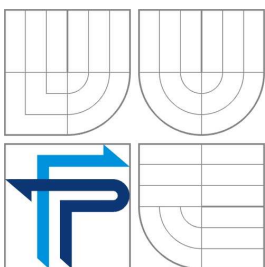


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUT OF INFORMATICS

VÝVOJ A ZAVEDENÍ INFORMAČNÍHO SYSTÉMU PRO PODPORU PODNIKOVÝCH PROCESŮ

DEVELOPMENT AND INTRODUCTION OF THE INFORMATION SYSTEM TO SUPPORT
BUSINESS PROCESSES

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. IVAN KOSTURÁK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR DYDOWICZ, Ph. D.

BRNO 2012

ZADÁNÍ DIPLOMOVÉ PRÁCE

Kosturák Ivan, Bc.

Informační management (6209T015)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává diplomovou práci s názvem:

Vývoj a zavedení informačního systému pro podporu podnikových procesů

v anglickém jazyce:

Development and Introduction of the Information System to Support Business Processes

Pokyny pro vypracování:

Úvod

Cíle práce, metody a postupy zpracování

Teoretická východiska práce

Analýza problému

Vlastní návrhy řešení

Závěr

Seznam použité literatury

Přílohy

Seznam odborné literatury:

- GILMORE, J. Velká kniha PHP a MySQL 5: kompendium znalostí pro začátečníky i profesionály. 1. vydání. Brno: Zoner press, 2005. 712 s. ISBN 80-86815-20-X.
- KOFLER, M., ÖGGL, B. PHP 5 a MySQL 5: průvodce webového programátora. 1. vydání. Brno: Computer Press, 2007. 608 s. ISBN 9788025118139.
- TVRDÍKOVÁ, M. Zavádění a inovace informačních systémů ve firmách. 1. vydání. Praha: Grada Publishing, 2000. 116 s. ISBN 80-7169-703-6
- ULMAN, L. PHP pokročilé programování pro World Wide Web. 1. vydání. Praha: SoftPress, 2002. 512 s. ISBN 80-8649-73-64.
- VYMĚTAL, D. Informační systémy v podnicích. 1. vydání. Praha: Grada Publishing, 2009. 144 s. ISBN 978-80-247-3046-2.

Vedoucí diplomové práce: Ing. Petr Dydowicz, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2011/2012.

L.S.

Ing. Jiří Kříž, Ph.D.
Ředitel ústavu

doc. RNDr. Anna Putnová, Ph.D., MBA
Děkan fakulty

V Brně, dne 23.05.2012

Abstrakt

Diplomová práca je zameraná na problematiku informačných systémov. Jej pozornosť sa zameriava na popísanie nielen poznatkov z oblasti základnej terminológie informačných systémov, definíciu predpokladov nutných pre vývoj informačného systému do podniku, ale aj jeho implementáciu a následnú správu. Praktická časť diplomovej práce vychádza z poznatkov časti teoretickej. Jej hlavná časť sa zameriava na realizáciu informačného systému.

Kľúčové slová

Systém, vývoj, implementácia, Nette, framework, PHP, HTML, CSS, MySQL, JavaScript, web, aplikácia, SWOT, E-R diagram, vývojový diagram, DFD, stavový diagram, EPC diagram.

Abstract

Master's thesis is focused on the issue of information systems. Its attention is focused on describing not only the knowledge of the basic terminology of information systems, the definition of the necessary prerequisites for development an information system within the enterprise, but also its implementation and subsequent management. The practical part of diploma thesis is based on knowledge of the theory. Its main section focuses on the realization of information system.

Key Words

System, development, implementation, Nette, framework, PHP, HTML, CSS, MySQL, JavaScript, web, application, SWOT, E-R Diagram, Flowchart, DFD, State Diagram, EPC Diagram.

Bibliografická citácia

KOSTURÁK, I. *Vývoj a zavedení informačního systému pro podporu podnikových procesů*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2012. 107 s.

Vedúci diplomovej práce Ing. Petr Dydowicz, Ph.D.

Čestné prehlásenie

Vyhlasujem, že som celú diplomovú prácu spracoval samostatne na základe uvedenej literatúry a pod vedením svojho vedúceho diplomovej práce. Vyhlasujem, že citácia použitých prameňov je úplná, a že som v práci neporušil autorské práva (v zmysle zákona č 121/2000 Zb. O autorskom práve a právach súvisiacich s autorským právom).

V Brne, 24.05.2012

Ivan Kosturák

Pod'akovanie

Dovoľujem si týmto poďakovať pánovi Ing. Petrovi Dydowiczovi, Ph.D. za odbornú pomoc, vecné rady a pripomienky pri spracovaní tejto diplomovej práce.

Obsah

Úvod	- 15 -
1 Ciele práce, metódy a postupy spracovania.....	- 16 -
2 Teoretické východiská práce.....	- 17 -
2.1 Informačný systém a jeho projektovanie	- 17 -
2.1.1 Informačný systém.....	- 17 -
2.1.2 Projektovanie informačného systému	- 19 -
2.1.3 Typológia projektov podľa miery novosti riešenia systému.....	- 19 -
2.2 Metodológia projektovania informačných systémov	- 20 -
2.2.1 Štruktúrované metódy projektovania.....	- 20 -
2.2.2 Objektovo orientované metódy projektovania a jazyk UML	- 22 -
2.3 Riadenie projektu zavedenia IS.....	- 23 -
2.4 Rozhodnutie o zavedení IS	- 24 -
2.5 Najčastejšie problémy spojené so zavádzaním IS.....	- 25 -
2.5.1 „Krabicový“ software alebo na software na zákazku	- 25 -
2.5.2 Vytvorenie vlastného IS.....	- 26 -
2.6 Procesy zavedenia IS	- 27 -
2.7 Prevádzkovanie IS.....	- 27 -
2.8 Technológie tvorby webových IS	- 28 -
2.8.1 HTML	- 28 -
2.8.2 XHTML	- 29 -
2.8.3 CSS	- 29 -
2.8.4 MySQL	- 30 -
2.8.5 PHP	- 32 -
2.8.6 JavaScript.....	- 33 -
2.8.7 AJAX	- 33 -

2.9 Framework ako prostriedok tvorby webových IS	- 34 -
3 Analýza problému	- 35 -
3.1 Predstavenie spoločnosti	- 35 -
3.2 Právna forma a organizačná štruktúra	- 35 -
3.3 Predmet podnikania	- 36 -
3.4 Súčasný IS spoločnosti IJM Group	- 37 -
3.5 SWOT analýza spoločnosti IJM Group	- 37 -
3.5.1 Silné stránky	- 37 -
3.5.2 Slabé stránky	- 38 -
3.5.3 Príležitosti	- 38 -
3.5.4 Hrozby	- 39 -
3.5.5 Závery vyplývajúce zo SWOT analýzy	- 40 -
3.6 Základné požadované charakteristiky IS	- 41 -
3.7 Požadované charakteristiky jednotlivých administrácií	- 42 -
3.7.1 Administrácia produktov	- 42 -
3.7.2 Administrácia pracovníkov	- 43 -
3.7.3 Administrácia zákazníkov	- 43 -
3.7.4 Administrácia fakturácií	- 43 -
3.7.5 Administrácia objednávok	- 44 -
3.8 Výber spôsobu zavedenia IS pre spoločnosť IJM Group	- 44 -
3.9 Výber frameworku	- 45 -
3.9.1 Porovnanie priemernej rýchlostí frameworkov	- 46 -
3.9.2 Pamäťové nároky frameworkov a ich porovnanie	- 47 -
3.9.3 Framework Nette	- 48 -
4 Vlastné návrhy riešenia	- 52 -
4.1 Rámcová charakteristika návrhu a vývoja webového IS	- 52 -

4.1.1	Databáza a jej štruktúra.....	- 54 -
4.1.2	Adresárová a súborová štruktúra	- 54 -
4.2	Návrh systémovej podpory procesu spracovania objednávok	- 56 -
4.2.1	Entitno-relačný diagram	- 56 -
4.2.2	Vývojový diagram	- 57 -
4.2.3	Diagram dátových tokov (DFD - Data Flow Diagram).....	- 59 -
4.2.4	Stavový diagram	- 60 -
4.2.5	EPC (Event-driven Process Chain) diagram.....	- 60 -
4.3	Realizácia systémovej podpory procesu spracovanie objednávok	- 62 -
4.3.1	Pridávanie produktov do košíka	- 62 -
4.3.2	Prechod do košíka (prvý krok).....	- 65 -
4.3.3	Prechod do košíka (druhý krok).....	- 69 -
4.3.4	Registračný formulár	- 70 -
4.3.5	Dokončenie objednávky	- 72 -
4.3.6	Prihlásenie a odoslanie verifikačného e-mailu	- 77 -
4.3.7	Zobrazenie užívateľského účtu	- 80 -
4.3.8	Užívateľská správa objednávok (zoznam objednávok)	- 81 -
4.3.9	Detail objednávky	- 85 -
4.3.10	Zrušenie objednávky	- 87 -
4.3.11	Zaplatenie objednávky (obecne).....	- 89 -
4.3.12	Platba pomocou služby PayPal	- 90 -
4.3.13	Platba pomocou prevodu na účet	- 95 -
4.4	Zavedenie realizovaného IS	- 98 -
4.5	Prevádzka realizovaného IS	- 100 -
4.6	Zhodnotenie realizovaného návrhu	- 100 -
4.6.1	Náklady na zavedenie IS.....	- 100 -

4.6.2	Prínosy zavedenia IS	- 101 -
4.6.3	Výsledok ekonomického porovnania zavedenia IS	- 102 -
Záver		- 104 -
Zoznam použitej literatúry		- 105 -
Prílohy		- 107 -

Zoznam obrázkov

Obrázok 1: Životný cyklus IS v podniku.....	- 18 -
Obrázok 2: Rozhodovací diagram	- 24 -
Obrázok 3: Zavedenie IS	- 27 -
Obrázok 4: Organizačná štruktúra	- 36 -
Obrázok 5: Schéma MVP	- 50 -
Obrázok 6: Integrované administračné rozhranie	- 54 -
Obrázok 7: Entitno-relačný diagram.....	- 57 -
Obrázok 8: Vývojový diagram	- 58 -
Obrázok 9: Diagram dátových tokov	- 59 -
Obrázok 10: Stavový diagram (Zdroj: vlastný)	- 60 -
Obrázok 11: EPC Diagram	- 61 -
Obrázok 12: Ilustrácia pridávania do košíka (zoznam ponúkaných produktov)	- 62 -
Obrázok 13: Ilustrácia pridávania do košíka (detail ponúkaného produktu).....	- 63 -
Obrázok 14: Ilustrácia prechodu do košíka	- 65 -
Obrázok 15: Košík (prvý krok).....	- 66 -
Obrázok 16: Košík - registračný formulár (druhý krok - užívateľ nie je prihlásený) -	70 -
Obrázok 17: Náhľad verifikačného e-mailu	- 80 -
Obrázok 18: Odkaz na „Môj účet“	- 81 -
Obrázok 19: Ovládacie prvky užívateľského konta.....	- 81 -
Obrázok 20: Zoznam objednávok.....	- 84 -
Obrázok 21: Detail objednávky	- 86 -
Obrázok 22: Zrušenie objednávky	- 88 -
Obrázok 23: Informácie o platbe	- 90 -
Obrázok 24: PayPal	- 92 -
Obrázok 25: PayPal - zrušená platba	- 92 -
Obrázok 26: Paypal - úspešná platba	- 93 -
Obrázok 27: Formulár vyhľadávania.....	- 98 -
Obrázok 28: Výsledok vyhľadávania objednávok.....	- 98 -

Zoznam tabuliek

Tabuľka 1: Vlastný vývoj IS (klady a zápory)	- 26 -
Tabuľka 2: SWOT analýza	- 40 -
Tabuľka 3: Pridávanie produktu do košíka (zdrojový kód).....	- 65 -
Tabuľka 4: Vykreslenie košíka - prvý krok (zdrojový kód)	- 69 -
Tabuľka 5: Vytvorenie registračného formulára (zdrojový kód).....	- 72 -
Tabuľka 6: Vytvorenie nového užívateľského účtu (zdrojový kód).....	- 74 -
Tabuľka 7: Uloženie objednávky do systému (zdrojový kód).....	- 76 -
Tabuľka 8: Odoslanie verifikačného e-mailu (zdrojový kód)	- 78 -
Tabuľka 9: HTML kód verifikačného e-mailu (zdrojový kód)	- 79 -
Tabuľka 10: Zoznamu objednávok (zdrojový kód)	- 83 -
Tabuľka 11: Šablóna pre detail objednávky (zdrojový kód)	- 86 -
Tabuľka 12: Zrušenie objednávky (zdrojový kód)	- 87 -
Tabuľka 13: Zrušenie objednávky (zdrojový kód)	- 88 -
Tabuľka 14: Informácií o platbe (zdrojový kód)	- 90 -
Tabuľka 15: Volanie funkcie doExpressCheckout (zdrojový kód)	- 91 -
Tabuľka 16: Údaje a PayPal platbe za objednávku	- 93 -
Tabuľka 17: Aktivovanie produktov zo zaplatenej objednávky (zdrojový kód).....	- 94 -
Tabuľka 18: Vyhľadávanie objednávok (zdrojový kód)	- 97 -
Tabuľka 19: Konfigurácia servera	- 99 -
Tabuľka 20: Odhad nákladov realizácie IS.....	- 100 -
Tabuľka 21: Úspora nákladov po zavedení IS.....	- 101 -
Tabuľka 22: Potenciálne výnosy z predaja systému	- 102 -

Zoznam grafov

Graf 1: Porovnanie priemernej rýchlosti frameworkov	- 47 -
Graf 2: Porovnanie pamäťových náročností frameworkov	- 48 -

Úvod

Neodmysliteľnou súčasťou úspešného rastu spoločnosti je spoľahlivý a efektívny IS, ktorý je schopný podporiť podnikové procesy. Spoločnosť IJM Group, s.r.o. má ambíciu rozšíriť podnikateľskú činnosť aj za hranice svojej domovskej krajiny. Bez kvalitného IS by nemalo zmysel o podobných cieľoch uvažovať. Vedenie spoločnosti si však dôležitosť kvalitného informačného systému uvedomuje a problematiku nepodceňuje.

Proces zavádzania informačného systému do spoločnosti predstavuje zložitú úlohu, ktorej úspešné zvládnutie zásadným spôsobom ovplyvňuje podnikateľskú činnosť celej spoločnosti, a preto si vyžaduje dôslednú prípravu a systematickú realizáciu.

Situácia na trhu informačných systémov, ako aj programátorské schopnosti pracovníkov spoločnosti IJM Group otvárajú širokú škálu možností realizácie projektu zavedenie podnikového IS. Zo spomenutého dôvodu je potrebná dôkladná analýza možností trhu, ako aj samotnej spoločnosti, na základe ktorej bude zvolený a prevedený najvhodnejší spôsob realizácie projektu zavádzania IS, ktorý zaručí vytvorenie IS so špecifickými požiadavkami spoločnosti IJM Group.

1 Ciele práce, metódy a postupy spracovania

Diplomová práca zrozumiteľnou formou vysvetľuje problematiku tvorby informačného systému pre potreby podniku IJM Group, s.r.o. Práca sa bude systematicky zaoberať postupom a technikami potrebnými pri návrhu, realizácii a implementácii informačného systému, čím poskytne teoretický základ potrebný pre pochopenie problematiky návrhu, tvorby a zavádzania IS.

Následným cieľom diplomovej práce je zvolenie a realizácia riešenia zavedenia informačného systému, ktoré zabezpečí maximálnu možnú zhodu predstáv o výslednom systéme a jeho reálnej podoby. Základným kameňom dosiahnutia prijateľnej miery spomínanej zhody je dôkladná analýza problému obsiahnutá v analytická časti diplomovej práce. Analýza problému využíva najmä populárnu SWOT analýzu, pomocou ktorej bola v kombinácii s metódou pozorovania vytvorená špecifikácia požiadaviek na výsledný informačný systém.

Kľúčovú úlohu diplomovej práce, ktorá je obsiahnutá v návrhovej časti práce, predstavuje samotný návrh IS a jeho následná realizácia. Diplomová práca vzhľadom na rozsiahlosť realizovaného informačného systému predstaví najprv celý systém rámcovo a následne sa bude detailne venovať jednému z hlavných podnikových procesov, ktorým je spracovanie objednávky. Pri návrhu informačného systému boli pre lepšiu zrozumiteľnosť a prehľadnosť použité diagramy ako entitno-relačný diagram, vývojový diagram, Data Flow Diagram (DFD), stavový diagram a EPC diagram.

