledu ilustrační fotografie. Tato stránka je navržena i pro budoucí rozšíření nabízených produktů.



Obrázek 2: Návrh vzhledu nabízených produktů

⁴Banner je druhem reklamy, který se používá na webových stránkách.

1.4.3 Detail produktu

„Detail“ je stránka, která zobrazuje podrobný popis produktu s ilustrační fotografií. Dále je zde možnost produkt ohodnotit (známky jako ve škole 1 nejlepší, 5 nejhorší) a napsat hodnocení. Hodnotit produkty mohou pouze registrovaní uživatelé.

1.4.4 Stránka Objednávka

Při dokončování objednávky je nutné vyplnit všechny nezbytné údaje. Tedy způsob nákupu, kontaktní a doručovací údaje. Registrovaní uživatelé mají usnadněné vyplňování (údaje poskytli při registraci) a navíc získávají body (viz. 1.6) za objednané produkty.

1.4.5 Info

Na jednotlivých stránkách v sekci „Info“ se nacházejí užitečné informace pro zákazníky, ať už o novinkách či jak se platí body atd.. Nachází se zde i stránka „Přehled alergenů“, protože podle nařízení Evropského Parlamentu a Rady EU č. 1168/2011 je nutnost u každého prodávávaného potravinářského produktu spotřebitelům poskytnout informace o alergenních složkách a produktech, které byly použity při jejich výrobě.

1.4.6 Napište Nám

Jak už napovídá název stránky, zde je možnost zaslat vedení pobočky svůj názor, stížnost či nápad. Protože zpětná vazba od uživatelů webu je to nejdůležitější. Zde mohou zákazníci komplexně zhodnotit obdržený produkt. Tedy objednané zboží a průběh objednávky na našem internetovém portále. Vedle kvality samotných prodávávaných produktů je nezbytný také pohodlný nákup a příjemná navigace napříč webem. Zde se tak mají uživatelé (zákazníci) možnost vyjádřit, a web i nabízené produkty je tak možné upravovat k jejich spokojenosti.

1.4.7 Kontakty

Stránka „Kontakty“ je zaměřena na informace o prodejně. Nalezneme zde adresu prodejny spolu s mapou a případně kontakt. Nezbytností je také uvedení provozní doby prodejny.

1.5 Obsah aplikace Backend

Hlavní menu administrační sekce obsahuje stránky „Přehled“, „Objednávky“, „Statistiky“, „Produkty“, „Kategorie“, „Uživatelé“, „Nastavení“ a „Jít na web“. „Přehled“ slouží jako hlavní stránka mimo jiné slouží především k rychlé orientaci, jaké objednávky byly naposledy přijaty, a na kterých se už pracuje. V následujících odstavcích si jednotlivé sekce podrobně rozepíšeme.

1.5.1 Objednávky

Stránka nabízí možnost zobrazit všechny objednávky spolu s jejich stavy. Každá objednávka se nachází právě v jednom stavu. Rozlišujeme právě pět stavů:

- **Přijaté** - počáteční stav, každá právě vytvořená objednávka se nachází v tomto stavu
- **Zpracovávané** - stav ve kterém se na objednávkách pracuje
- **Dokončené** - objednávka je připravena
- **Převzaté** - objednávka je předána zákazníkovi
- **Zrušené** - slouží ke stornování chybné či pizzerií zrušené objednávky

Přesun objednávky do následujícího stavu se musí potvrdit. Ke každé objednávce lze zobrazit fakturu (jsou zde uvedeny ceny jednotlivých položek, celková cena objednávky, datum, dodavatel i odběratel).

1.5.2 Statistiky

Jednotlivé stránky statistik obsahují souhrnné informace o počtu objednávek za jednotlivé měsíce spolu s celkovou prodejní cenou. Další ze statistik ukazuje přehled, kolik bylo objednáno jednotlivých druhů produktů v daném měsíci. K jednotlivým statistickým tabulkám odpovídají grafy, ty jsou zde použity pro větší přehlednost.

1.5.3 Produkty

Zde je seznam všech nabízených produktů s cenou. Produkty lze přidávat, dodatečně upravit nebo je odebrat z nabídky.

1.5.4 Kategorie

Stránka „Kategorie“ slouží pro přidávání a upravování či smazání jednotlivých kategorií.

1.5.5 Uživatelé

Obsahuje seznam všech zaregistrovaných uživatelů s možností změnit roli uživatele. Rozlišují se tyto role: **Admin**, **Zaměstnanec** a **Zákazník**.

1.5.6 Nastavení

Stránka sloužící pro nastavení všech proměnných parametrů prodejny, kterými jsou kontaktní údaje a také otevírací doba.

1.5.7 Další možnosti

Tlačítko „Jít na web“ slouží k přesunu do Frontend části webu. „Odhlásit se“ slouží k odhlášení.

1.6 Bodový systém

Jako lákadlo pro zvýšení prodeje byl v aplikaci použit bodový systém. Zákazník sbírá body nákupem produktů, a ty může později uplatnit, aby získal slevu za nákup či nákup zdarma. U každého nabízeného produktu je cena v Kč i a cena v bodech. Body se získávají pouze za nákup za reálné peníze. Počet získaných bodů za nákup se počítá z celkové ceny.

Uplatňování bodů je v aplikaci řešeno podobným způsobem jako „**Problém batohu**“. Optimálním řešením uplatnění bodů je následující rovnice:

$$BODY(i, w) = \begin{cases} 0 & \text{pro } i = 0 \\ BODY(i - 1, w) & \text{pro } i \geq 1 \text{ a } w < w_i \\ \max\{w_i + BODY(i - 1, w - w_i), \\ BODY(i - 1, w)\} & \text{pro } i \geq 1 \text{ a } w \geq w_i \end{cases} \quad (1)$$

Kde: i - i -tý produkt objednávky i , w - limit bodů a w_i - cena i -tého produktu v bodech.

Snažíme se utratit co největší množství bodů. V případě, kdy bychom chtěli ušetřit zákazníkovi co nejvíce peněz, změnili bychom w_i za v_i (kde v_i je skutečná cena produktu i). Funkci BODY spouštíme s parametrem w jako aktuálním množstvím bodů uživatele. Jelikož je předpis funkce rekurzivní, v kódu aplikace se tak způsobem dynamického programování nachází iterativní přepis této funkce. Složitost algoritmu je úměrná nejen k počtu objednávaných produktů, ale také k počtu bodů, které daný zákazník v danou chvíli vlastní.

Proto je přidán také jednodušší algoritmus, který je aplikován v případě velkého množství bodů. Vezme se nejdražší kupovaný produkt a kontroluje se jestli ho lze zaplatit pomocí bodů. Pokud lze, přejde se k dalšímu nakupovanému produktu a takto projde celou objednávkou. Jestli v nějaké chvíli dojde do situace, že produkt nelze zaplatit pomocí bodů, na produkt se sleva vztahovat nebude. Jinými slovy body se uplatní jen na jednotlivé položky, na které je dostatek bodů k zaplacení. Body se nijak nepůlí.

1.7 Výběr platformy

Z obchodního hlediska je nutné, aby byl k aplikaci snadný přístup. Zákazník potřebuje pouze připojení k internetu a webový prohlížeč. Přiznejme si, kdo v dnešní době nepoužívá internet, ať už v mobilním zařízení, tabletu či na PC. Z těchto důvodů je webová aplikace jasná volba a to také díky multiplatformnosti. Použitými základními technologiemi tak jsou:

- Značkovací jazyk HTML (HyperText Markup Language) - pro vytvoření statické webové stránky .
- Jazyk CSS (Cascading Style Sheets) - pro pokročilejší úpravu vzhledu webové stránky (s využitím knihovny Bootstrap).
- Skriptovací jazyk JavaScript - pro vytvoření dynamických webových stránek (s využitím knihovny jQuery).
- Skriptovací jazyk PHP (rekurzivní zkratka: Hypertext Preprocessor) - používaný převážně pro tvorbu webových stránek (s využitím frameworku Nette a knihoven Nextras ORM, atd.).

2 Použité technologie

Jak už bylo zmíněno v kapitole (1.7), byl jako hlavní programovací jazyk zvolen skriptovací jazyk PHP. Pro usnadnění implementace bylo použito několik podpůrných knihoven.

2.1 Nette Framework

Základem celé aplikace je právě Nette Framework. Jedná se o webový framework⁵, který zjednodušuje návrh a implementaci webových aplikací a dále tak slouží k přehlednějšímu a strukturovanějšímu kódu. Nette Framework je free software⁶, nabízený pod licencemi GNU⁷ a BSD⁸. Využívá se pro tvorbu webových aplikací v PHP 5 a vyšším. Byl vytvořen vývojářem Davidem Grudlem. Nyní se o jeho další rozvoj stará organizace Nette Foundation. Mezi jeho přednosti patří: [1]

- moderní framework
- zaměřuje se na bezpečnost a odstranění případných chyb
- promyšlený objektový návrh využívající nových vlastností PHP 5
- jeden z nejrozšířenějších PHP frameworků v ČR
- dokumentace v češtině

2.1.1 Adresářová struktura

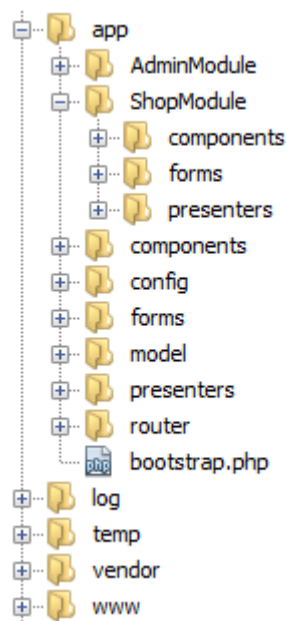
Pro jednoduchou orientaci se v Nette Frameworku využívají tzn. moduly, které rozdělí aplikaci na části, v našem případě: AdminModule a ShopModule. V Ad-

⁵Frameworky jsou knihovny, které ulehčují programování aplikace.

⁶Je software, ke kterému je k dispozici zdrojový kód, s možností ho zdarma používat, upravovat i použít ke komerčnímu využití.

⁷GNU General Public License – licence pro svobodný software.

⁸BSD Berkeley Software Distribution Licence – umožňuje volné šíření licencovaného obsahu, vyžaduje pouze uvedení autora a informace o licenci.



Obrázek 3: Názorná ukázka adresářové struktury

minModule se nachází administrační část webu, také označováno backend. ShopModule pak reprezentuje pohled pro běžného návštěvníka, také označováno frontend. Konkrétní adresářová struktura aplikace je zobrazena na obrázku 3.

2.1.2 MVP (Model-View-Presenter)

Nette je založeno na softwarové architektuře MVP, která zpřehledňuje kód aplikace a usnadňuje její budoucí rozšíření. V praxi se ukázalo, že jde o velmi užitečný přístup. MVP je nástupcem starší architektury MVC, která byla využívána zejména na terminálových systémech.

2.1.3 Model

Model zajišťuje přístup k datům (v našem případě z databáze nebo session) a manipulaci s nimi. Tedy akce uživatelů, ať už přihlášení, vložení produktu do košíku atd., to vše představuje akci modelu. Model o existenci view nebo presenteru neví.

2.1.4 View

View, tedy náhled, má na starost zobrazení výsledku požadavku uživateli ve webovém prohlížeči. View se starají o správné vykreslení stránky pomocí HTML, CSS a JavaScriptu. V Nette se využívá šablonovací systém Latte. Ten zásadně zjednodušuje zápis kódu a tím se stává celá stránka z programátorského pohledu přehlednější (viz. zdrojové kódy 1 a 2).

```

1 <tbody>
2   <tr n:foreach="$categories as $category">
3     <td>{ $category->id}</td>
4     <td>{ $category->name}</td>
5     <td>{ $category->products->count() }</td>
6   </tr>
7 </tbody>

```

Zdrojový kód 1: Přehlednější kód za pomoci latte

```

1 <tbody>
2 <?php $i = 0; foreach ($categories as $category): ?>
3   <tr>
4     <td><?php echo Nette\Templating\Helpers::escapeHtml($category->id
5     , ENT_NOQUOTES) ?></td>
6     <td><?php echo Nette\Templating\Helpers::escapeHtml(
7     $category->name, ENT_NOQUOTES) ?></td>
8     <td><?php echo Nette\Templating\Helpers::escapeHtml(
9     $category->products->count(), ENT_NOQUOTES) ?></td>
10  </tr>
11 <?php $i++; ?>
12 </tbody>

```

Zdrojový kód 2: Pro porovnání kód bez pomoci latte

Umožňuje nám v HTML kódu vypisovat PHP pomocí maker. Také je zajištěno vyhnutí se bezpečnostním rizikům jako je XSS (více o zabezpečení v kapitole [2.1.6](#)).

Dále nám Latte umožňuje vytváření a používání filtrů, které pomáhají upravit nebo přeformátovat data do výsledné podoby.

Latte také umožňuje kompozici šablon, to nám dovoluje zobrazovat část stránky na více místech bez nutnosti zbytečného opakování kódu.