Súhrnným cieľom diplomovej práce je vytvorenie funkčného informačného systému, ktorý bude spĺňať definované požiadavky a efektívne podporovať hlavné podnikové procesy spoločnosti IJM Group, čím zásadnou mierou prispeje k zvýšeniu konkurencieschopnosti a pravdepodobnosti úspechu spoločnosti v jej podnikateľskej činnosti.

2 Teoretické východiská práce

Nasledujúca kapitola poskytuje teoretické informácie potrebné pre pochopenie problematiky a širších súvislostí diplomovej práce.

2.1 Informačný systém a jeho projektovanie

Projektovanie predstavuje jeden z vrcholov inžinierskej tvorivej práce a jeho správne prevedenie je možné považovať za kľúčový faktor úspechu akejkoľvek komplexnej činnosti.

2.1.1 Informačný systém

Informačný systém (v ďalšom texte často ako IS) je ťažiskom diplomovej práce a preto je potrebné nastoliť terminologický základ problematiky systémov a informačných systémov.

„Systém je komplex prvků nacházejících se ve vzájemné interakci, který je charakterizovaný cílovým chováním.“¹

Podľa Molnára Z. ide za **informačný systém** považovať: *„soubor lidí, technických prostředků a metod (programů), zabezpečujících sběr, přenos, zpracování, uchování dat, za účelem prezentace informací pro potřeby uživatelů činných v systémech řízení.“²*

Informačný systém sa skladá z niekoľkých zložiek:³

- Technické prostriedky, tj. predovšetkým počítačová technika (hardware), ktorá zahrňuje rôzne počítačové systémy s periférnymi jednotkami.
- Technologické prostriedky, tj. predovšetkým programové vybavenie výpočtovej techniky (software). Technologické prostriedky sa skladajú zo systémových programov, ktoré riadia chod systému a aplikačných programov, ktoré riadia

¹ GÁLA, L. - POUR, J. - PROKOP, T. *Podniková informatika*. (2006). str. 21.

² MOLNÁR, Z. *Moderní metody řízení informačních systémů*. (1992). str. 19.

³ VLASÁK, R. - BULÍČKOVÁ, S. *Základy projektování informačních systémů*. (2003). str. 13.

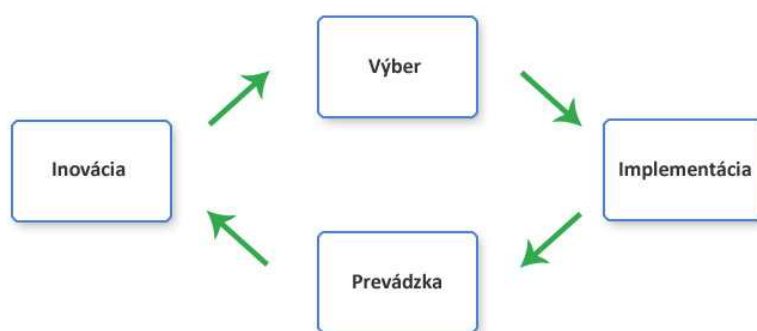
spracovateľské úlohy pri práci s dátami a komunikačných úloh systému v jeho rámci a s jeho okolím.

- Organizačné prostriedky, tzn. orgaware, predstavujú najmä legislatívny rámec, pravidlá a predpísané postupy určujúce organizácií prevádzku daného informačného systému a často aj metodické pokyny, návody, normy a pod.
- Ľudská zložka, tzn. peopleware, ktorá určuje zaraďenie, úlohy a uplatnenie človeka v rámci prevádzky informačného systému.
- Okolie systému je prostredie, v ktorom systém pracuje, z ktorého čerpá vstupy, a ktorému poskytuje výstupy svojich spracovateľských úloh. Okolie systému je tvorené najmä vonkajšími informačnými zdrojmi, ktoré do systému vstupujú, užívateľskými nárokmi a požiadavkami, technickými a inými normami, legislatívou a pod.

Je dôležité spomenúť, že informačný systém má životný cyklus skladajúci sa zo štyroch hlavných fáz:⁴

1. **výber systému** – výber riešenia spĺňajúceho požiadavky spoločnosti
2. **implementácia systému** – zavedenie IS do spoločnosti (školenia a pod.)
3. **prevádzka systému** – zabezpečenie funkčnosti systému
4. **inovácia systému** – aktualizácie systému, prípadne zmena systému

Z pohľadu spoločnosti zobrazuje životný cyklus informačného systému obrázok 1.



Obrázok 1: Životný cyklus IS v podniku⁵

⁴ BASL, J. - BLAŽÍČEK, R. *Podnikové informační systémy*. (2008), str. 216.

⁵ Upravené podľa: BASL, J. - BLAŽÍČEK, R. *Podnikové informační systémy*. (2008), str. 216.

2.1.2 Projektovanie informačného systému

Dlho predtým ako môžeme informačný systém využívať, musí prejsť celou radou vývojových etáp, ktoré v konečnom dôsledku nazývame projektovaním informačného systému.

*„Projektování je činnost, kterou se vymezuje a dokumentuje vzájemný vztah činitelů (lidského, technického a metodického) vstupujících do daného procesu a jejich působení směřující k dosažení zadaných cílů. Vzájemný vztah a působení těchto činitelů se realizuje ve třech rovinách: v čase, tj. v určité posloupnosti, v prostoru (dispozičně) a strukturálně (odpovědnostně, kompetentně).“*⁶

Projekty môžeme rozdeliť podľa typológie. Typológiou podľa miery novosti riešenia sa zaoberá nasledujúca kapitola.

2.1.3 Typológia projektov podľa miery novosti riešenia systému

Všeobecne sa rozlišujú tri kategórie, do ktorých je možné novosť v projektovaní odstupňovať:⁷

- úplne novo budovaný systém,
- prestavba súčasného systému v zásadných rysoch a parametroch,
- inovácia systému.

Či už je zámerom projektovej činnosti inovácia, prestavba alebo vývoj nového informačného systému platí, že v súčasnej dobe je preferovaný príklon k externým zdrojom. Znamená to nevyvíjať vlastný software, nevymýšľať príliš špecifické kategórie dát, pokiaľ je možné prevziať štandardizované riešenia a pod. Takmer vo všetkých prípadoch je vývoj vlastného informačného systému i vývoj na zákazku náročnejší, čo je veľmi dôležitý faktor pri rozhodovaní o ďalšom postupe. Môže sa však stať, že ponúkané produkty nevyhovujú požiadavkám spoločnosti a následne do úvahy prichádzajú dve možnosti. Prvou možnosťou je vývoj systému vlastnými silami.

⁶ HORNÝ, S. *Analýza a návrh systémů*. 1. vyd. Praha: Vysoká škola ekonomická, 1999, str. 27.

⁷ VLASÁK, R. - BULÍČKOVÁ, S. *Základy projektování informačních systémů*. (2003). str. 15.

Alternatívou je externý vývoj informačného systému na zákazku. Diplomová práca sa vo svojej praktickej časti venuje práve vývoju systému vlastnými silami. Aj napriek tomu, že sa jedná o menej štandardné riešenia, tak ako bude ukázané neskôr, v určitých prípadoch stále nachádza svoje uplatnenie.

2.2 Metodológia projektovania informačných systémov

V projektovaní informačných systémov sa v priebehu času presadili dva hlavné projektové prístupy:

- štruktúrovaný,
- objektovo orientovaný.

2.2.1 Štruktúrované metódy projektovania

Štruktúrované metódy prístupu k projektovaniu využívajú pre modelovanie systémové diagramy. Základné zložky týchto metód tvoria dáta, dátové toky, dátové súbory a tiež procesy súvisiace s dátami v systéme. Do tejto skupiny patria predovšetkým metódy:

Metóda De Marco

Metóda bola publikovaná už v roku 1978. Základné rysy tvorí využívanie modelovacej techniky pochopiteľnej užívateľom aj návrhárom. Využíva sa:

- grafická notácia,
- členenie komplikovaných procesov na jednoduchšie,
- diagramy dátových tokov,
- dátový slovník (špecifikácia dát),
- rozhodovacia tabuľka, prípadne strom.

SSADM (Structured Systems Analysis and Design)

Využívanie metódy SSADM spadá už do 80. rokov 20. storočia, keď sa stala jedným zo štandardov pre analýzu a návrh systémov. Dôraz je kladený na oddelenie logického

a fyzického návrhu systému. Na začiatku sa pracuje iba s logickými modelmi a prioritou je odstránenie neštandardných situácií a chybových stavov systému. Metodika tejto metódy sa však nezaobrá fázami realizácie, implementácie a údržby.

Metóda YSM (Yourdon Structured Method)

Yourdonova metóda modelovania je popisom skutočného chovania systému. Vychádza z predpokladu chovania, ktorý charakterizuje napr. Baný „... *chování systému zjistíme podle jeho reakce na podnět. Prioritním krokem je vytvoření esenciálního modelu - podstaty systému, podle kterého potom vytváříme softwarovou implementaci pro konkrétní prostředí.*“⁸

Projekt informačného systému sa podľa Yourdonovej metódy delí na štyri submodely:

- dátový model systému,
- funkčný model systému,
- model riadenia systému,
- model štruktúry príslušného aplikačného programového vybavenia.

Yourdon odporúča projektantom informačných systémov súbor zásad:

- Uplatniť grafické postupy pri analýze i navrhovaní vlastného riešenia celého systému aj jeho subsystémov.
- Vychádzať pri riešení z tzv. „principu Top-Down“, tj. členenie. resp. dekompozícia systému na subsystémy. Dôležité je túto systémovú štruktúru dôsledne dodržiavať v celom projekte a všetky spoločné a zhodné detaily vo vnútri jednotlivých subsystémov riešiť na úrovni celého systému.
- Minimalizovať redundanciu popisov a projektovej dokumentácie. Tomuto cieľu napomáha hlavne jednotne riešená špecifikácia v slovníku údajov (DD - Data Dictionary).

⁸ BANÝ, P a kol. *Základní metody a nástroje systémové analýzy a projektování informačních systémů*. In: Sborník Mendelovy zemědělské a lesnické univerzity v Brně, 1999, č. 2, str. 209-216.

- Snažiť sa predvídať chovanie budovaného systému do maximálnej miery. Dôraz je kladený na možné chyby a havárie.
- Dbať vo všetkých prípadoch, keď sa do styku so systémom dostáva užívateľ, na priateľské užívateľské rozhranie.

Yourdonova metóda spadá medzi metódy, ktoré definujú systém ako model, ktorý by mal obsahovať minimálne nasledovné metodologické prvky:

- vývojový diagram,
- diagram dátových tokov (DFD - Data Flow Diagram),
- inventár dátových prvkov,
- špecifikácia procesov,
- entitno-relačný diagram,
- diagram závislosti systému na vonkajších podnetoch.

Na špecifikáciu procesov môže slúžiť napríklad EPC diagram (Event-driven Process Chain), ktoré je spolu s ďalšími využitý aj v návrhovej časti diplomovej práce.

2.2.2 Objektovo orientované metódy projektovania a jazyk UML⁹

Pri objektovo orientovaných metódach projektovania sa dnes prevažne využíva jazyk UML (Unified Modeling Language), ktorý bol vytvorený v 90. rokoch. UML je v podstate jedinou používanou objektovou notáciou, ktoré je podporovaná drvivou väčšinou CASE (Computer Aided Software Engineering) nástrojov. CASE nástroje sa používajú pri tvorbe programov a slúžia na dosiahnutie vyššej kvality, bezchybnosti a udržateľnosti výsledných produktov.

Jazyk UML umožňuje pomocou rôznych typov diagramov zachytiť systém z rôznych pohľadov a na rôznej úrovni abstrakcie. Použitie správneho diagramu má vplyv na spôsoby riešenia problému. Najvýznamnejšie diagramy UML sú:

⁹ BRUCKNER, T a kol. *Tvorba informačných systémů: princípy, metodiky, architektury*. (2012). str. 303-317.

- **diagram prípadov užitia** - diagram, ktorý umožňuje popísať chovanie systému z hľadiska užívateľa. V diagrame sa špecifikuje, aké typy užívateľov používajú systém a aké činnosti vykonávajú.
- **diagram tried** - diagram, ktorý predstavuje statický pohľad na modelovaný systém. Diagram nevyjadruje zmeny v čase. Základným prvkom diagramu je trieda ako abstrakcia prvkov s rovnakými vlastnosťami.
- **sekvenčný diagram** - diagram, ktorý graficky zachytáva priebeh spracovania v systéme v podobe zasielania správ. Správy si väčšinou posielajú objekty, pričom základnou vlastnosťou objektu je jeho schopnosť správu prijať a reagovať na ňu. Reakcia na správu znamená, že objekt spustí nejakú metódu. V rámci tejto metódy môže objekt poslať správu ďalšiemu objektu a tak vzniká sekvencia správ, ktorá sa zachytáva práve sekvenčným diagramom.
- **diagram aktivít** - diagram, ktorý umožňuje zachytiť postupnosť aktivít prebiehajúcich sekvenčne alebo paralelne.

2.3 Riadenie projektu zavedenia IS

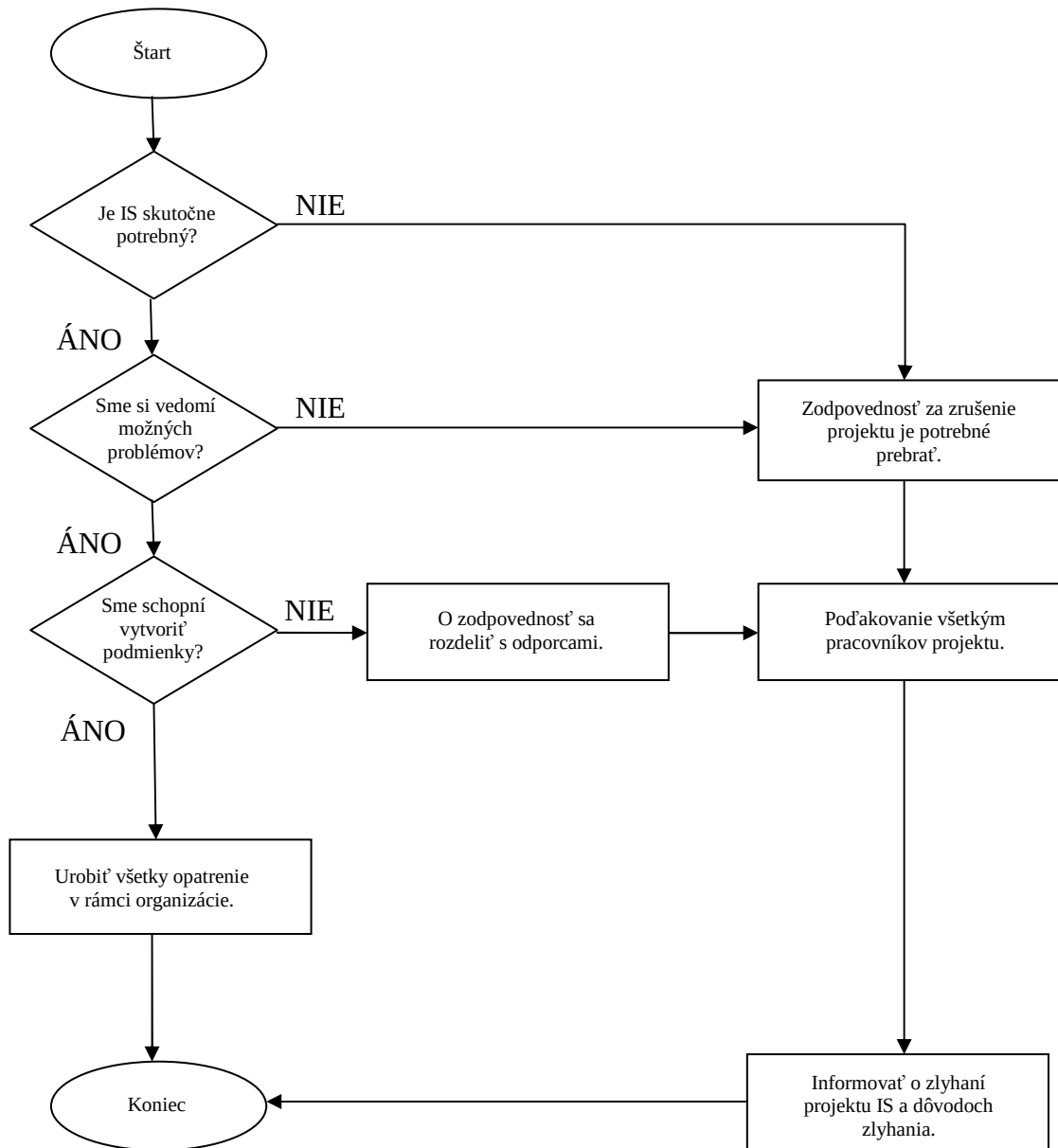
Projekt môžeme z informačného hľadiska rozdeliť na 3 fázy (príprava IS, zavedenie IS, prevádzkovanie IS), ktoré úzko súvisia so životným cyklom IS, ktorý bol spomenutý v predošlej časti diplomovej práce. Projekt IS môžeme chápať nasledovne:¹⁰

- má pevne a jasne stanovený cieľ (zmena časti IS, zmena celého IS a podobne),
- má určený začiatok a koniec,
- spravidla sa vyznačuje obmedzenými zdrojmi (finančnými i ľudskými),
- vyznačuje sa stupňom rizika (nebezpečenstvo oneskorenia a rastu nákladov),
- málokedy sa opakuje (z pohľadu odberateľa),
- u dodávateľa je situácia opačná, pretože ten často ponúka celú škálu metodík zavádzania a realizácie projektu.