2.1.5 Presenter

Reaguje na události od uživatele a zajišťuje změny v modelu a poté požádá view o vykreslení dat. Neboli akce od uživatele mohou provést změny v databázi (model), případně změnit vzhled (view).

2.1.6 Zabezpečení

Nette bere velký ohled na dokonalé zabezpečení a zaměřuje se na odstranění bezpečnostních rizik. Mezi jeden z nejčastějších útoků na webové stránky patří XSS ⁹.

⁹Cross-Site Scripting zneužívá neošetřené výstupy. Útočník pak dokáže do stránky podsunout svůj kód a tím může stránku pozměnit nebo získat důležité údaje.

Nette automaticky ošetřuje všechny výstupy pomocí své technologie Context-Aware Escaping ¹⁰. Jeho automatická aplikace je vidět na kódech 1 a 2, kde v kódu 1 se nestaráme o žádné zabezpečení výpisu dat. V kódu 2 však vidíme, že Nette (potažmo Latte) se postaralo o obalení výpisu PHP proměnných tzv. escapovacími funkcemi. Mezi další bezpečnostní rizika patří session ¹¹. Obrana spočívá ve správném nastavení, přičemž Nette se o to stará automaticky. Formuláře [2] jsou další z řady slabých míst, protože nikdy nemůžeme věřit uživateli, co zadává. Nejdůležitější je dobře ošetřit vstupy, přičemž Nette odfiltruje ze vstupů kontrolní znaky a ujistí se, že všechny textové vstupy představují validní řetězce. Z řetězců jsou odstraněny mezery jak na začátku, tak i na konci. Dále je také ošetřena možnost podstrčení jiné než nabízené hodnoty. K ochraně formulářů se také používají validační pravidla (viz. zdrojový kód 3).

```
1 $form->addText('email', 'E-mail:');
2     ->addCondition(Form::FILLED) // podmínka: pokud je e-mail
    vyplněn
3     ->addRule(Form::EMAIL, 'E-mailová adresa není platná'); // pak
    musí být platný
```

Zdrojový kód 3: Ukázka použití validačních pravidel

2.2 Nextras ORM

Nextras ORM [4] je další generací ORM ¹² navržena pro efektivní a jednoduchou práci s databází. Je velmi flexibilní a podporuje například změnu databázového systému, přejmenování názvu sloupců v databázi nebo entitě bez nutnosti úpravy těchto polí napříč celou aplikací apod. Je postavena na Nette Database. ORM je rozdělena na jednotlivé logické části Mapper, Repozitář a Entity. Tyhle části jsou popsány níže.

2.2.1 Entity

Entita je třída, jejíž instance v podstatě obsahuje data pro jeden řádek tabulky. Každé tabulce v databázi tak odpovídá jedna Entita. Ta obsahuje definici proměnných (hodnoty sloupců tabulky) a obslužné funkce. Entity se definují pomocí anotací ¹³. Zápis entit pomocí anotací je přehledný a šetří čas. Vše se definuje přes

¹⁰Jedná se o techniku zajišťující správě ošetření kódu podle umístění jeho výskytu. Např. kód bude ošetřen odlišným způsobem v případě, kdy se vyskytuje v tagu <div> a <script>.

¹¹Jedná se o uložení informací o uživateli na straně serveru. V PHP není tato služba sama o sobě dostatečně bezpečná.

¹²ORM Object Relation Model, je technika, která se používá na překlad výsledků získaných z databáze na objekty, jde to i opačně. Zjednodušuje dotazování na databázi, tím slouží k jednoduchosti a přehlednosti kódu.

¹³Anotace jsou speciálním druhem komentářů, který rozšiřuje schopnosti PHP o další funkcionalitu. S anotacemi se v Nette pracuje pomocí reflexe.

anotaci `@property`, pak následuje typ (např. `string`, `int`, `float`, `bool` atd.) a jméno. K typu můžeme přidat příznak, zda může položka obsahovat **null** (tedy prázdnou hodnotu). Nalezneme tu také podporu výčtového typu tzv. `enum` ¹⁴. ORM poskytuje velmi efektivní způsob, jak pracovat s vazbami mezi tabulkami (v ORM tedy vazbami mezi entitami). Nextras podporuje tyto základní druhy vazeb: [6].

- **1:m** – one has many: autor má více knih
- **m:1** – many has one: kniha má více autorů
- **m:n** – many has many: kniha má mnoho štítků, štítek má více knih
- **1:1** – one has one: kniha má jednoho autora

ORM podporuje také tzv. traverzování mezi vazbami entit. Tedy například v entita `Order` obsahuje referenci na entitu `User` (uživatel, který vytvořil danou objednávku). Z entity `Order` se pak k entitě `User` dostaneme pomocí konstrukce `$order -> user` a navíc načítání vazebních záznamů probíhá tzv. „lazy“ způsobem (tedy dojde k načtení až je záznam potřeba).

2.2.2 Mapper

Mapper nám umožňuje interakci s databází, respektive vykonává dotazy nad Nextras Dbal 6 nebo přímo v SQL 5. Dbal je knihovna, která pracuje s databází a poskytuje speciální dotazovací jazyk, jedná se prakticky o SQL takové, ve kterém se vytvářejí dotazy nad entitami a jejich hodnotami. Mapper také slouží jako překladová tabulka pro názvy proměnných (property) entity a databázových sloupců. Poté v případě změny názvu sloupce v databázi nebo názvu property v entitě stačí pouze zavést překlad pro tuto property. V aplikaci pak můžeme dále pracovat s původním názvem, který je však při dotazech do databáze přeložen na nový název. Při takovéto změně je pak potřeba upravit pouze SQL dotazy obsahující původní název. Dbal dotazy není potřeba upravovat, protože se pracuje nad názvy property entity, zde se tedy bude využívat stále původní označení.

SQL dotazy tvořené v Mapperu by tak měly být ojedinelé a využívat by se měly pouze v případě, kdy je potřeba vytvořit specifický SQL dotaz (využívající agregace apod.).