¹⁰ VYMĚTAL, D. *Informační systémy v podnicích*. (2009). str. 39 - 40.

2.4 Rozhodnutie o zavedení IS

Rozhodnutie o zavedení IS je podstatný krok, ktorý predchádza samotnému zavedeniu IS a zobrazuje ho diagram na obrázku 2.



Obrázok 2: Rozhodovací diagram¹¹

¹¹ Upravené podľa: VRANA, I. - RICHTA, K. *Zásady a postupy zavádění podnikových informačních systémů*. (2005). str. 16.

2.5 Najčastejšie problémy spojené so zavádzaním IS

Nasledujúca kapitola sa venuje problematike typických komplikácií pri výbere informačného systému.

2.5.1 „Krabicový“ software alebo na software na zákazku

V prípade, že sa spoločnosť rozhodne nevyvíjať vlastný software, ponúkajú sa jej dve základné možnosti, ako si zabezpečiť žiadaný systém. Možnosťami sú „krabicový“ software alebo software na zákazku, ktorých porovnanie bolo vypracované pomocou práce Bébra a Douceka, vydané v publikácii *Informační systémy pro podporu manažerské práce*.¹²

Využitie „krabicového“ riešenia zabezpečuje potreby veľkého množstva užívateľov. Riešenie je lacné a rýchle. Problémom však je, že nie je možné, aby existoval produkt, ktorý by vyhovoval všetkým užívateľom. „Krabicový“ rieši rozličné preferencie možnosťou kustomizácie - úpravy podľa špecifických požiadaviek užívateľa (postupy práce, klávesové skratky, normy a predpisy, vzhľad a pod.). Nastávajú však situácie, kedy ani možnosť kustomizácie nepostačuje.

Po softwari na zákazku nesiahajú iba väčšie spoločnosti. Software na zákazku sa využíva najmä v prípadoch, keď nie je možné na trhu nájsť vyhovujúce „krabicové riešenie“. Vhodnými ukážkami softwaru na zákazku sú niektoré verejné systémy štátnej správy a rôzne systémy pre vedecké výpočty, riadenie procesov, simulácie a pod. Nevýhodou individuálneho riešenia je jeho finančná, prípadne časová náročnosť. Ďalším negatívom je závislosť na dodávateľovi. Napríklad v prípade, keď je potrebné po čase systém upraviť, môže sa stať, že dodávateľ už nie je k dispozícii a celý systém je potrebné vytvoriť znovu. Nie je možné zabudnúť ani na zvýšený stupeň ohrozenie citlivých interných informácií podniku, keďže je o nich potrebné oboznámiť dodávateľa, aby mohol plnohodnotne vykonať svoju úlohu.

¹² BÉBR, R. - DOUCEK, P. *Informační systémy pro podporu manažerské práce*. (2005). str. 96.

Najčastejšie sa spoločnosti prikláňajú k možnosti využitie „krabicového“ riešenia. Ideálne od prestížnej dodávateľskej spoločnosti. Žiadané sú maximálne možnosti kustomizácie. V prípade, že potrebný produkt nie je na trhu dostupný (prípadne implementácia by bola neúmerne nákladná), je vhodné siahnuť po zákazkovom softwari. Základom zadávania zákazky je príprava dôkladného zadania a výber spoľahlivého a trhom overeného dodávateľa softwaru.

2.5.2 Vytvorenie vlastného IS

V situácií, že sa spoločnosť rozhodne pre vytváranie vlastného IS je dôležité, aby si uvedomovala klady a zápory zvoleného riešenia, ktoré obsahuje tabuľka 1.

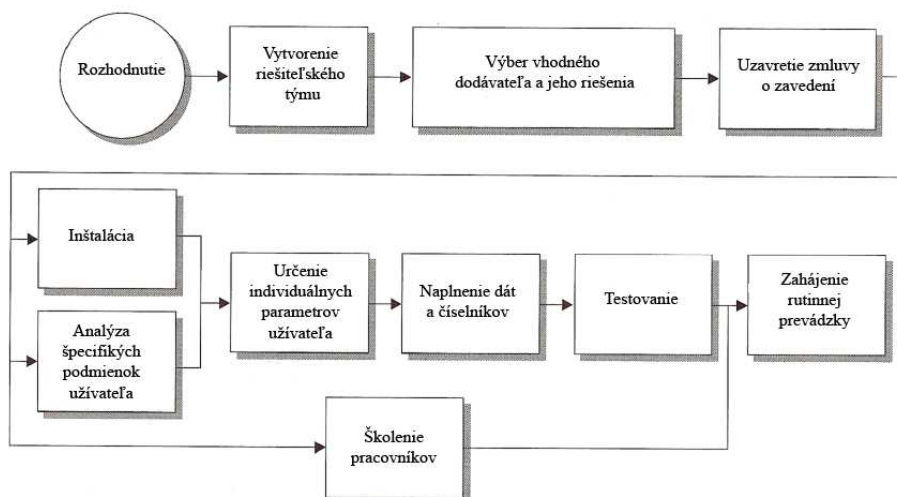
Vlastný vývoj IS	
Klady	Zápory
• IS šitý na mieru potrebám firmy, • možnosť rastu IS podľa potrieb firmy, • detailná znalosť prevádzkovaného IS, • konkurencia nepozná silné a slabé stránky prevádzkovaného IS, • dodávateľ neodhalí stratégiu firmy, • ľahká reakcia na potreby užívateľov.	• vysoké náklady, • časová náročnosť, • obvykle nižšia kvalita IS, spôsobená nie vždy špičkovou kvalitou interných riešiteľov, • značné riziko nekonzistencie systému pri fluktuácií riešiteľov, • kooperatívna náročnosť (budovanie vzťahy so subdodávateľmi).

Tabuľka 1: Vlastný vývoj IS (klady a zápory)¹³

¹³ Upravené podľa: TVRDÍKOVÁ, M. *Zavádění a inovace informačních systémů ve firmách*. (2000). str. 53.

2.6 Procesy zavedenia IS

Počas zavádzania informačného systému sa často môže vykonávať úprava spôsobu komunikácia a celej podnikovej kultúry. Zmeny sa zvyčajne prejavujú v množstve procesov. Najdôležitejšie procesy zavedenia informačného systému ukazuje obrázok 3.



Obrázok 3: Zavedenie IS¹⁴

2.7 Prevádzkovanie IS

V snahe zabezpečiť optimalizácie IS je potrebné zamerať sa aj na riadenie následnej prevádzky. Samotné riadenie implementácie nestačí. Príčinou sú zmeny v technológiách, ktoré za sebou častou nesú aj potreby obmeny metodík. Dve známe metódy riadenie prevádzky sú:

ITIL

IT Infrastructure Library (knihnica infraštruktúry IT) obsahuje súbory pre riadenie IT služieb. Spomínané súbory rozoberajú, čo je potrebné vykonať. O následnej forme prevedenia už rozhoduje konkrétna spoločnosť. ITIL predstavuje návod na prevádzkovanie IS. Súčasťou ITIL sú aj konzultácie a rôzne školenia.

¹⁴ Upravené podľa: BASL, J. - BLAŽÍČEK, R. *Podnikové informační systémy*. (2008), str. 194.

COBIT

Control Objectives for Information and Related Technology predstavuje komplexný systém cieľov a metrík podnikovej informatiky, ktorý poskytuje jednoliaty pohľad na podnikovú informatiku so zameraním na jej riadenie. Úlohou COBIT je odpovedať na požiadavky spoločnosti a súčasne si zachovať nezávislosť na technológii, ktorú podnik pre svoju činnosť využíva.

2.8 Technológie tvorby webových IS

Nasledujúca kapitola sa zameriava na technológie tvorby webových IS. Dôraz je predovšetkým kladený na HTML/XHTML, PHP, PHP, JavaScript a stále častejšie používaný AJAX a väzby medzi nimi. Kombináciou spomínaných prvkov je možné vytvoriť plnohodnotný, interaktívny IS. Majoritná časť kapitoly, ktorá je doplnená o rôzne parafrázy, bola vypracovaná na základe práce publikovanej v monografií AJAX a PHP tvoříme interaktivní webové aplikace PROFESIONÁLNĚ od autorov Darie a kolektív.¹⁵

2.8.1 HTML

HyperText Markup Language (HTML) je štandardným typom dokumentu využívaným na internete. HTML pozostáva zo značiek, ktoré dokáže spracovávať webový prehliadač. Pri spracovávaní HTML značiek dochádza k ich analyzovaniu („parsovaniu“) a následnému zobrazeniu. Návrh HTML nepočítal s tvorbou zložitých webových aplikácií s interaktívnym správaním a slúži na popis formátovania dokumentu a jeho obsahu, ktorý je zložený zo statického textu a obrázkov. V prípade, že je potrebné zobrazit' HTML stránku, dochádza k iniciovaniu jej načítania a pred vykonaním samotnej požiadavky, musí táto stránka existovať na určitom mieste v podobe statického dokumentu.

Na sprístupňovanie webových stránok slúži HyperText Transfer Protocol (HTTP). Dvojica prvkov HTML a HTTP vytvára základ internetu ako je v súčasnosti chápaný. Ich kombinácia je však vzhľadom na moderné nároky vysoko nedostatočná, keďže

¹⁵ DARIE, C. a kol. *AJAX a PHP tvoříme interaktivní webové aplikace PROFESIONÁLNĚ*. (2006).

umožňuje zobrazovať iba statický obsah webových stránok. Práve kvôli potrebe dynamického, interaktívneho a užívateľský prívetivého obsahu webových stránok vzniknú ďalšie technológie, ktoré naďalej využívajú protokol HTTP, avšak s dátami pracujú dynamicky. Dynamická práca s dátami umožňuje v porovnaní so statickými webovými stránkami rozsiahlu funkcionality, čo umožnilo vytvárať plnohodnotné webové aplikácie a informačné systémy.

2.8.2 XHTML

XHTML (Extensible Hypertext Markup Language) je podobne ako HTML značkový jazyk. Syntax XHTML je však prísnejšia a jazyk je rozšíriteľný o ďalšie značky. XHTML vychádza z jazyka XML (eXtensible Markup Language). Počas tvorby XHTML dokumentu je dôležité dbať na nasledujúce pravidlá:

- Špecifikácia DOCTYPE je potrebná.
- Všetky značky musia byť napísané podľa pravidiel XML.
- Všetky značky musia byť uzatvorené.
- Značky musia byť napísané malými písmenami.
- Hodnoty parametrov musia byť v úvodzovkách.
- Všetky parametre musia mať hodnotu.

2.8.3 CSS

Jazyk CSS (Cascading Style Sheets) umožňuje oddeliť vizuálnu prezentáciu dokumentu od jeho obsahu, ktorý je najčastejšie vytvorený pomocou HTML, od jeho obsahovej časti. Oddelenie obsahu od vizuálnej prezentácie prináša väčšiu prehľadnosť zdrojového kódu, vyššiu flexibilitu a kontrolu nad špecifickými vlastnosťami prezentácie. CSS predstavuje užitočný nástroj zdieľania formátovania pre viaceré podstránky a znižuje redundantnosť použitého kódu. V neposlednom rade je potrebné spomenúť, že pomocou CSS je možné stránku štýlovať podľa zariadenie, na ktorom sa zobrazuje (mobilný telefón, monitor a pod.) Technológia CSS je v súčasnosti štandardom, ktorý sa pri vytváraní webových stránok využíva takmer bezpodmienečne.

2.8.4 MySQL

MySQL (My Structured Query Language) je relačná databáza. MySQL vychádza z deklaratívneho programovacieho jazyka SQL (Structured Query Language) a je typu DBMS (database management system).

Architektúra databázy MySQL má široké uplatnenie a je vhodná pre riešenia rôznych úloh. Architektúra MySQL je dostatočne flexibilná a vhodná aj pre riešenie náročných projektov, ako sú napríklad webové IS. MySQL môže byť vhodnou voľbou pri dátových skladoch (data warehouses), online spracovaní transakcií (online transaction processing), software pre indexovanie obsahu, redundantných systémoch s vysokou dostupnosťou a iných.

Flexibilitu MySQL je možné vidieť vo viacerých ohľadoch. Možnosti konfigurácie umožňujú, aby MySQL správne fungovalo na rôznom hardwari. Súčasťou je podpora viacerých dátových typov. Najtypickejším a najpodstatnejším znakom MySQL je jeho architektúra úložných enginov (storage engines), ktorých návrh zaručuje oddelenie vykonávania dotazov a ďalších úloh servera od získavania a ukladania dát. Spomínaná separácia funkcií dovoľuje na úrovni jednotlivých tabuliek určovať, ako sa budú dáta ukladať, aké funkcie, výkon a ďalšie charakteristiky budú pre konkrétny prípad využité.

Každému klientskemu pripojeniu v MySQL je vo vnútri serverového procesu priradené jeho vlastné vlákno (thread). Následne sa dotazy dotyčného pripojenia vykonávajú v rámci jedného vlákna, ktoré je zdieľané na jednom jadre alebo procesore. Činnosť urýchľuje cache pamäť, v ktorej si server uchováva vlákna, a preto sa nemusia vytvárať a ukončovať pri každom novom pripojení. V prípade, že sa klient (aplikácia) chce spojiť s MySQL serverom, musí prebehnúť autentizácia. Zloženie autentizácie je nasledovné:¹⁶

- užívateľské meno,
- hostiteľ,

¹⁶ GILMORE W. J. *Velká kniha PHP a MySQL 5: kompendium znalostí pro začátečníky i profesionály*. (2005).

- miesto pôvodu klienta,
- heslo.

Ďalšou možnosťou je použitie rôznych certifikátov cez spojenie SSL (Secure Sockets Layer). Po úspešnom pripojení klienta dochádza k overovaniu každého dotazu, ktorý bol klientom vydaný. Zisťuje sa, či má klient potrebné oprávnenie pre akciu, ktorú sa snaží vykonať.

V snahe optimalizovať svoju činnosť MySQL vykonáva rozbor dotazov. Výsledkom rozboru je interná stromová štruktúra (parse tree), na ktorú sa potom optimalizácia aplikuje. Určuje sa poradie čítania tabuliek, indexy a pod. Ešte pred rozborom dotazov sa prechádza cache dotazmi (query cache), kde sú uložené príkazy SELECT a ich výsledky. V prípade, že je vydaný dotaz, ktorý je zhodný s dotazom nachádzajúcim sa v cache server, už nemusí vykonávať žiadny rozbor a dokonca ani dotaz, pretože ako výsledok vráti už uloženú informáciu.

V MySQL existujú tzv. pokyny (hints), čo sú špeciálne kľúčové slová, ktoré ovplyvňujú rozhodovací proces optimalizácie. Server je rovnako možné požiadať o vysvetlenie jeho vlastných optimalizačných rozhodnutí pomocou príkazu EXPLAIN. Získané informácie napomáhajú čo najefektívnejšiemu chodu databázy.