2.2.3 Repozitář

Repository je prostředník, který přes mapper připravuje data a tvoří z nich entity. Repozitář je již zcela nezávislý na konkrétní implementaci databáze. Zde se už SQL dotazy neprovádí, na to slouží pouze mapper. Prováděné dotazy jsou pouze nad Dbal (s využitím tzv. builderů a kolekcí) nebo s použitím předpřipravených metod:

¹⁴Enum z `enumerated` je datový typ, který je tvořen konečnou množinou pojmenovaných hodnot.

```

1 use Nextras\Dbal\Utils\DateTime;
2 use Nextras\Orm\Relationships\OneHasMany;
3
4 /**
5  * @property string $username
6  * @property string $password
7  * @property string $name
8  * @property string $surname
9  * @property string $phone
10 * @property string $email
11 * @property string $city
12 * @property string $street
13 * @property string $zip
14 * @property string $role {enum self::ROLE_*}
15 * @property boolean $newsletter {default false}
16 * @property int $points {default 0}
17 * @property DateTime $createdDate {default now}
18 * @property OneHasMany|Review[] $reviews {1:m ReviewsRepository}
19 * @property OneHasMany|Order[] $orders {1:m OrdersRepository}
20 *
21 * @property-read string $fullName {virtual}
22 */
23 class User extends Entity {
24     protected function getterFullName() {
25         return $this->name . ' ' . $this->surname;
26     }
27
28     public function getAddress() {
29         return \Model\Utils::formatAddress($this->zip, $this->city,
30             $this->street);
31     }
32 }

```

Zdrojový kód 4: Zápis entity pomocí anotací

- **get*** – vrátí jednu entitu
- **find*** – vrátí kolekci

Takže změny v databázi nevyžadují úpravy v repozitáři. Ukázka kódu v repozitáři (viz. 7).

2.3 Bootstrap

Bootstrap framework [7] je volně dostupná sada připravených knihoven, která nabízí podporu webovým technologiím HTML a CSS. Zajišťuje, aby se nám webová stránka zobrazila ve všech webových prohlížečích stejně (zvládá i starší verze prohlížečů).

Mezi další výhody patří responzivní zobrazení (viz. 5). To nám zajišťuje, že se webová stránka přizpůsobí velikosti daného displeje. Umí i rozlišovat, jestli je

```

1
2 public function getYearStatistics($year) {
3     return $this->connection->query("SELECT MONTH(created_date) AS
        month, COUNT(id) AS orders, SUM(price) AS revenue FROM {$this
        ->getTableName()} WHERE YEAR(created_date) = %i AND status =
        %s GROUP BY month ORDER BY month", $year, Order::
        STATUS_DELIVERED);
4 }

```

Zdrojový kód 5: Mapper entity Orders, dotaz pomocí SQL

```

1 public function findBestRated() {
2     $builder = $this->builder();
3     $builder->addSelect('[products.*]')
4     ->addSelect('COALESCE((SUM([r.rating]) / COUNT([r.id])), 0) AS
        [rating]')
5     ->leftJoin('products', '[reviews]', 'r', '[products.id] = [r.
        product]')
6     ->addGroupBy('[products.id]')
7     ->orderBy('[rating] DESC');
8     ...
9 }

```

Zdrojový kód 6: Mapper entity Products, dotaz nad Dbal

na výšku nebo na šířku. V situaci malého rozlišení displeje se jednotlivé prvky přeskupí, naopak s velkým rozlišením se využije prostor naplno. Vše je řešeno pomocí mřížky o 12 sloupcích (viz. obr. 4). Jak již bylo zmíněno v úvodu textu, během vývoje aplikace byl kladen důraz na použitelnost nejen na klasických PC se standardním rozlišením, ale také na tabletech a mobilních telefonech (smartphonech).

Bootstrap toho řeší více než jen responsivitu, obsahuje celou řadu předem připravených komponent (např. navbar, dropdowns, alerts, labels a mnoho dalších), které nám ušetří čas a také splňují zásady UX ¹⁵. Mnoho z těchto komponent bylo využito při vytváření této aplikace (především v administrační části).

2.4 JQuery

JQuery je javascriptová knihovna, která usnadňuje práci s javascriptem. Klade důraz na jednoduchost a přehlednost. Výrazným způsobem usnadňuje psaní javascriptového kódu.

¹⁵UX (User Experience), jednoduše na co jsou uživatelé zvyklí.

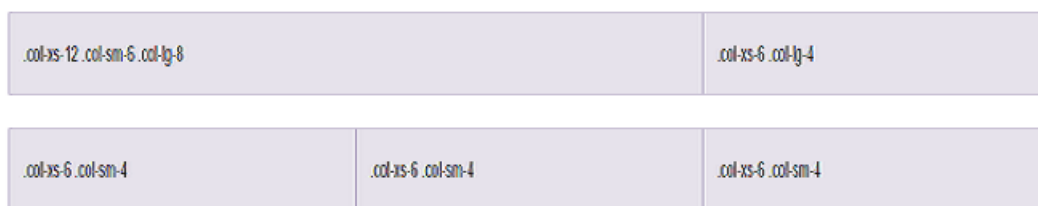
```

1 class UsersRepository extends Repository {
2
3     public function getByUsername($username) {
4         return $this->getBy(['username' => $username]);
5     }
6 }

```

Zdrojový kód 7: Vrátí entitu uživatele, který má zadanou přezdívku

Mixed: mobile, tablet, and desktop



Obrázek 4: Rozvržení stránky na jednotlivé části

2.5 mPDF

mPDF je PHP knihovna, která dokáže vygenerovat soubor ve formátu PDF (Portable Document Format) z HTML. Jelikož se jedná o internetový obchod, kde je potřeba tisk faktur, našla své uplatnění i v této práci.

2.6 Eciovní

Český programátor Ondřej Brejla je autorem tohoto pluginu ¹⁶, který umožňuje vytvářet faktury pomocí Nette šablony a knihovny mPDF (viz. 2.5).

2.7 Paypal

Paypal je jedním z nejrozšířenějších platebních nástrojů na webu. Jedná se vlastně o internetovou peněženku. Jeho popularita se v ČR značně zvyšuje. Obsluha je velmi jednoduchá. Paypal nabízí knihovny pro jeho implementaci na web. V aplikaci byl využit jako možnost online platby objednávky.

2.8 Flot

Flot je knihovna pro JQuery, která umožňuje snadné vykreslování různorodých grafů. Vizualní reprezentace je velmi názorná, proto pro zobrazování statistik (1.5.2) byl použit právě tento přístup.

¹⁶Plugin je doplněk, který rozšiřuje aplikaci o její funkčnost



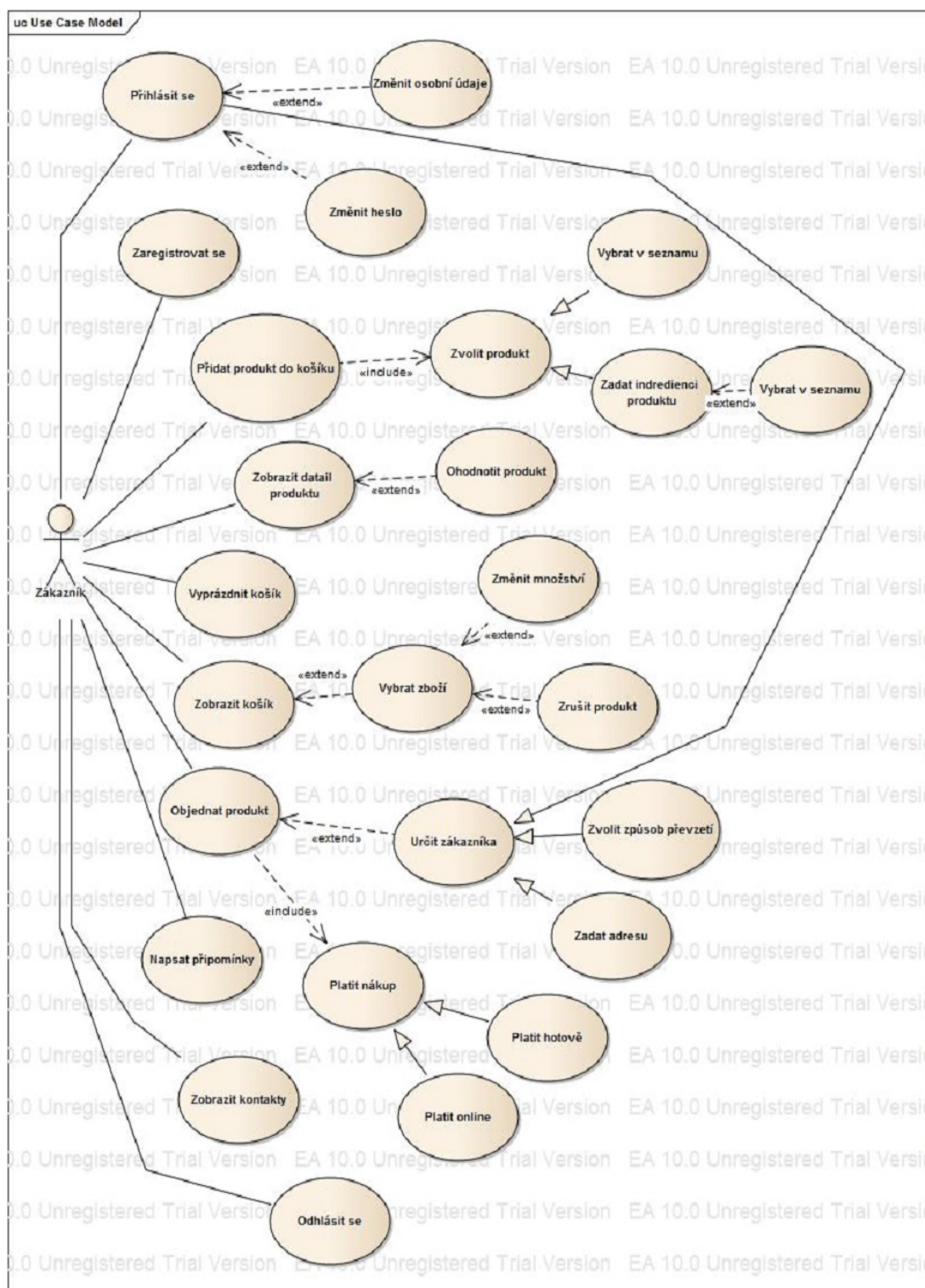
Obrázek 5: Přizpůsobená hlavní stránka na menší rozlišení displeje

2.9 Použité diagramy pro návrh aplikace

Pro názornost funkčních požadavků na systém, byly pro jednotlivé role uživatelů vypracovány UML¹⁷ diagramy, specificky „Use-Case-Model“ tzv. diagramy případu užití. Dále pro zobrazení databáze a vztahů mezi jednotlivými tabulkami byl použit „Entity-Relationship-Model“ tzv. databázový diagram.

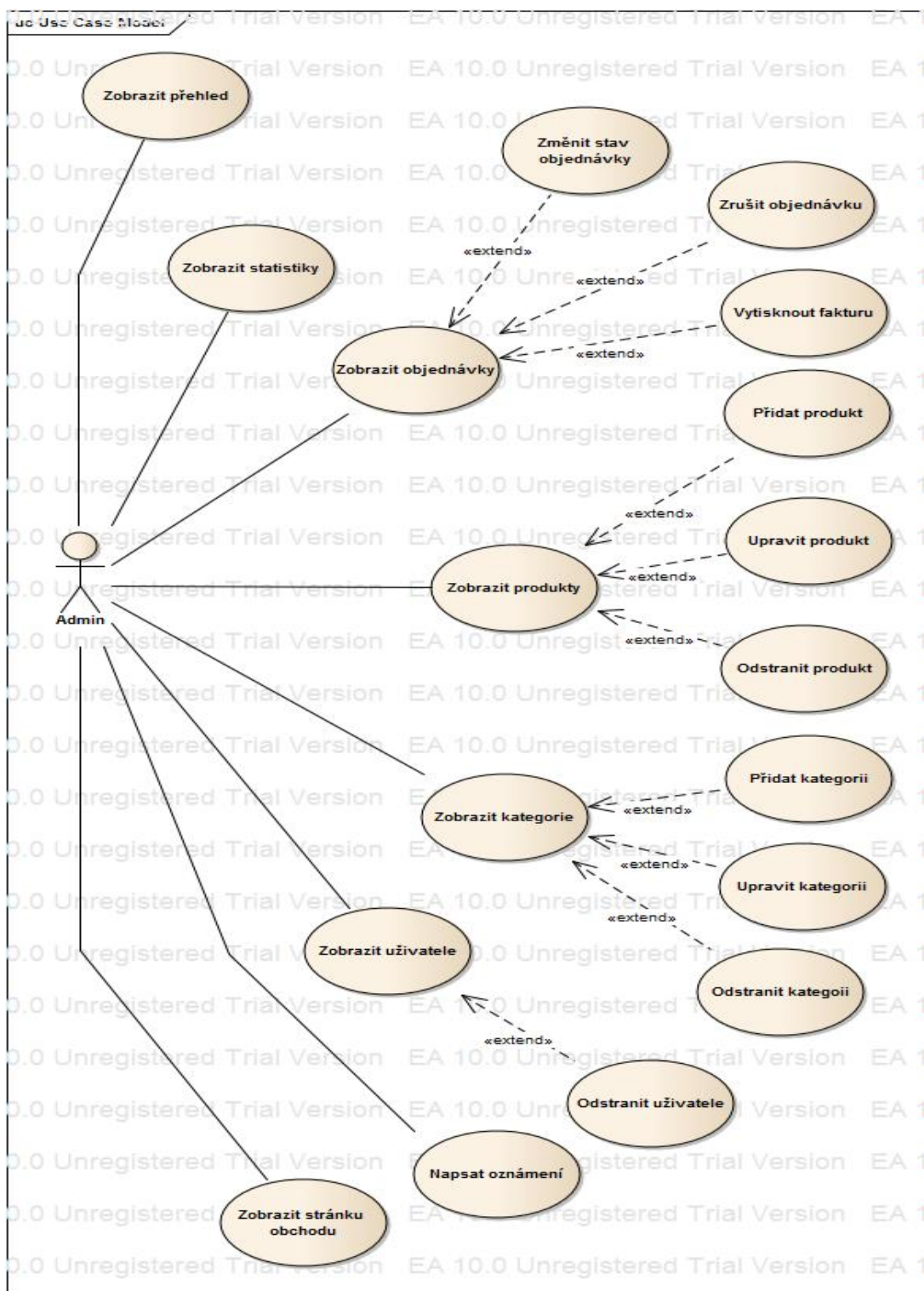
¹⁷Unified Modeling Language - grafický jazyk pro vizualizaci, specifikaci a dokumentaci