MySQL sa používa pod voľne šíriteľnou GPL licenciou, alebo pomocou komerčnej Enterprise edície. MySQL má pomerne jednoduchú inštaláciu a na spravovanie databázy sa používa aplikácia phpMyAdmin, ktorá slúži na vytváranie, upravovanie, vymazávanie databáz, export a import dát a správu užívateľov. Funkčnosť aplikácie zvyčajne zabezpečuje server Apache s podporou PHP.¹⁷

Pri databáze je dôležité spomenúť aj normalizáciu jej relácií, ktorá sa využíva najmä v transakčných databázach. Práve s transakčnou databázou bude diplomová práca

¹⁷ KOFLER, M a ÖGGL, B. *PHP 5 a MySQL 5: príručka webového programátora*. (2007), str. 233.

narábať vo svojej návrhovej časti. V rámci normalizácie sú známe štyri normálne formy:¹⁸

- **Prvá normálna forma** - relácia je v prvej normálnej forme, ak sú všetky atribúty entity jednoduché (atomické).
- **Druhá normálna forma** - relácia je v druhej normálnej forme, ak sa nachádza v prvej normálnej forme a navyše sú všetky jej atribúty závislé na celom kandidátnom (primárnom) kľúči.
- **Tretia normálna forma** - relácia je v tretej normálnej forme, ak sa nachádza v druhej normálnej forme a navyše sú všetky jej neklúčové atribúty navzájom nezávislé.

2.8.5 PHP

Webové technológie súčasnosti umožňujú na strane servera viac ako poslanie HTML súboru klientovi, ako napríklad zložité kalkulácie objektovo orientovaného programovania, využívanie databáz a ďalšie. Nárast funkcionality a úrovne spracovania dát na strane servera sú jednými z hlavných dôvodov úspechu internetu.

PHP je jednou z technológií, ktorá slúži na zavedenie logiky na strane servera. PHP je určený na tvorbu dynamických webových stránok. PHP však nie je jediná technológia pre programovanie na strane servera. Ďalšími konkurenčnými technológiami sú napr. ASP.NET, Java Server Pages (JSP), Perl a ďalšie.

Pre fungovanie jazyka PHP je potrebný webový server, na ktorom sa budú skripty vykonávať. Výhodou je, že webový server je možné spustiť na najpoužívanejších platformách ako sú Unix, FreeBSD, Solaris, Microsoft Windows. Medzi najznámejšie a najpoužívanejšie webové servery patrí Apache.

Úlohou PHP skriptu je generovanie HTML stránky na serveri, ktorá sa následne pomocou protokolu HTTP posiela užívateľovi. Vznik jazyka PHP sa datuje do roku

¹⁸ KOCH, M. *Dátové a funkční modelování*. (2006), str. 56-60.

1995. Popularita PHP veľmi rýchlo rástla a v krátkej dobe sa zaradil medzi najpoužívanejšie programovacie jazyky na webe. Súčasťou PHP kódu môže byť aj HTML značky. Zdrojové kódy jazyka PHP sú voľne dostupné a šírené pod licenciou GPL. Základom syntax jazyka PHP sú programovacie jazyky C, Java a Perl. Pre skúsenejších programátorov je jazyk práve zo spomínaného dôvodu pomerne jednoduchý na naučenie.¹⁹

2.8.6 JavaScript

JavaScript patrí medzi technológie používané na strane klienta a jedná sa o skriptovací programovací jazyk, ktorý je podporovaný každým moderným webovým prehliadačom. Kód JavaScriptu nadobúda podobu jednoduchého textu, ktorý je vkladajú do HTML kódu. Účelom použitia JavaScriptu je zvýšenie funkcionality webovej aplikácie. Podmienkou používania JavaScriptu nie sú žiadne dodatočné inštalácie rozšírení a komponent. Javascript nepatrí medzi kompilované jazyky a nie je vhodný pre náročné výpočtové úlohy. Dôležitou vlastnosťou JavaScriptu je jeho samostatnosť, ktoré spôsobuje, že JavaScript sa na vývoj webu priamo neviaže. Bezpečnosť JavaScriptu je narušená potrebou plného dodanie kódu do klientskeho prehliadača.

2.8.7 AJAX

AJAX je skrátený názov pre asynchrónny JavaScript a XML (XML - Extensible Markup Language). Jedná sa o modernú technológiu, ktorá slúži pre vývoj interaktívnych webových aplikácií. Podstata AJAXu spočíva v možnosti volať server JavaScriptom na pozadí, a tým získavať potrebné dáta bez nutnosti opätovného načítania stránky. Rozoberaná technológia slúži na dosiahnutie dokonalejšej rovnováhy medzi akciami požadovanými klientom a aktivitami servera. AJAX vyrovnáva záťaž medzi klientom a serverom. Užívateľ tým pádom môže nerušene pracovať s webom, pričom server komunikuje na pozadí. Aby bolo možné technológiu AJAX využívať, nie je potrebné nič inštalovať, pretože technológia je implementovaná vo všetkých moderných webových prehliadačoch. Pomocou AJAXu je možné vytvárať užívateľsky prívetivé webové aplikácie.

¹⁹ ULLMAN, L. *PHP pokročilé programování pro World Wide Web*. (2002).

Na prezentovanie získaných dát sa používajú programovacie jazyky HTML, XHTML, CSS a JavaScript pre zobrazovanie dynamických zmien. Výmena informácií so serverom je zabezpečená prostredníctvom jazyka XML. Zo spomínaného vyplýva, že AJAX nie je programovací jazyk, ale je to využitie jednotlivých technológií spoločne.

2.9 Framework ako prostriedok tvorby webových IS

Framework je možné chápať ako softwarovú štruktúru, ktorá podporuje proces programovania, vývoja a organizácie rôznych softvérových projektov. Typickou vlastnosťou framework je, že obsahuje pomocné algoritmy, knižnice API (Application Programming Interface), návrhové vzory alebo postupy, ktorý sú odporúčané pri vývoji softwaru.

Primárnym účelom frameworku je uľahčenie práce návrhárom a vývojárom softwarových projektov, ktorý sa pomocou využitia môže sústrediť len na špecifiká svojho zadania a nemusia sa venovať štandardným úlohám a problémom, keďže ich riešenie už obsahuje framework. Použitím frameworku je výrazne redukovaná nutnosť písať rozsiahle zdrojové kódy a navyše je vo výsledku zdrojový kód prehľadnejší a štruktúrovanejší. Ďalšou výhodou frameworku je znižovanie časovej náročnosti programovania.

Existujú názory, že využitie frameworku prináša aj rôzne nevýhody ako nižšiu rýchlosť vykonávania algoritmu, či nutnosť naštudovať cudzí kód. Diplomová práca sa testovaniu rýchlosti jednotlivých frameworkov bude ešte venovať vo svojej návrhovej časti, avšak už teraz je možné tvrdiť, že opakované využívanie frameworku v množstve prípadov zvyšuje efektívnosť práce návrhárov a vývojárov a predstavuje významné uľahčenie ich pracovnej činnosti.

3 Analýza problému

Nasledujúca časť diplomovej práce bude rozoberať charakteristiky spoločnosti IJM Group, s.r.o. (ďalej v texte iba IJM Group). Súčasťou kapitoly je priblíženie slabých a silných stránok spoločnosti pomocou SWOT analýzy. Kapitola sa následne bude venovať výberu vhodného riešenia zaobstarania informačného systému pre spoločnosť IJM Group.

3.1 Predstavenie spoločnosti

Podnik IJM Group vznikol 24.02.2011. Jedná sa o veľmi mladú spoločnosť, ktorá si hľadá svoje miesto na trhu. Základné informácie o spoločnosti sú nasledovné:

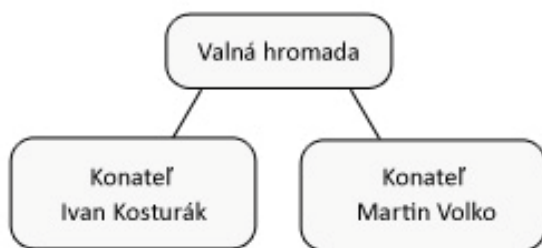
- **IČO:** 46050868,
- **DIČ:** 2023204865,
- **Právna forma:** spoločnosť s ručením obmedzeným,
- **Sídlo:** Sokolská 2518/48, Zvolen 960 01 Slovenská republika.

IJM Group momentálne výrazným spôsobom rozširuje svoje podnikateľské aktivity. S rozšírením činnosti je priamo spojený aj nárast náročnosti zabezpečenia dlhodobej udržateľnosti podnikových procesov. Jedinou výkonnou zložkou spoločnosti sú konatelia, ktorých činnosť je dopĺňovaná o výkony externých subjektov.

3.2 Právna forma a organizačná štruktúra

IJM Group s.r.o. bola založená tromi spoločníkmi ako spoločnosť s ručením obmedzeným. Každý z troch spoločníkov vlastní tretinový podiel. Základný kapitál spoločnosti dosahuje výšku 5 100 EUR. Štatutárnym orgánom spoločnosti sú dvaja konatelia z radov spoločníkov.

Organizačnú štruktúru je možné vidieť na obrázku 4.



Obrázok 4: Organizačná štruktúra²⁰

3.3 Predmet podnikania

Predmet činnosti spoločnosti IJM Group je:²¹

- kúpa tovaru na účely jeho predaja konečnému spotrebiteľovi (maloobchod) alebo iným prevádzkovateľom živnosti (veľkoobchod),
- sprostredkovateľská činnosť v oblasti obchodu,
- sprostredkovateľská činnosť v oblasti služieb,
- činnosť podnikateľských, organizačných a ekonomických poradcov,
- reklamné a marketingové služby,
- prieskum trhu a verejnej mienky,
- počítačové služby,
- služby súvisiace s počítačovým spracovaním údajov,
- vykonávanie mimoškolskej vzdelávacej činnosti.

Najintenzívnejšie zameranie je na oblasti elektronického obchodu a internetových služieb. Jadro predmetu podnikania IJM Group je možné rozdeliť na tvorbu:

- webových stránok,
- webových aplikácií,
- elektronických spoločenských hier,
- automatických obchodných systémov pre obchodovanie s menovými pármami.

²⁰ Zdroj: vlastný

²¹ *Obchodný register na internete* [online]. 2011 [cit. 2012-01-28]. Slovensky. Dostupný z URL <<http://orsr.sk/vypis.asp?ID=200687&SID=3&P=0>>.

Na prvý pohľad sa môže zameranie spoločnosti zdať rôznorodé. Dôležitý však je jeden spoločný faktor, ktorým je elektronická podoba poskytovaných služieb.

3.4 Súčasný IS spoločnosti IJM Group

V čase začiatku písania diplomovej práce spoločnosť IJM Group nedisponovala žiadnym IS. Vedenie spoločnosti si plne uvedomovalo dôležitosť kvalitného IS podporujúceho podnikové procesy. Spomínané uvedomenie bolo podporené aj narastajúcim množstvom práce, ktoré bez IS a zautomatizovania často sa opakujúcich procesov bránilo ďalšiemu rastu spoločnosti. Práve z dôvodu spomínaných skutočností predstavuje spoločnosť IJM Group ideálne prostredie pre zavedenie vlastného informačného systému, čím získa väčší prehľad a kontrolu nad svojimi procesmi. Ďalšie časti diplomovej práce budú začatú problematiku bližšie konkretizovať.

3.5 SWOT analýza spoločnosti IJM Group

V nasledujúcej kapitole je spracovaná SWOT analýza, ktorá sa venuje silným stránkam spoločnosti, slabým stránkam spoločnosti, príležitostiam a hrozbám, s ktorými sa spoločnosť IJM Group stretáva. Na konci kapitoly sa nachádza tabuľka č. 2, ktorá obsahuje zhrnutie výsledkov SWOT analýzy.

3.5.1 Silné stránky

Medzi silné stránky spoločnosti patrí najmä ambicióznosť, mladícka dravosť a snaha pracovať nielen pre finančnú odmenu, ale aj pre radosť.

Spoločnosť IJM Group môže aj napriek pomerne nízkemu vekovému priemeru stavať na dlhoročných skúsenostiach v oblasti vývoja webových aplikácií a správy serverov. Nezanedbateľnou konkurenčnou výhodou je snaha o neustálu inováciu.

Vedenie spoločnosti IJM Group si uvedomuje, že jedinou cestou ako skutočne uspieť, je spokojný zákazník a podľa tohto presvedčenia aj koná.

3.5.2 Slabé stránky

Nakoľko je spoločnosť IJM Group mladá a začínajúca, tak jej činnosť zatiaľ nebola systematicky spracovaná a neexistuje stále rozdelenie úloh. Ďalším faktorom, ktorý bráni rastu spoločnosti je absencia informačného systému, ktorý by umožňoval efektívnu správu produktov a služieb ponúkaných spoločnosťou ako aj správu zamestnancov a zákazníkov. Výkonná zložka spoločnosti často stráca čas procesmi a činnosťami, ktoré môžu byť automatizované. Terajší spôsob fungovania je možné označiť pojmom „organický“, čo určite nie je žiadaný stav a jeho zachovanie by s najväčšou pravdepodobnosťou nevedlo k úspešným výsledkom. Práve zo spomínaných dôvodov je momentálna činnosť spoločnosti poznačená značnou neefektívnosťou.

Za slabú stránku spoločnosti je možné považovať aj nízky vek podniku, ktorý nevytvára dostatočne seriózný dojem.

Typickým problémom spoločností zakladaných nielen mladými ľuďmi je nedostatok skúseností v oblasti podnikania, ktoré môžu vyústiť až do nezhôd medzi spoločníkmi. Kritickými oblasťami bývajú najmä smerovanie a zamerania spoločnosti, delba práce, práv a povinností a spôsoby odmeňovania.

3.5.3 Príležitosti

S neustálou snahou spoločnosti inovovať súvisí aj tendencia hľadať nové potenciálne trhy, prípadne neštandardné spôsoby oslovenia už existujúcich trhov. IJM Group operuje na elektronickom prevažne internetovom trhu, ktorý sa neustále mení. Pri vhodnom využití zmien a schopnosti na ne reagovať je možné premenlivosť trhu chápať ako príležitosť.

Jednu z najväčších príležitostí ponúka samotný predmet podnikania, ktorý vzhľadom na svoju nehmotnú podobu, umožňuje osloviť globálny trh, pričom bariéry vstupu takmer neexistujú. V prípade kvalitnej služby, efektívneho marketingu a jazykového prekladu spoločnosti nič nebráni predávať svoju službu na celom svete a tým osloviť nepomerne

väčšie množstvo zákazníkov, ako v prípade, keby bola podnikateľská činnosť zameraná na konkrétnu krajinu.

Spoločnosť IJM počas leta roku 2011 rozšíri svoju činnosť o podporu obchodovania na forexe. Trh s menovými pármami je najlikvidnejším trhom na svete s denným obratom prevyšujúcim 3 bilióny amerických dolárov, a preto je možné podporu forex obchodovania jednoznačne zaradiť medzi príležitosti spoločnosti.

3.5.4 Hrozby

V analýze príležitostí bola spomínaná rýchlosť zmeny elektronického trhu, ktorú je však možné zaradiť aj do hrozieb. Záleží len na tom, akým spôsobom spoločnosť pristupuje k zmenám a na schopnosti prispôbiť sa.

Operovanie na globálnom trhu prináša svoje riziká. Jedná sa najmä o možnosti vstupu konkurencie, ktorá je veľmi častá a pravdepodobná, keďže aj konkurencia má pomerne nízke bariéry vstupu. Práve z tohto dôvodu je esenciálne neustále zlepšovanie ponúkaných služieb a pridávanie hodnoty značke napríklad prostredníctvom ústretovej komunikácie a rôznych akcií podporujúcich vzťahy s verejnosťou, ktoré budujú dobré meno spoločnosti.

Aj keď IJM Group priamo neobchoduje s menovými pármami, nestabilná situácia na finančných trhoch predstavuje hrozbu pre podnikateľské aktivity spoločnosti. Spôsobené to je najmä tým, že automatické obchodné systémy, ktoré spoločnosť vyvíja a ponúka sú vhodnejšie pre použitie počas stabilnejších a pokojnejších období. Rizikovosť použitia obchodných systémov, ako aj náročnosť ich použitia, výrazne narastá práve počas turbulentných období, aké svet zažíva najmä od leta roku 2011.