2.9.1 UC diagram pro zákazníky



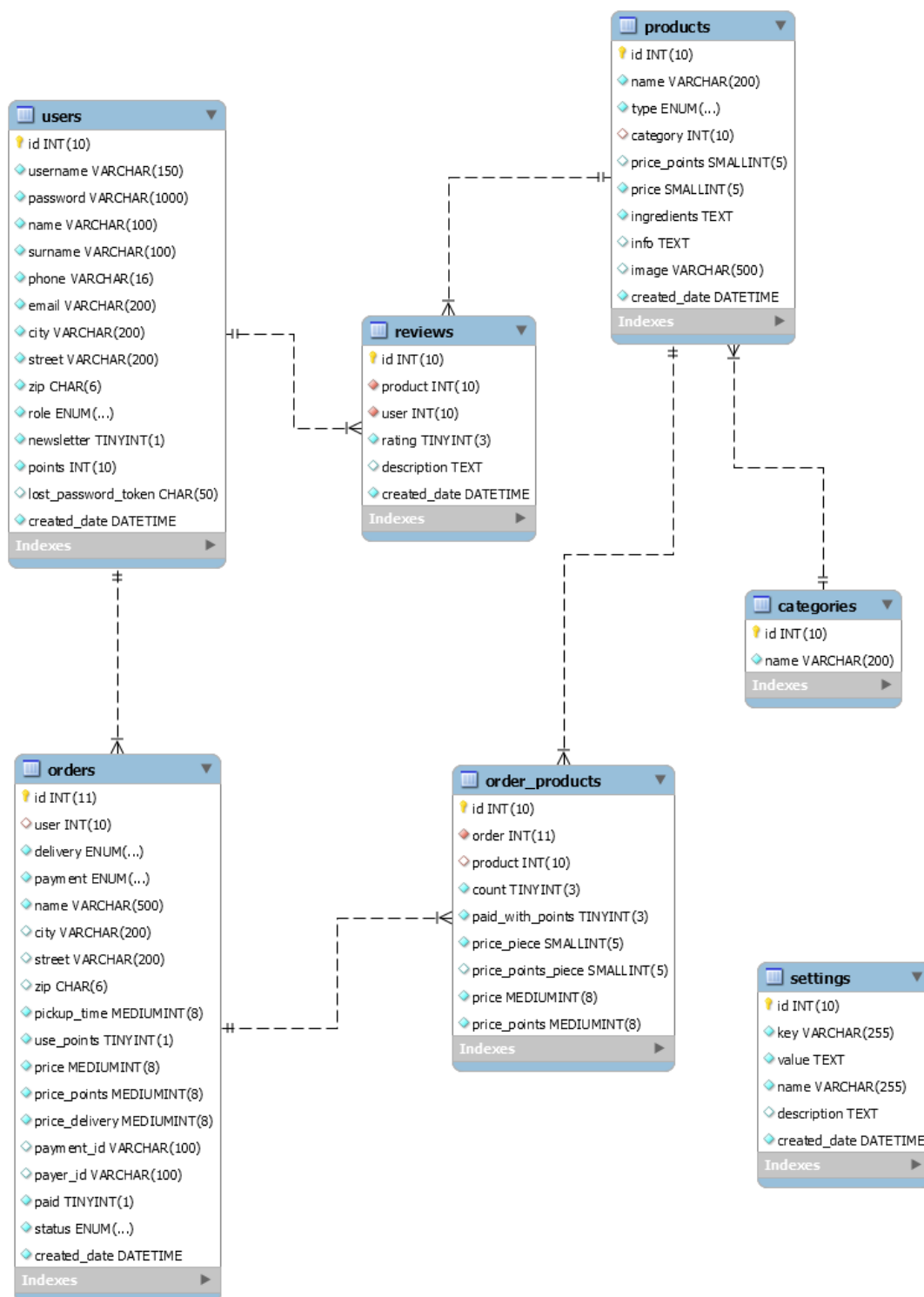
Obrázek 6: Diagram případů užití pro zákazníka

2.9.2 UC diagram pro administrátora



Obrázek 7: Diagram případů užití pro administrátora

2.9.3 ERM diagram



Obrázek 8: Databázový diagram

3 Uživatelská příručka

Aplikace slouží jako internetový obchod. Zobrazuje se ve webovém prohlížeči, přičemž ovládání je velmi intuitivní. Pro ovládání aplikace se uživatel nemusí registrovat, tím ale přichází o některé výhody.

3.1 Vytvoření a správa uživatelského účtu

3.1.1 Registrace

Tlačítko na registraci se nachází v horní liště hlavní stránky. Pro registraci je nutné vyplnit všechna povinná pole, označená pomocí symbolu „*“. Po vyplnění formuláře přijde na zadanou email adresu uvítací mail.

3.1.2 Přihlášení

Pro přihlášení je nutné vyplnit uživatelské jméno s heslem. Po přihlášení se na hlavní stránce objeví spolu se jménem a počtem nasbíraných bodů i tlačítka pro zobrazení profilu a odhlášení.

3.1.3 Změna osobních údajů

Tlačítko „Profil“ slouží ke správě uživatelského účtu. Zobrazuje uživatelské jméno, telefon, email a adresu, kdykoliv je možnost upravit zadané údaje pomocí tlačítka „Změnit osobní údaje“. Změna hesla je možná pod „Změnit heslo“. Dále v profilu nalezneme kupní historii, sloužící jako přehled zakoupených produktů.

3.1.4 Zapomenuté heslo

U přihlašovacího formuláře nalezneme tlačítko „Zapomenuté heslo“. To slouží k získání nového hesla. Po vyplnění mailu (ten, který se vyplňoval při registraci), bude na zadanou emailovou adresu odeslán mail s odkazem. Po kliknutí na odkaz se zobrazí formulář pro zadání nového hesla.

3.2 Výběr produktů

Stránka „Pizza“ zobrazuje seznam nabízených produktů. Každý produkt je zobrazen v samostatném boxu pod názvem kategorie, ve které je přiřazen. V každém boxu se nachází minimalistická ilustrační fotografie, kterou lze po najetí myši zvětšit. Dále je uveden název produktu, po kliknutí se zobrazí detail daného produktu. Pod názvem je seznam ingrediencí. Následuje cena jak v Kč tak v bodech. Posledním a nejdůležitějším prvkem boxu je tlačítko na objednání.

Pro jednodušší hledání produktů je možnost využít dvou přístupů. Pomocí vyhledávače, po zadání ingredience zobrazí pouze produkty, které ji obsahují. Nebo využít tlačítka pro zobrazení dané kategorie s produkty.

3.3 Vytvoření objednávky

Prvním krokem je vybrat požadované produkty (viz. 3.2) a pomocí tlačítka „Objednat“ je přidáme do nákupního košíku. Obsah košíku se neustále zobrazuje na pravé straně stránky s možností změnit množství či odebrat produkt z košíku. Pro přehlednost je vytvoření objednávky rozděleno na jednotlivé úseky, které spolu souvisí.

3.3.1 Přehled objednávky

Po vybrání produktu lze přejít do „Přehled objednávky“ zobrazující název produktu s počtem kusů a cenou, stále je možnost upravit množství produktů či jejich odebrání.

3.3.2 Vyplnění údajů

Následuje vyplňování údajů o objednávce. Při vyplňování objednávky je možnost zvolit způsob převzetí „Osobní převzetí“ (s možností zvolit přibližný čas vyzvednutí) či „Dovážka“ a způsob platby „Hotově“ či „Online“ 3.3.4. Při využití dovážky je nutnost vyplnit adresu doručení. Registrovaní uživatelé si zvolí kam chtějí objednávku doručit, ať už na stávající adresu (ta která se vyplňovala při registraci) či zadají novou. Dále mohou využít slevu v podobě bodů (viz. 1.6).

3.3.3 Kontrola objednávky

Poslední stránkou před dokončením objednávky je konečné shrnutí. Na levé části jsou pro kontrolu doručovací údaje, na pravé straně objednávané produkty. Při zjištění špatně vyplněných údajů je stále možnost vrátit se zpět. Posledním krokem k dokončení objednávky je nutnost souhlasit s obchodními podmínkami, které si lze zobrazit. Po kliknutí „Objednat“ je objednávka vytvořena.

3.3.4 Platba online

Platit online lze pouze přes Paypal (v budoucnu lze přidat další např. Payza, moneybookers atd.). Po dokončení objednávky a výběru platby online jsme přesměrováni na platbu pomocí Paypal. Je zde pouze slepá verze (platba je provedena bez peněžní transakce).

3.4 Napsat recenzi

Pro napsání recenze¹⁸ je potřeba vytvořit si uživatelský účet (viz. 3.1). V detailu produktu pak nalezneme „Napsat recenzi“, dále stačí zadat počet hvězd, popřípadě text (není vyžadován) a recenze se zobrazí v detailu produktu. Recenzi je možnost kdykoliv upravit.

¹⁸Recenze je text shrnující hodnocení daného produktu.

3.5 Zpracování objednávky

Každá nově vytvořená objednávka se zobrazí na stránce „Přehled“. Objednávky jsou řazeny podle data doručení. Každá objednávka se zpracovává individuálně. Na stránce „Objednávky“ jsou zobrazeny jednotlivé stavy („Přijaté“, „Zpracovávané“, „Dokončené“, „Převzaté“ a „Zrušené“, popis jednotlivých stavů viz. 1.5.1), ve kterých se objednávka právě nachází. Zobrazením jednotlivých stavů zjistíme kolik objednávek se v ní zrovna nachází. Jsou zde zobrazeny základní informace o objednatelce (výpis produktů, jméno, způsob doručení a platby, adresa objednatelce, celková cena a datum). Dále u každé objednávky jsou tlačítka na změnu stavu, stornování a zobrazení faktury. Objednávky, které přejdou do stavu „Převzaté“ nabízí už pouze možnost zobrazení faktury. Pro přehlednost jsou jednotlivé stavy vyznačeny barevně.

3.6 Vytvoření produktu a jejich správa

Na stránce „Produkty“ je pro vytvoření nového produktu tlačítko „Přidat nový produkt“. Dále vyplnit všechna povinná pole (není zde možnost zapsat cenu v bodech, ta je určena vždy z reálné ceny produktu). Pokud nevybereme ilustrační fotografii, je použita vždy univerzální, ta je pro jednotlivé typy odlišná. Po vyplnění je přidán nový produkt a zobrazí se na stránce „Produkty“. Jednotlivé produkty jsou přehledně vypsány spolu s názvem, kategorií, typem, cenou a datem vytvoření.

Pro úpravu produktů slouží tlačítko „Upravit“, které zobrazuje produkt s předchozím nastavením. Po požadovaných úpravách a uložení se produkt aktualizuje. Produkty lze také smazat po kliknutí na tlačítko „Smazat“ a produkt bude odstraněn.

3.7 Vytvoření kategorie jejich správa

Stránka „kategorie“ zobrazuje seznam dostupných kategorií spolu s počtem přiřazených produktů. Pro vytvoření nové kategorie slouží tlačítko „Přidat novou kategorii“, kde stačí pouze vyplnit název.

Pro úpravu slouží tlačítko „Upravit“ a pro odstranění kategorie tlačítko „Smazat“.

3.8 Uživatelé a jejich správa

Stránka „Uživatelé“ zobrazuje seznam uživatelů a jejich jméno, roli (administrátor, zaměstnanec a zákazník) a datum registrace. Pod tlačítkem „Upravit roli“ lze změnit roli uživatelů a tím také uživatelovo oprávnění. Smazání uživatele zde nenalezneme.

3.9 Nastavení prodejny

3.9.1 Všeobecné

Na stránce „Nastavení“ pod „Obecné“ je možnost změnit jméno prodejny, její účet či adresu.

3.9.2 Otevírací doba

Možnost změnit otevírací dobu v závislosti na konkrétním dnu. V jednotlivých dnech v týdnu lze změnit začátek a konec pracovní doby.

Závěr

Vyvinutý informační systém pro pizzerii slouží jako internetový obchod. Design je prostý a ovládání velmi intuitivní a díky tomu je aplikace přizpůsobena široké škále uživatelů.