Dôležitou hrozbou, ktorá má potenciál spôsobiť podniku IJM Group výrazné straty je softwarové pirátstvo. Výsledky všetkých podnikateľských aktivít spoločnosti majú digitálnu podobu, čím sa stávajú náchylné na ukradnutie, resp. nelegálne šírenie. Spoločnosť IJM Group sa snaží prijímať opatrenia znižujúce riziko porušenia autorských práv svojich vlastných produktov, avšak nelegálne šírenia digitálneho

obsahu je globálny problém, ktorého riešenie je veľmi náročné a doteraz sa ho nepodarilo uspokojivo vyriešiť.

Súhrn SWOT analýzy je možné vidieť v tabuľke 2.

SWOT analýza	
Silné stránky	Slabé stránky
• mladý ambiciózny tím, • skúsenosti v oblasti webových aplikácií a správy serverov, • kreatívne zmýšľanie, • max. zameranie na spokojnosť zákazníka.	• neexistujúce systémové riadenie procesov, • chýbajúci IS, • nízky vek spoločnosti, • vnútorné rozpory.
Príležitosti	Hrozby
• nové neobjavené trhy, • premenlivosť trhu, • podpora forex obchodovania.	• premenlivosť trhu, • konkurencia, • znížená predvídateľnosť a stabilita finančných trhov, • porušovanie autorských práv a softwarové pirátstvo.

Tabuľka 2: SWOT analýza²²

3.5.5 Závbery vyplývajúce zo SWOT analýzy

Zo SWOT analýzy vyplýva, že spoločnosť IJM Group je skutočne len na začiatku svojej cesty. Jedná sa o mladú spoločnosť, ktorá si musí najprv svoje miesto na trhu nájsť a až následne realizovať upevňovanie pozície.

Dôležitým faktorom úspechu bude otvorená komunikácia vo vnútri spoločnosti, ktorá zabráni zbytočným nezhodám a zabezpečí jasnú predstavu o cieľoch a smerovaní spoločnosti. Kvalitná komunikácia okrem iného predstavuje aj najsilnejší nástroj

²² Zdroj: vlastný

podporujúci kreatívne myslenie, bez ktorého by sa spoločnosť stala len ďalšou v sivom zástupe firiem.

Jednoznačne najväčším problémom IJM Group je nesystematická činnosť, ktorá v podstate neumožňuje ďalší rast spoločnosti. Odstránenie spomínaného problému je hlavnou náplňou diplomovej práce, keďže v prípade ignorovania potreby zavedenia IS hrozí, že spoločnosť svoj konkurenčný boj nezvládne a zanikne. Podstatné však je, že zavedenie IS má v podniku momentálne najvyššiu prioritu a výkonná zložka spoločnosti pracuje v plnom nasadení na vytvorení informačného systému, ktorý poslúži ako základný kameň stabilného rastu spoločnosti.

3.6 Základné požadované charakteristiky IS

V prípade úspešného zavedenia sa IS stane ťažiskovým prvkom kompletnej podnikateľskej činnosti spoločnosti IJM Group. Práve preto je potrebná čo najväčšia integrácia informačného systému s podnikovými procesmi s prihliadnutím na špecifické požiadavky, možnosti spoločnosti a cenu navrhovaného riešenia.

Požiadavky spoločnosti IJM Group na IS:

- **Prehľadnosť** - intuitívnosť používania, jednoznačná viditeľnosť vzťahu medzi akciou a jej výsledkom,
- **Stabilita** - schopnosť odolať užívateľským chybám a modifikáciám hardwarových a softwarových prostriedkov,
- **Bezpečnosť** - schopnosť odolať útokom na citlivé dáta a stabilitu systému a dôsledná politika práv (problematika je bližšie popísaná v teoretickej časti diplomovej práce),
- **Modulárnosť** - systém je tvorený viacerými samostatnými spolupracujúcimi celkami, ktoré zabezpečujú flexibilitu a možnosť pridávania ďalších funkcií,
- **Administrácia produktov,**
- **Administrácia zamestnancov,**
- **Administrácia zákazníkov,**

- *Administrácia fakturácií,*
- *Administrácia objednávok,*
- *Dostupnosť systému z ľubovoľného terminálu s internetovým pripojením,*
- *Možnosť efektívnej správy predaja do krajín sveta v rôznych jazykoch.*

Je potrebné zdôrazniť posledné kritérium. Spoločnosť IJM sa zameriava vyslovene na predaj produktov a služieb v elektronickej podobe. Tento spôsob podnikania za sebou nesie nezanedbateľnú výhodu a to minimálnu náročnosť distribúcie produktov do ľubovoľných končín sveta. Práve zo spomínaného dôvodu je možnosť spravovať podnikanie so zohľadnením správy jednotlivých krajín a jazykov kľúčová vlastnosť, ktorou nový informačný systém disponovať musí.

3.7 Požadované charakteristiky jednotlivých administrácií

Kapitola bližšie rozoberá nároky kladené na vlastnosti požadovaných administrácií systému.

3.7.1 Administrácia produktov

Administrácia produktov predstavuje súhrn funkcií starajúcich sa o správu produktov. Je vyžadované, aby administrácia obsahovala základné funkcie ako:

- pridávanie / úprava / vymazávanie produktu,
- zaraďovanie produktov do minimálne dvoch úrovní,
- vkladanie popisu k produktom,
- pridávanie obrázkov k produktom,
- pridávanie súborov k produktom,
- samostatné oceňovanie pre jednotlivé krajiny,
- hromadné oceňovanie produktu pre skupiny krajín,
- nastavovanie hodnoty provízie z predaja pre autorov produktov,
- vytváranie jazykových verzií produktu,
- prideľovanie jazykových verzií produktu do ponuky jednotlivých krajín.

3.7.2 Administrácia pracovníkov

Nejedná sa o plnohodnotnú administráciu miezd. Úlohou administrácie pracovníkov je pridávanie, editácia a vymazávanie pracovníkov, pričom pridaním nového pracovníka je takémuto pracovníkovi vygenerovaný prístup do systému a vytvorené e-mailové konto. Administrácia pracovníkov ďalej kalkuluje orientačné odmeny pracovníkov na základe dohodnutého kľúča. Predpoklad je, že odmeňovanie bude prebiehať províznym spôsobom na základe objemu predajov.

3.7.3 Administrácia zákazníkov

Administrácia zákazníkov má poskytovať prehľadný výpis zákazníkov spoločnosti s možnosťou ich filtrovania podľa logických atribútov. Vyhľadávanie zákazníkov je povinnou súčasťou systému a minimálne požadované vyhľadávacie kritérium je e-mail zákazníka.

Vyžadovaná je spolupráca s administráciou objednávok za účelom jednoduchšej a prehľadnej správy objednávok konkrétnych zákazníkov. Rovnako je potrebná kompatibilita s administráciou zamestnancov nakoľko existuje predpoklad, že každý zákazník bude pridelený konkrétnemu zamestnancovi podľa krajiny pôvodu zákazníka a jeho preferovaného jazyka komunikácie.

3.7.4 Administrácia fakturácií

Administrácia fakturácií má slúžiť ako podpora a zdroj podkladov pre účtovníctvo. Nejedná sa o plnohodnotnú náhradu ekonomického informačného systému, ktorým disponuje externý subjekt zaoberajúci sa účtovníctvom spoločnosti. Úlohou administrácie fakturácií je najmä:

- zbieranie a ukladanie údajov potrebných na fakturáciu,
- automatické odosielanie predfaktúr po vykonaní objednávok,
- poloautomatické odosielanie faktúr po kontrole platieb zodpovedným pracovníkom,

- sledovanie stavu faktúr (nezaplatená / zaplatená / po splatnosti / pred splatnosťou / a pod.)
- automatické odosielanie faktúr pri produktoch s periodickými platbami,
- možnosť vkladať základné údaje z prijatých faktúr a ich digitálne kópie.

3.7.5 Administrácia objednávok

Základom administrácie objednávok je prehľadný výpis objednávok s možnosťou logického filtrovania a sumarizáciou pre štatistické účely. Výpis je doplnený o vyhľadávanie objednávok minimálne podľa kritérií:

- číslo objednávky,
- e-mail zákazníka.

Modul objednávky rozlišuje rôzne stavy objednávok ako napríklad:

- vybavená objednávka,
- objednávka v procese vybavovania,
- zrušená objednávka.

3.8 Výber spôsobu zavedenia IS pre spoločnosť IJM Group

Pri rozhodnutiach o zavedení IS vždy dochádza k výberu spôsobu, akým bude IS zavedený. Základnými otázkami, ktoré je potrebné rozhodnúť sú:

- Zavedie podnik IS vo vlastnej réžii alebo využije externého dodávateľa?
- Využije podnik „krabicový“ IS alebo IS na mieru?

Bližšie charakteristiky jednotlivých možností sú bližšie spracované v teoretickej časti diplomovej práce.

Spoločnosť IJM Group rozhodovala o potrebe zaviesť IS necelého pol roka po svojom vzniku v čase, keď disponovala voľnými finančnými zdrojmi na úrovni rádovo niekoľko tisíc Eur. Vzhľadom na vysokú špecifickosť niektorých požiadaviek na IS je zrejmé, že nebolo možné nájsť „krabicové“ riešenie IS, ktoré by tieto požiadavky spĺňalo, čo znamenalo, že štandardne najpopulárnejšie riešenie neprichádzalo do úvahy. Ďalšou možnosťou bolo vytvorenie IS na mieru externým subjektom. Tento variant by v prípade jeho zvolenia znamenal potrebu dôslednej analýzy potrieb spoločnosti IJM Group externým subjektom, ako aj pravidelné konzultácie a otvorenú komunikáciu so subjektmi mimo spoločnosti. Vo výsledku by sa zvýšila časová náročnosť a riziko využitia dôverných interných informácií spoločnosti. Dôležitým faktorom samozrejme ostáva vysoká cena riešenia na mieru, na ktorú by rozpočet začínajúceho podniku bez využitia úveru nestačil.

Výkonná zložka spoločnosti IJM Group má skúsenosti a znalosti potrebné na vytvorenie IS podľa vlastných predstáv a po zohľadnení vyššie spomínaných faktorov bolo rozhodnuté, že IS bude vytvorený vlastnými silami. Odhadovaná časová náročnosť je vyššia ako pri využití riešenia na mieru, avšak za cenu výrazných úspor a nárastu pravdepodobnosti zhody finálneho systému a predstavy o ňom. Spoločnosť IJM Group sa rozhodla investovať čas a energiu svojej výkonnej zložky, ktorým disponuje, namiesto finančných prostriedkov, ktoré v dostatočnej miere k dispozícii neboli.

Nasledujúca kapitola priblíži zvolený spôsob vytvárania vlastného IS, ktorý využíva softwarovú štruktúru framework.

3.9 Výber frameworku

Po výbere spôsobu zavedenia je nutné zvolený spôsob špecifikovať. Teoretická časť diplomovej práce naznačila výhody využitia frameworku ako softwarových štruktúr uľahčujúcich proces návrhu a vývoja aplikácií. Práve pre užitočnosť frameworkov bolo rozhodnuté o použití jedného z nich.

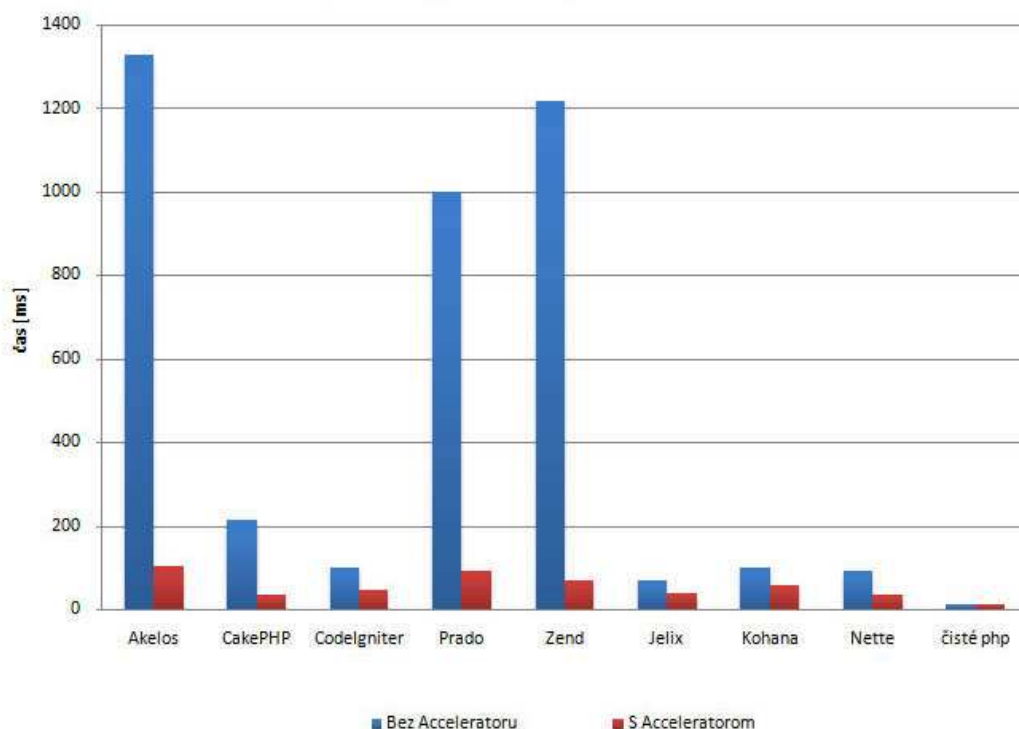
Nasledujúcim dôležitým krokom je v pomerne širokej ponuke vybrať framework, ktorý bude najlepšie pokrývať potreby realizovaného softwarového projektu. Najpoužívanejšie PHP frameworky sú:

- Akelos
- CakePHP,
- CodeIgniter,
- Prado,
- Zend Framework,
- Jelix,
- Kohana,
- Nette.

3.9.1 Porovnanie priemernej rýchlostí frameworkov²³

Graf 1 zobrazuje priemerné hodnoty získané zo zrovnávacieho testu výkonosti frameworkov. Test bol založený na štyroch meraniach (30x zobrazení užívateľov, 30x zobrazení členov, 30x update dotaz na databázu MySQL, 30x editačné formuláre). Pre ilustráciu bolo v teste zahrnuté aj čisté PHP. Z výsledkom je zrejmé, že frameworky ako Akelos, Prado a Zend majú problémy s rýchlosťou. Primerný čas troch spomínaných frameworkov dosahoval alebo dokonca prekračoval 1 sekundu. Po použití nástroja eAcceleratoru sa výsledky najpomalších frameworkov značne zlepšili.

²³ *Velký test PHP frameworků – Root.cz* [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL <<http://www.root.cz/clanky/velky-test-php-frameworku-2008>>.



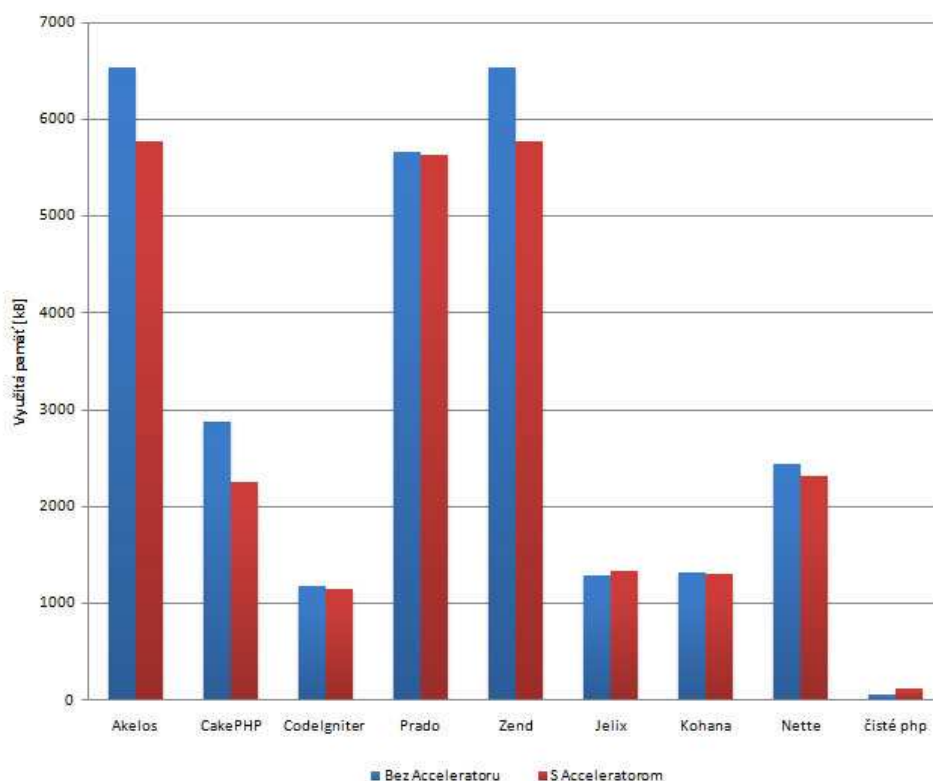
Graf 1: Porovnanie priemernej rýchlosti frameworkov²⁴

eAccelerator je nástroj pre zvýšenie spracovávanie PHP skriptu. Základom eAcceleratoru je princíp vytvárania medzipamäte (cache) PHP skriptov. Po zapnutí eAcceleratora, je možné zaznamenať výrazné zlepšenie priemerných časov všetkých frameworkov. Ďalšie frameworky, ktoré boli súčasťou testu, dosiahli veľmi pozitívne výsledky. Konkrétne český Framework Nette dosahuje rovnako výborné výsledky.

3.9.2 Pamäťové nároky frameworkov a ich porovnanie

Pamäťová náročnosť je zásadnou vlastnosťou frameworkov. Graf 2 ukazuje, že pamäťová náročnosť frameworkov Akelos, Prado, Zend dosahuje hodnoty okolo 6 MB. Výsledky ostatných frameworkov sú lepšie a dosahujú nižšie hodnoty. Použitie eAcceleratoru nemá na celkové pamäťové nároky žiadny významný vplyv.

²⁴ Upravené podľa: *Velký test PHP frameworků – Root.cz* [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL <<http://www.root.cz/clanky/velky-test-php-frameworku-2008>>.



Graf 2: Porovnanie pamäťových náročností frameworkov²⁵

Na vývoj navrhovaného IS, ktorý zohráva dôležitú úlohu v pokračovaní návrhovej časti diplomovej práce, bol zvolený framework Nette. Nette dosiahol v testoch výborné výsledky, keď patril medzi najrýchlejšie frameworky s pomerne nízkymi nárokmi na pamäť. Medzi jeho ďalšie výhody patrí podpora v českom jazyku a veľká komunita, ktorá vytvára množstvo doplnkov a modulov.

3.9.3 Framework Nette

Nasledujúca kapitola v krátkosti predstavuje zvolený framework. Nette je výkonný framework pre rýchle vytváranie kvalitných webových aplikácií v PHP 5 pohodlnou cestou. Silnou stránkou frameworku Nette je odstránenie bezpečnostných rizík. Nette Framework je rozširovaný pod licenciou GPL ako slobodný software.

²⁵ Upravené podľa: *Velký test PHP frameworků – Root.cz* [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL <<http://www.root.cz/clanky/velky-test-php-frameworku-2008>>.

Hlavné výhody Nette sú²⁶:

- kvalitný šablónovací systém,
- objektový návrh,
- moderný framework s podporou HTML5, AJAX alebo SEO,
- bezpečnosť pred útokmi,
- kvalitná dokumentácia
- aktívna komunita v Českej republike,
- nástroje ladenia chodu programu,
- databázová vrstva dibi,
- je zadarmo.

Diplomová práca v krátkosti rozoberá jednu z najsilnejších stránok frameworku Nette, ktorou je zabezpečenie webových aplikácií. Nette využíva viaceré bezpečnostné technológie:²⁷

- **URL attack, control codes, invalid UTF-8** je metóda, ktorá sa snaží podstrčiť webovej aplikácii škodlivý vstup. Zväčša sa jedná o poškodenie XML výstupov, prípadne získanie informácií z databázy. Ošetrením každého vstupu na úrovni bytov je možné útoku zabrániť.
- **Cross-site scripting XSS** je metóda útoku na webové stránky, ktorá využíva neošetrené výstupy. Útočník dokáže pomocou Cross-site scripting XSS podstrčiť do stránky svoj vlastný kód a zmení stránku, prípadne získa citlivé údaje. Pre ochranu je potrebné korektné ošetrenie reťazcov.

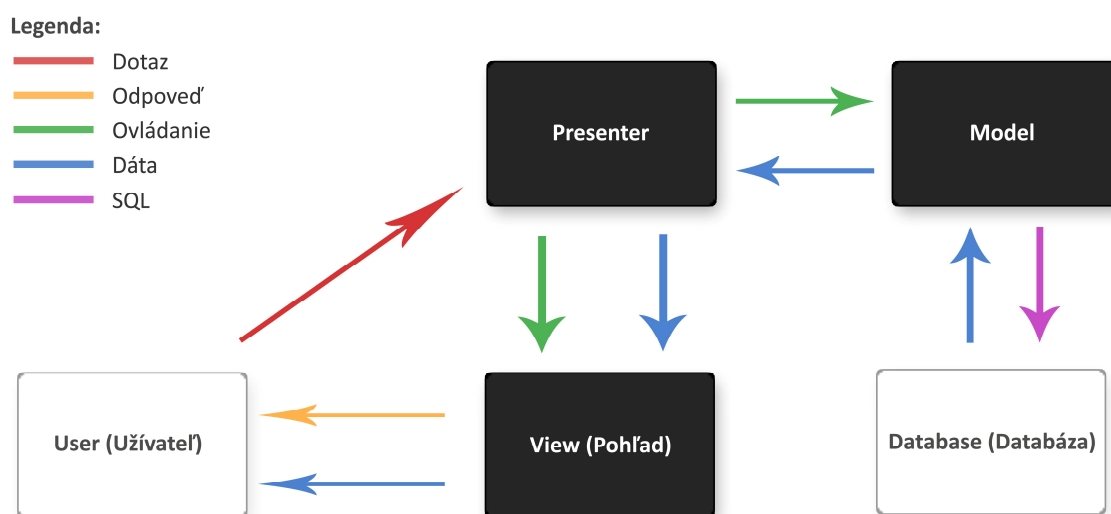
²⁶ *Nette Framework* [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL<<http://nette.org/cs/>>.

²⁷ *Dokonalé zabezpečenie webových aplikácií | Nette Framework* [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL <<http://nette.org/cs/hlavni-prednosti/bezpecnost>>.

- **Session hijacking, session stealing, session fixation** je metóda, pri ktorej útočník snaží podstrčiť svoje session ID. Týmto spôsobom sa mu môže podariť získať vstup do webovej aplikácie bez znalosti hesla. Útok bráni správna konfigurácia PHP, ktorú Nette vykonáva automaticky.
- **Cross-site request forgery** je metóda, ktorej základom je donútenie užívateľa navštíviť stránku, ktorá zaútočí na webovú aplikáciu. Ochrana proti útoku spočíva v generovaní a overovaní autorizačného tokenu.

Ďalšou dôležitou vlastnosťou frameworku Nette je podpora SEO (Search Engine Optimization) prostredníctvom bránenia existencie duplicitných URL vedúcich na totožný obsah. Vo výsledku vyhľadávače každú stránku indexujú iba jedenkrát.²⁸

Framework Nette však je postavený na vzore MVP (Model, View, Presenter), ktorý ukazuje obrázok 5. Aplikácia je rozdelená na tri časti a každú časť je možné vyvíjať samostatne. Účelom je zvýšenie efektivity a flexibility vývoja a udržiavania aplikácie.



Obrázok 5: Schéma MVP²⁹

²⁸ Vytvoření Presenteru | Nette Framework [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL <<http://doc09.nette.org/cs/quickstart/vytvoreni-presenteru>>

²⁹ Upravené podľa: Vytvoření Presenteru | Nette Framework [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL <<http://doc09.nette.org/cs/quickstart/vytvoreni-presenteru>>

Cyklus fungovania vzoru MVP začína užívateľom, ktorý vydá dotaz. Dotaz je prijatý frameworkom Nette, ktorý posunie a spustí príslušný presenter (riadiaci člen). Presenter následne komunikuje s modelom, od ktorého si žiada potrebné dáta. Model dáta získava z databázy, súboru a pod. Presenter dáta prevezme a posúva ich pohľad. Úlohou pohľadu je zobrazit' dáta užívateľovi.

Nezanedbateľnou výhodou Nette je databázová vrstva Dibi. Účelom Dibi je zjednodušiť písanie SQL príkazov a zvýšiť efektivitu programátorskej rutiny.

Hlavné prednosti databázovej vrstvy dibi: ³⁰

- zjednodušený zápis SQL,
- uľahčenie práce programátorom,
- zníženie výskytu chýb (prehľadnosť),
- automatické formátovanie špeciálnych typov (dátum, reťazec a pod.),
- jednotné základné funkcie pre vykonanie príkazu, získanie výsledku a pod.

³⁰ *Quick Start | dibi* [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL<
<http://dibiphp.com/cs/quick-start>>.

4 Vlastné návrh riešenia

Návrhová čas diplomovej práce sa bude postupne venovať rámcovému popisu vyvíjaného webového IS a detailnej charakteristiky podpory jedného z hlavných podnikových procesov spoločnosti IJM Group. Detailná charakteristika podpory hlavného podnikového procesu obsahuje štruktúrované ukážky zdrojového kódu, predstavenie činnosti ukážok zdrojového kódu a ilustračné obrázky užívateľského rozhrania.

4.1 Rámcová charakteristika návrhu a vývoja webového IS

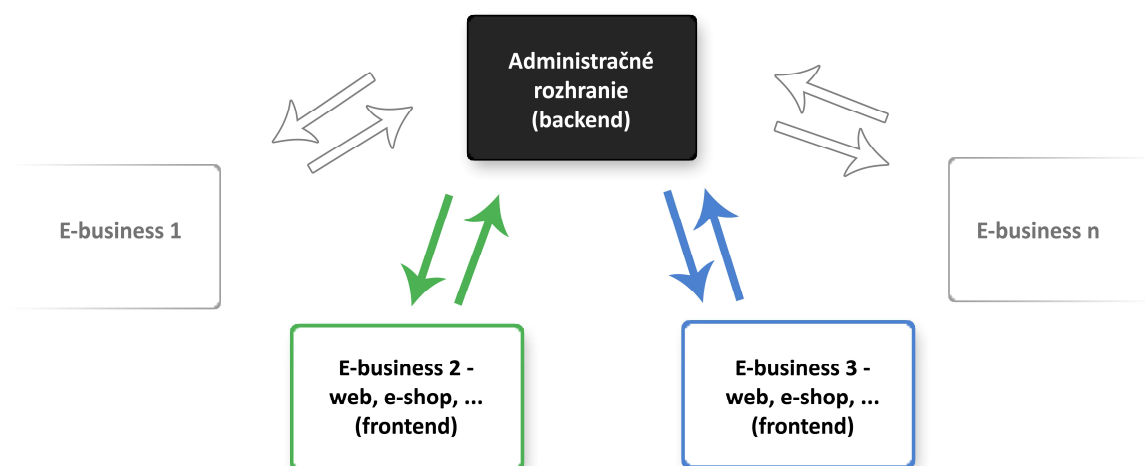
Návrh a realizácia priamo nadväzuje na funkčné požiadavky na IS, ktoré sú rozpracované v analytickej časti diplomovej práce. Úlohou vyvíjaného IS je umožniť hodnotné fungovanie spoločnosti. Bez zavedenie rozoberaného IS nebude spoločnosť IJM Group schopná vykonávať svoju podnikateľskú činnosť v sfére, ktorú si zvolila ako súčasť svojich strategických plánov.

Hlavné vlastnosti IS, ktoré je možné požadovať aj za časť dôvodov, prečo bolo uprednostnené vlastné riešenie pred kúpou už vytvoreného IS sú:

- **Plnohodnotná multijazyčná podpora** – nejedná sa len o možnosť prepínania jazykových mutácií webových stránok. Plnohodnotná podpora jazyka spočíva v možnosti úplnej variability pri vytváraní ponuky produktov pre jednotlivé krajiny. Podporovanou možnosťou je aj uplatnenie cenovej diskriminácie 3. stupňa, čo znamená, že systém umožňuje efektívne určovať cenu produktov pre rôzne krajiny, či skupiny krajín.
- **Optimalizácia systému pre el. produkty** – vlastnosť spočíva v prispôbení sa systému na prezentovanie a distribúciu produktov výhradne v el. podobe. Súčasťou el. produktov sú rôzne druhy súborov, ktoré systém umožňuje editovať a prehľadne zobrazovať užívateľovi. Spôsob distribúcie produktov zohľadňuje nároky na výpočtový výkon a spoľahlivosť doručenia produktu koncovému zákazníkovi. Zvolený spôsob distribúcie je postavený na sprístupňovaní určitého

obsahu súborového servera určitým zákazníkom. Proces pridelovania práv k jednotlivým súborom prebieha na základe splnenia vopred stanovených podmienok, ako sú napríklad zaplatenie objednávky, či uplatnenie zľavového kupónu a pod. Vyžadovaná je maximálna možná automatizácia procesov.

- **Podpora globálneho managementu krajín** – jednou z najväčších výhod elektronickej podoby produktov je možnosť ich globálnej distribúcie. Vyvíjaný IS musí spomínanú výhodu využívať, čo znamená, že vzhľadom na náročnosť celosvetovej správy ponúk produktov, či už po stránke časovej, lingvistickej alebo znalostnej, je potrebné, aby správu ponúk produktov mohol efektívne vykonávať celý tím ľudí. Spomínaný tím je tvorený tzv. správcami krajín, pričom každý správca má na starosti ponuku a starostlivosť o zákazníkov pre konkrétnu krajinu, či skupinu krajín. Odmeňovací proces správcov krajín prebieha na základe provízneho systému.
- **Integrované administratívne rozhranie** - koncept informačného systému bol navrhnutý tak, aby sa k jednému administratívne rozhranie (backendu) pripájali viaceré webové stránky, e-shopy a pod. Jedným slovom je možné práve vymenované položky označiť slovom „frontendy“. Spoločnosť IJM Group sa zameriava na viaceré formy e-business, ktoré však majú spoločné črty a je ich možné spoločne spravovať. Práve zo spomenutých dôvodov je integrácia viacerých frontendov pre jednotlivé podnikateľské zámery veľmi vítanou vlastnosťou realizovaného IS. Popisovaný koncept bližšie predstavuje obrázok 6.
- **Štandardné vlastnosti a funkcie** – medzi štandardy známe aj z bežne predávaných systémov, ktoré sú podporované aj systémom vyvíjaným, patrí napr. registrácia užívateľov, prihlasovanie užívateľov s možnosťou správy vlastného účtu, odosielanie rôznych verifikačných a informačných e-mailov, podpora platieb PayPal a pod.



Obrázok 6: Integrované administračné rozhranie³¹

4.1.1 Databáza a jej štruktúra

Databáza ako aj jej štruktúra je navrhnutá pre transakčné účely, čo znamená, že tabuľky databázy sú v normalizovanej podobe. Bližšie informácie o normalizácii je možné nájsť v teoretickej časti diplomovej práce. Súčasťou návrhu databázy je optimalizácia dátových typov za účelom čo najrýchlejšej odozvy. Detailný popis databázy vyžaduje množstvo priestoru. Diplomová práca z dôvodu svojich obsahových obmedzení neobsahuje detailný popis databázy, ktorý naviac pre plné pochopenie problematiky diplomovej práce nie je potrebný. Ukážku entitno-relačného modelu databázy pre spracovanie konkrétneho procesu spracovania objednávky ukazuje obrázok 7 v kapitole 4.2.1.

4.1.2 Adresárová a súborová štruktúra

Štruktúrované rozčlenenie súborov informačného systému má zásadný vplyv na prehľadnosť, udržateľnosť a prípadné rozširovanie systému. Štandardná štruktúra frameworku Nette je z malej časti upravená a vo výsledku vyzerá nasledovne:

- /koreňový adresár

³¹ Zdroj: vlastný

- o ../log
- o ../sub
 - ../../admin
- o ../web
 - ../../app
 - ../../log
 - ../../libs
 - ../../temp
 - ../../www

Adresár log

Adresár obsahuje záznamy o činnosti frameworku Nette.