Uživatelům je poskytnuto co největší množství informací o produktech pro usnadnění rozhodování. Vytvoření objednávky je velmi jednoduché. Celým nákupním procesem je uživatel neustále navigován. V nákupním procesu jsou uživatelům neustále zobrazovány důležité údaje o objednávce. V případě potřeby je možnost údaje kdykoliv pozměnit.

K získání zájmu zákazníků byl vytvořen bodový systém. Ten se snaží co nejvýhodnějším způsobem ušetřit zákazníkovi peníze a vyvolat zájem o další nákup.

Administrace poskytuje nástroje pro správu jednotlivých objednávek, produktů i uživatelů. Vedeny jsou také podrobné statistiky, ty napomáhají k obchodní stránce prodejny.

Aplikace je psána s ohledem na budoucí rozšíření. Jednotlivé nedostatky se projeví až uvedením do praxe.

Conclusions

A well-developed information system for a pizzeria functions as an internet shop. The design is simple and the control is very intuitive, and thanks to this the application is adaptable to a wide variety of users. The highest amount of information about the products is presented to users for easier decision making. Order creation is very simple. The user is continuously guided through the whole purchasing process.

During the purchasing process, important data concerning the order is continuously available to the user. In cases of any specific requirements, it is possible to modify the data at any time. With respect to acquiring the interest of customers, a points system was created. This endeavours to save the customer money in the most advantageous manner, and to initiate interest in further purchases.

The administration provides tools for the managing of individual orders, products, or users. Particular statistics are also kept and these aid towards the commercial side of the outlet. Any imperfections are found during operation. The possibility of expansion. The application is written with respect to future expansion. Individual faults are identified following an initial period of use.

A Příloha: Návod ke spuštění aplikace

Aplikace slouží jako internetový obchod, čili jedná se o webovou aplikaci. Na straně serveru se požaduje:

- webový server
- podporu PHP 5, minimálně verzi 5.4
- databázový server MySQL, ve verzi 5.5 nebo vyšší
- splnění podmínek, které přináší použití Nette Frameworku (ty nalezneme na: <http://doc.nette.org/cs/2.4/requirements>)

Na straně klienta se požaduje:

- moderní webový prohlížeč (webový prohlížeč ve verzi, kterou jeho vydavatel neoznačil za zastaralou), v ideálním případě Google Chrome nebo Mozilla Firefox

A.1 Instalace aplikace

A.1.1 Databáze

Data, se kterými aplikace pracuje se ukládají do relační databáze. V databázovém serveru je tedy potřeba vytvořit novou databázi a naimportovat do ní soubor **data/pizza.sql** (tato databáze má požadovanou strukturu spolu s předem předpřípravenými daty).

A.1.2 Webový server

1. Do určené složky webového serveru je potřeba překopírovat obsah archívu ve složce `bin/` na přiloženém CD.
2. Webovému serveru je třeba povolit zapisování do složek `app/temp/` a `app/log/`.
3. V souboru `app/config/config.local.neon` je potřeba nakonfigurovat lokálně závislé parametry. Jedná se tedy o vyplnění přístupových údajů do databáze (vytvořené v sekci [A.1.1](#)) a dále také přístupové údaje (konkrétně `clientID` a `secret`) k online platebnímu portálu PayPal.

A.2 Spuštění aplikace

Aplikaci spustíme přes webový prohlížeč načtením příslušné URL adresy. Načtením webu se dostaneme do Frontend sekce, přes kterou se již navigujeme pomocí menu. Do administrace webu (tzn. Backendu) se dostaneme po přihlášení a kliknutím na tlačítko „Administrace“ v záhlaví hlavní stránky (toto tlačítko je viditelné pouze oprávněným uživatelům).

Přihlašovací údaje jsou:

- přihlašovací jméno: **admin**
- heslo: **admin**

Testovací webovou stránku lze najít na:
<http://firma-cerny.cz:301/pizzaenirt/www/>.

B Obsah přiloženého CD

bin/

Kompletní adresářová struktura webové aplikace (v ZIP archivu) pro zkopírování na webový server.

doc/

Následující složka obsahuje text práce ve formátu PDF. Dále se v tomto umístění nachází archiv obsahující veškeré soubory nezbytné pro vygenerování PDF dokumentu.

src/

Kompletní zdrojové texty webové aplikace se všemi potřebnými zdrojovými texty, knihovnamí a dalšími soubory nezbytnými pro chod webového serveru.

readme.txt

Soubor popisuje minimální požadavky pro spuštění aplikace spolu s instrukcemi pro nasazení aplikace na webový server.

Navíc CD obsahuje:

data/

Složka obsahující soubor **pizza.sql**, ve kterém se nachází předem připravená data pro databázi. Dále jsou zde náhledy jednotlivých diagramů.

Literatura

- [1] *Nette Framework* [online]. 2008-2014 [cit. 2016-06-10]. Dostupné z: <https://nette.org/cs/>
- [2] *Nette Framework Forms* [online]. 2008-2014 [cit. 2016-06-10]. Dostupné z: <https://doc.nette.org/cs/2.4/forms>
- [3] *Nette Framework Forms* [online]. 2009-2014 [cit. 2016-06-10]. Dostupné z: <https://www.zdrojak.cz/clanky/nette-framework-neprustrelne-formulare-ii/>
- [4] *Nextras* [online]. 2013-2016 [cit. 2016-06-20]. Dostupné z: <https://nextras.org/>
- [5] *Nextras Orm* [online]. 2013-2016 [cit. 2016-06-20]. Dostupné z: <https://nextras.org/orm/docs/2.0/>
- [6] *Nextras Relationship* [online]. 2013-2016 [cit. 2016-06-20]. Dostupné z: <https://nextras.org/orm/docs/2.0/relationships>
- [7] *Bootstrap* [online]. 2014 [cit. 2016-06-15]. Dostupné z: <http://getbootstrap.com/>
- [8] *JQuery* [online]. 2015 [cit. 2016-06-15]. Dostupné z: <https://jquery.com/>
- [9] *mPDF* [online]. 2005 - 2016 [cit. 2016-06-15]. Dostupné z: <http://www.mpdf.com/mpdf/index.php>
- [10] *OndrejBrejla/Eciovni* [online]. 2014 [cit. 2016-06-16]. Dostupné z: <https://addons.nette.org/OndrejBrejla/Eciovni>
- [11] *Paypal* [online]. 2002 - 2016 [cit. 2016-06-16]. Dostupné z: <http://paypal.github.io/PayPal-PHP-SDK/>
- [12] *Flot: Attractive JavaScript plotting for jQuery* [online]. 2007 - 2014 [cit. 2016-06-16]. Dostupné z: <http://www.flotcharts.org/>