Adresár sub

Adresár slúži ako priestor pre subdomény. V prípade realizovaného informačného systému sa jedná o subdoménu admin, ktorá obsahuje súbory administratívneho rozhrania systému.

Adresár web

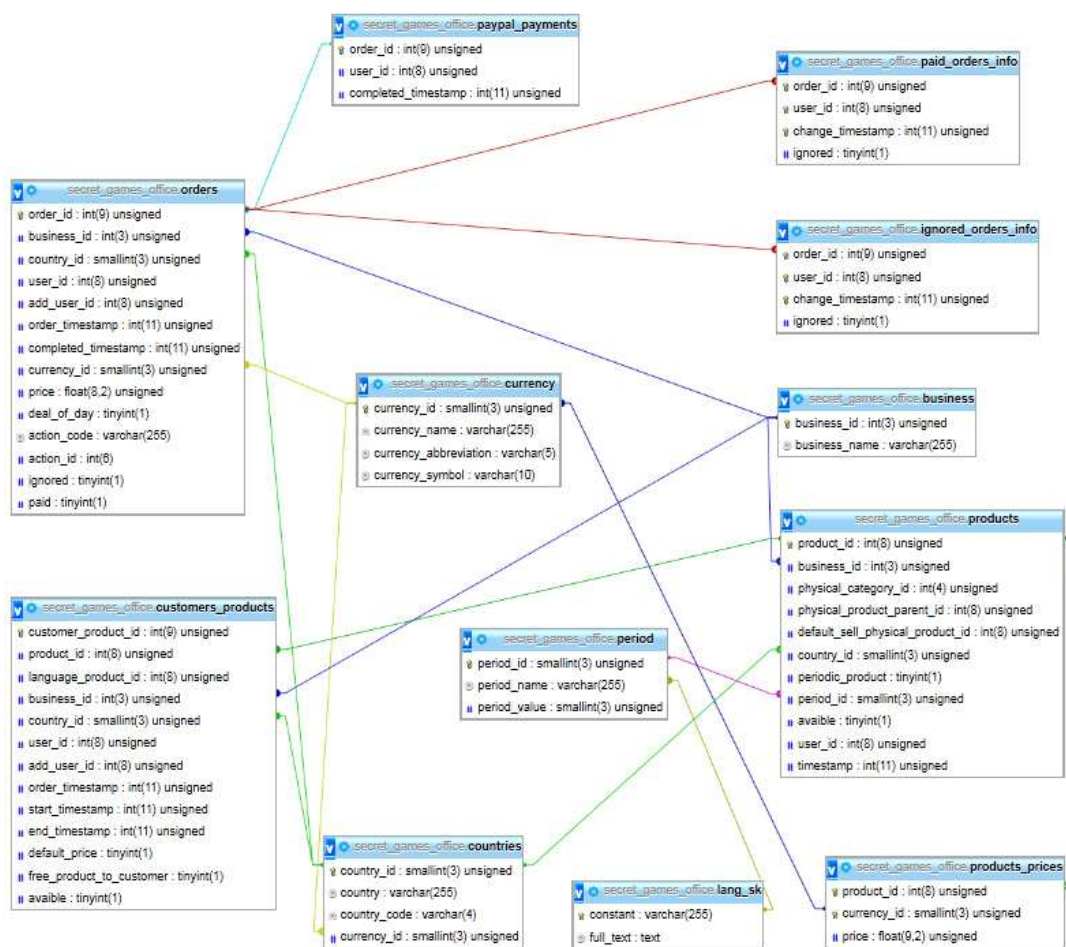
Adresár obsahuje hlavné komponenty webového informačného systému. V podadresári „app“ je možné nájsť riadiace skripty v podobe presenterov, šablóny zobrazujúce informácie o užívateľovi a moduly na obsluhu databázy. V adresári „log“ sa nachádzajú záznamy o činnosti frameworku Nette. Adresár „libs“ obsahuje knižnice, t.j. súbory, ktoré nie sú viazané na ostatné časti systému a môžu byť ľubovoľne využívané v rôznych častiach systému bez zbytočných duplícít. V adresári „temp“ sú uložené dočasné súbory, ktoré Nette framework využíva pri svojej činnosti. V adresári „www“ je možné nájsť súbory obsluhujúce grafickú úpravu systému.

4.2 Návrh systémovej podpory procesu spracovania objednávok

Jeden z najdôležitejších podnikových procesov v spoločnosti IJM Group je spracovanie objednávky. Práve zo spomenutého dôvodu bol proces spracovania objednávky v rámci diplomovej práce zvolený pre detailný rozbor. Diplomová práca sa vo svojej ďalšej časti bude venovať popisu návrhu daného procesu od jeho abstraktného začiatku v podobe „vôle objednať“ až po vybavenie objednávky. Ako prostriedok priblíženia riešeného procesu boli zvolené rôzne diagramy, ktorým sa venujú jednotlivé podkapitoly.

4.2.1 Entitno-relačný diagram

Entitno-relačný diagram, ktorý je zobrazený na obrázku 7 nezachytáva iba dáta ale aj väzby medzi nimi. V prípade použitého entitno-relačného diagramu sa nejedná len o návrh databázy, ale diagram predstavuje ukážku skutočnej databázovej štruktúry realizovaného IS. Dôležité je poznamenať, že zobrazený diagram nepredstavuje kompletnú databázovú štruktúru, ktorá sa využíva pri vykonávaní procesu spracovania objednávok, keďže zložitosť kompletnej reálne používanej databázovej štruktúry výrazným spôsobom znižuje jej prehľadnosť. Práve zo spomínaného dôvodu bol vytvorený ilustračný diagram, ktorý síce neobsahuje všetky využívané tabuľky z rôznych databáz, avšak zameriava sa na jadro procesu spracovania objednávok a pre priblíženie problematiky využívaných dát a ich väzieb počas rozoberaného procesu plne postačuje.

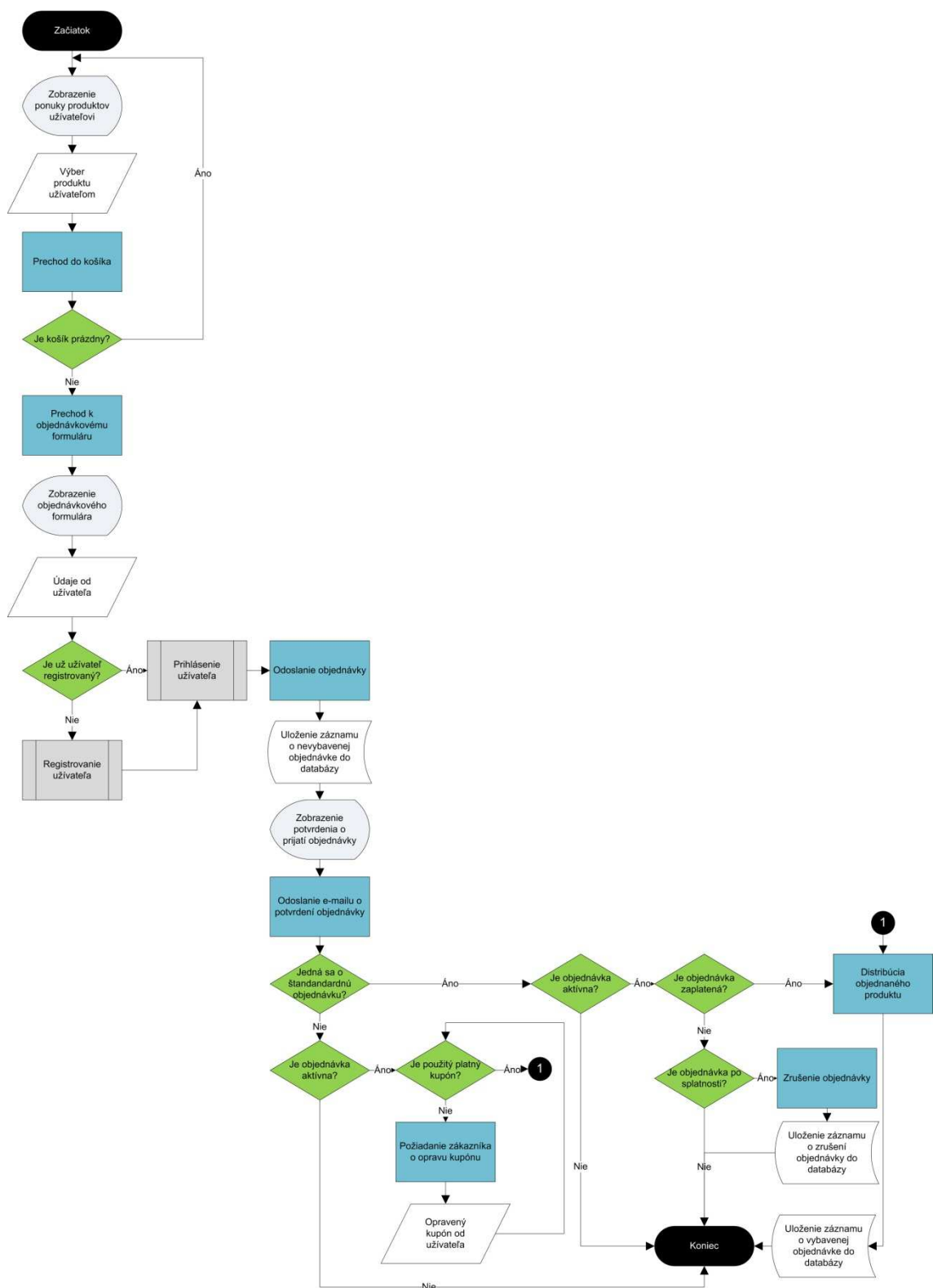


Obrázok 7: Entitno-relačný diagram³²

4.2.2 Vývojový diagram

Vývojový diagram patrí k najpoužívanejším a zvolený bol z dôvodu jeho možnosti veľmi dobre a prehľadne zachytiť vetvenie spracovania úlohy podľa splnenia, či nesplnenia požadovaných podmienok. Vývojový diagram podnikového procesu spracovania objednávok zobrazuje obrázok 8.

³² Zdroj: vlastný

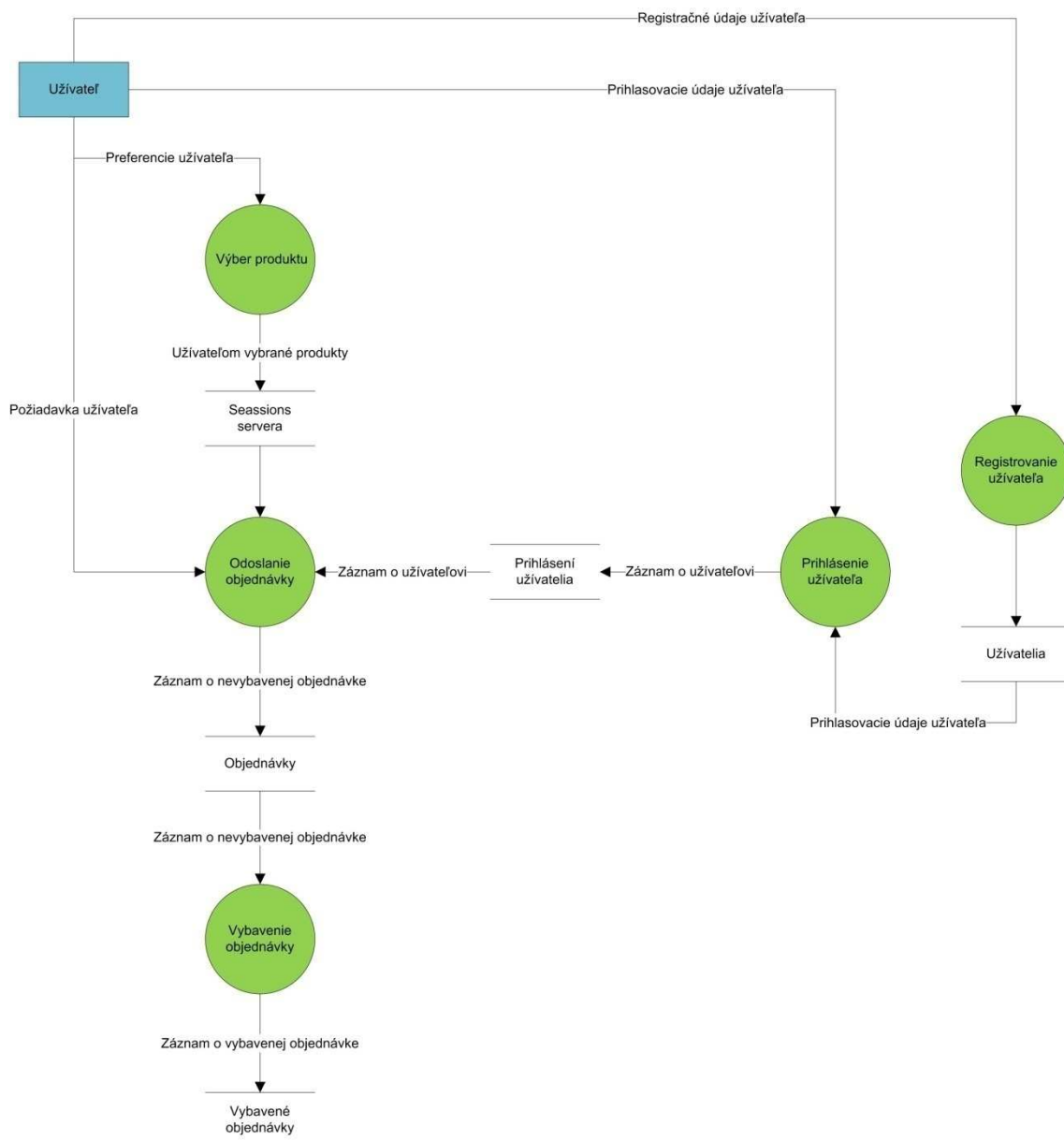


Obrázok 8: Vývojový diagram³³

³³ Zdroj: vlastný

4.2.3 Diagram datových toků (DFD - Data Flow Diagram)

DFD diagram poskytuje pohľad na problematiku procesu spracovania objednávok, ktorý je zameraný na jednotlivé dátové vstupy a výstupy jednotlivých prevádzaných činností rozoberanej úlohy. DFD diagram ilustruje obrázok 9.

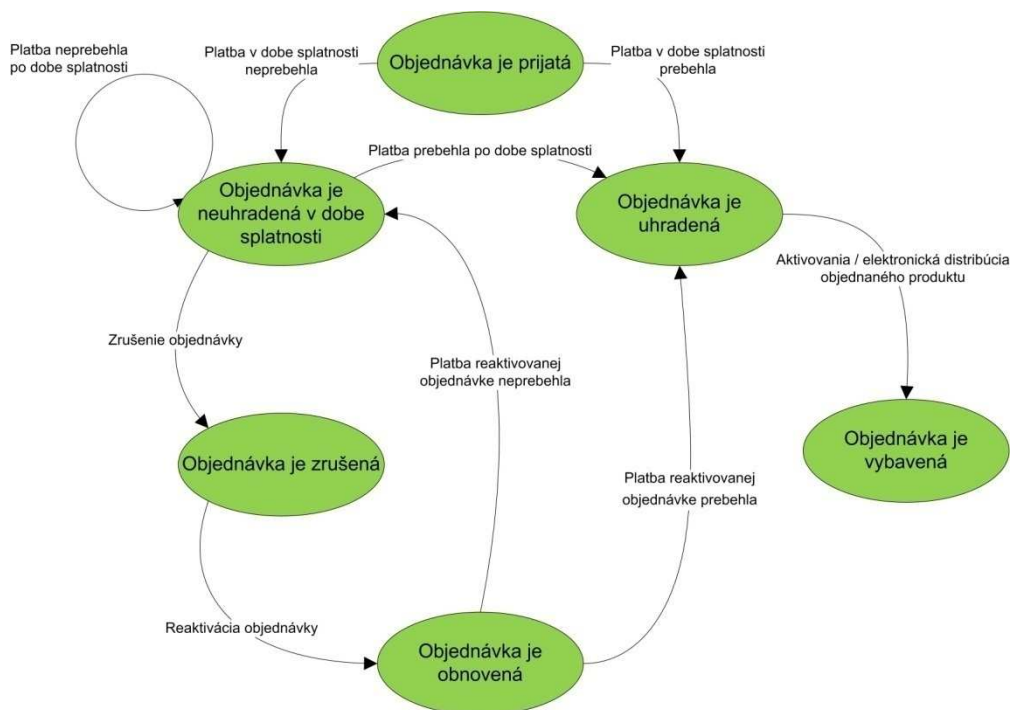


Obrázok 9: Diagram dátových tokov³⁴

³⁴ Zdroj: vlastný

4.2.4 Stavový diagram

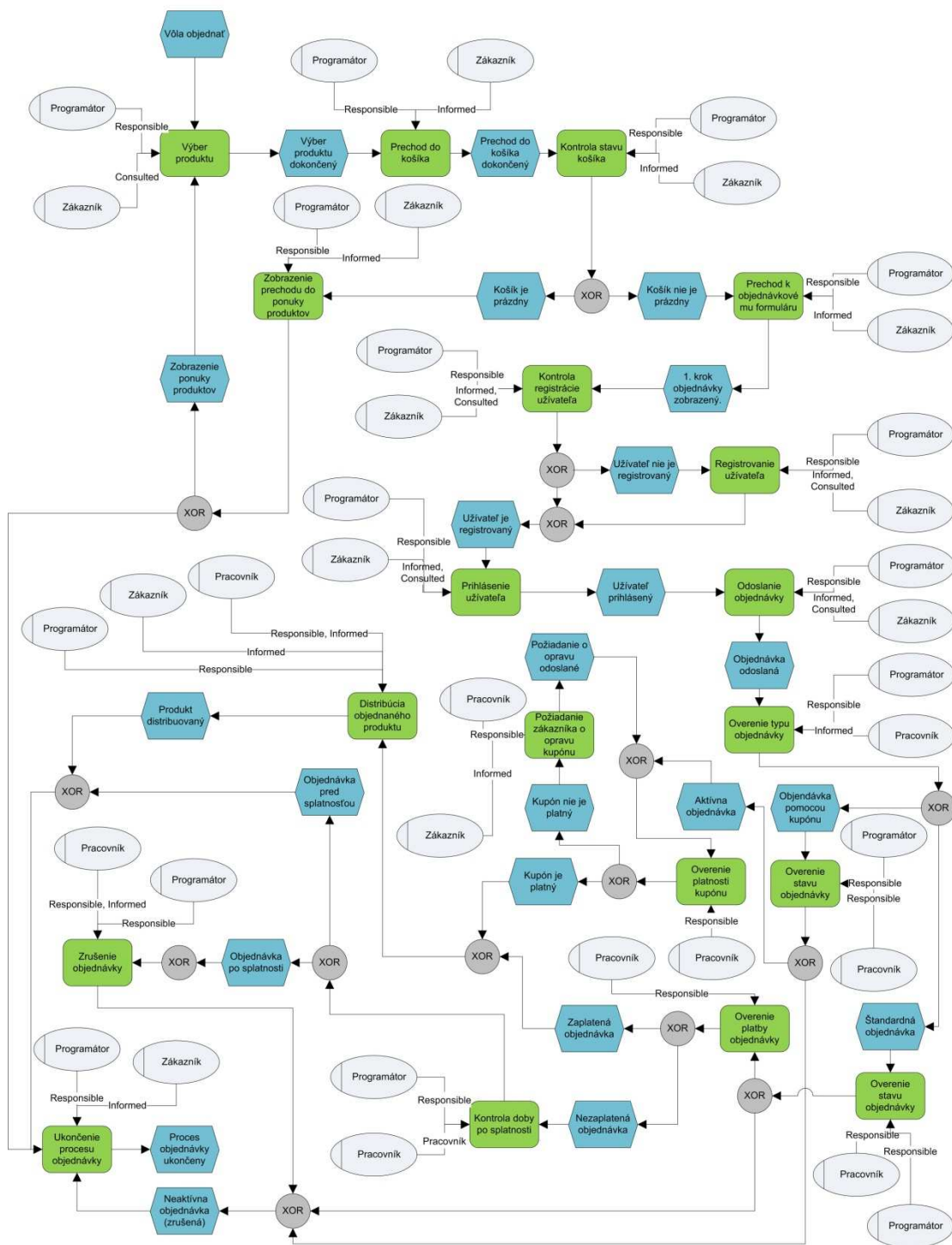
Na základný popis stavov, ktoré môže nadobúdať objednávka bol pre svoju jednoduchosť a zrozumiteľnosť vybraný stavový diagram zobrazený na obrázku 10.



Obrázok 10: Stavový diagram (Zdroj: vlastný)

4.2.5 EPC (Event-driven Process Chain) diagram

Zo zvolených diagramov predstavuje EPC diagram najkomplexnejší pohľad na problematiku procesu spracovania objednávky. EPC diagram obsahuje okrem vetvenia úlohy, procesných aktivít (funkcií) a udalostí aj procesné role priradené jednotlivým procesným aktivitám. Spomínaný EPC diagram je možné vidieť na obrázku 11.



Obrázok 11: EPC Diagram³⁵

³⁵ Zdroj: vlastný

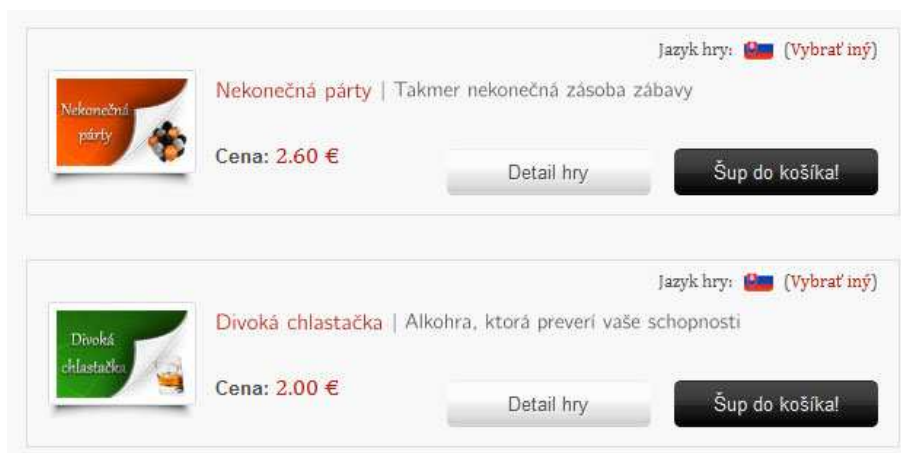
4.3 Realizácia systémovej podpory procesu spracovanie objednávok

Nasledujúca kapitola sa venuje samotnej realizácii systémovej podpory procesu spracovania objednávok so zameraním na štandardnú objednávku. Objednávka pomocou zľavového kupónu, ktorá je spomínaná v návrhovej časti systémovej podpory riešeného procesu už nie je z rozsahových dôvodov detailne rozoberaná.

Jednotlivé kroky, ktoré systém vykonáva sú spracované v samostatných podkapitolách a podložené relevantnými časťami zdrojového kódu s komentármi funkcionality. Popisné komentáre funkcionality sú od zvyšku zdrojového kódu odlíšené sivou farbou. Pre ilustráciu sú v kapitolách použité aj obrázky užívateľskej a administrátorskej časti webového IS.

4.3.1 Pridávanie produktov do košíka

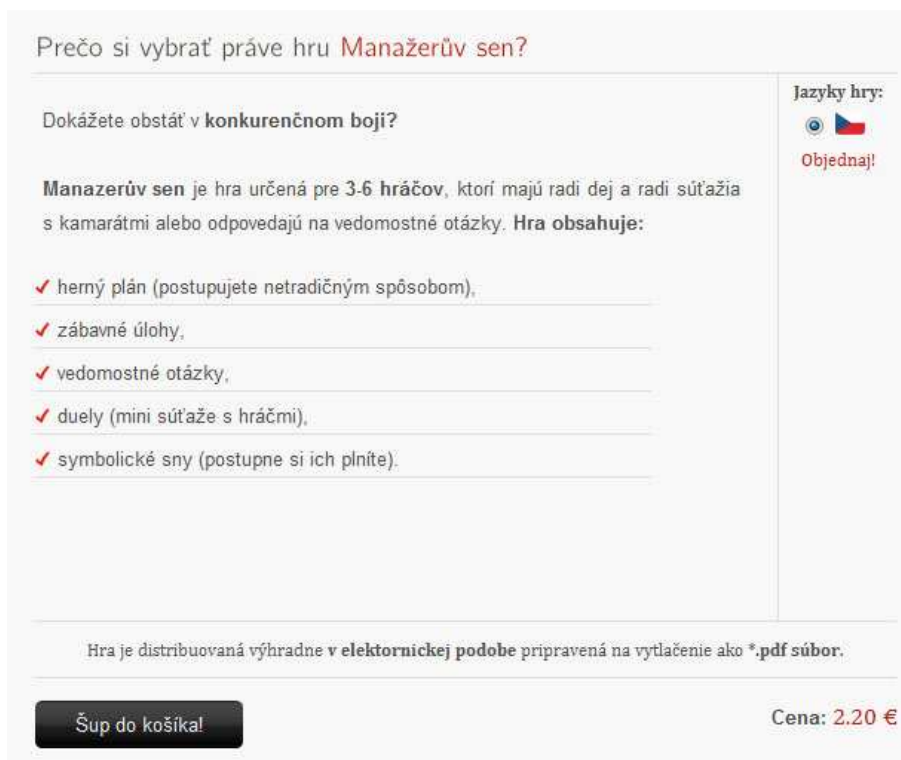
Užívatelia majú možnosť pridávania produktov v liste ponúkaných produktov, ktorý je ilustrovaný obrázkom 12.



Obrázok 12: Ilustrácia pridávania do košíka (zoznam ponúkaných produktov)³⁶

Ďalšou možnosťou je pridanie produktu do košíka priamo z detailného popisu produktu, ako ukazuje obrázok 13.

³⁶ Zdroj: vlastný



Obrázok 13: Ilustrácia pridávania do košíka (detail ponúkaného produktu)³⁷

Produkty sú v systéme vedené pod unikátnym ID, ktoré špecifikuje zaradenie hier do konkrétneho e-business spoločnosti, kategóriu produktu, rodičovský produkt (viažu sa naň jazykové mutácie), jazykovú verziu produktu, zaradenie do krajiny a pod. Ku všetkým produktom sú pridelené ceny na základe krajiny, z ktorej si užívateľ ponuku prezerá. Jedná sa o cenovú diskrimináciu 3. stupňa, ktorej podpora bola jednou z požiadaviek na informačný systém.

Užívateľ objednáva hru prostredníctvom odkazu, ktorý je na obrázku 7 a 8 označený ako „Šup do košíka!“. Znamená to, že potrebné parametre sú systému podané prostredníctvom URL adresy. Objednávka je následne uložená do sessions servera a užívateľovi sa zobrazí informačná hláška. Obsah hlášky závisí na výsledku procesu ukladania produktu do košíka a môže obsahovať buď informáciu o chybe alebo úspechu procesu.

V tabuľke č. 3 je časť zdrojového kódu presentera, ktorý má na starosti práve spomenutú časť spracovávania objednávok.

³⁷ Zdroj: vlastný


```

// order - parameter v URL, ktorý špecifikuje, že sa jedná o akciu pridávania hry do
košíka.

if($id_2=="order")
{
    // Kontrola, či do url adresy prišlo číslo. V prípade že prišlo, je možné vykonať
    databázový dotaz.

    if(is_numeric($id_3))
    {
        // Databázový dotaz, ktorý zisťuje detailné informácie o danom produkte (zaradenie,
        meno, jazyk, cenu a pod.)

        $hra=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT ".$mena_db-
        >db_invoice_name.".products.product_id,
        ".$mena_db->db_invoice_name.".products.physical_product_parent_id,
        (SELECT ".$mena_db->db_site_name.".lang_admin"."$this->language"."full_text from
        ".$mena_db->db_site_name.".physical_products, ".$mena_db-
        >db_site_name.".lang_admin"."$this->language." where ".$mena_db-
        >db_invoice_name.".products.default_sell_physical_product_id and ".$mena_db-
        >db_invoice_name.".products.physical_products.product_id = ".$mena_db-
        >db_site_name.".physical_products.product_id and ".$mena_db-
        >db_site_name.".lang_admin"."$this->language"."constant) as product_name
        FROM ".$mena_db->db_invoice_name.".products, ".$mena_db-
        >db_invoice_name.".products_prices, ".$mena_db->db_site_name.".currency
        where ".$mena_db->db_invoice_name.".products.physical_category_id = %i
        and ".$mena_db->db_invoice_name.".products.product_id = ".$mena_db-
        >db_invoice_name.".products_prices.product_id
        and ".$mena_db->db_invoice_name.".products_prices.currency_id = %i
        and ".$mena_db->db_site_name.".currency.currency_id = %i
        and ".$mena_db->db_invoice_name.".products.business_id = %i
        and ".$mena_db->db_invoice_name.".products.country_id = %i
        and ".$mena_db->db_invoice_name.".products.available = 1
        and ".$mena_db->db_invoice_name.".products.product_id =
        %i;", $id_kategorie, $default_currency, $default_currency, $id_podnikania, $country_info-
        >country_id, $id_3)->fetch();

        // Kontrola, či sa podarilo nájsť v databáze daný produkt. V prípade, že je hodnota
        rôzna od prázdneho reťazca, hru sa podarilo nájsť.
        if($hra!="")
        {
            // Kontrola, či je aj ďalší parameter URL číslo.
            if(is_numeric($id_4))
            {
                // Kontrola, či je produkt k dispozícii aj v užívateľom nastavenom jazyku.
                $moznost_predavania=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT
                product_id FROM physical_products where product_id = %i and parent_id = %i and
                available = 1;", $id_4, $hra->physical_product_parent_id)->fetch();

                // V prípade, že dotaz vráti hodnotu rôznu od prázdneho reťazca, produkt je
                dostupný v danej jazykovej verzii.

                if($moznost_predavania!="")
                {
                    // Zistenie názvu produktu v užívateľom zvolenom jazyku.

                    $nazov_hry=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT ".$mena_db-
                    >db_site_name.".lang_admin"."$this->language"."full_text from ".$mena_db-
                    >db_site_name.".physical_products, ".$mena_db-
                    >db_site_name.".lang_admin"."$this->language." where ".$mena_db-
                    >db_site_name.".physical_products.product_id = %i and ".$mena_db-
                    >db_site_name.".physical_products.product_name = ".$mena_db-
                    >db_site_name.".lang_admin"."$this->language"."constant;", $id_4)->fetch()-
                    >full_text;

                    // Vypísanie úspechovej hlášky o úspešnom vložení produktu do košíka.

                    $this->flashMessage($this->mute->getText('GAMES_FLASHMESSAGE_SUCCESS_PART_1')."
                    ".$nazov_hry." ".$this->mute->getText('GAMES_FLASHMESSAGE_SUCCESS_PART_2'));
                }
            }
        }
    }
}

```



```

// Uloženie informácie o objednanom produkte do seassions

$namespace=NEnvironment::getSession('ordered_games');
$nazov_produkту="product_".$hra->product_id."_".$id_4;
$nazov_jazykoveho_produkту="language_product_".$id_4;
$namespace->$nazov_produkту=1;
$namespace->$nazov_jazykoveho_produkту=1;
}
else
{
    // Nastala chyba.

    $this->flashMessage($this->mute->getText('GAMES_FLASHMESSAGE_ERROR'));
}
else
{
    // Nastala chyba.
    $this->flashMessage($this->mute->getText('GAMES_FLASHMESSAGE_ERROR'));
}
else
{
    // Nastala chyba.
    $this->flashMessage($this->mute->getText('GAMES_FLASHMESSAGE_ERROR'));
}
else
{
    // Nastala chyba.
    $this->flashMessage($this->mute->getText('GAMES_FLASHMESSAGE_ERROR'));
}
}
}

```

Tabuľka 3: Pridávanie produktu do košíka (zdrojový kód)³⁸

4.3.2 Prechod do košíka (prvý krok)

Po pridaní produktu do košíka užívateľom sa možnosť pridanie produktu do košíka automaticky zmení na možnosť prechodu do košíka. Spomínanú možnosť užívateľ využíva v prípade, že je kladne rozhodnutý o kompletnosti svojho nákupu. Zmenu možností ilustruje obrázok 14.



Obrázok 14: Ilustrácia prechodu do košíka³⁹

Po prechode do košíka má užívateľ dve možnosti:

- vykonať štandardnú objednávku,

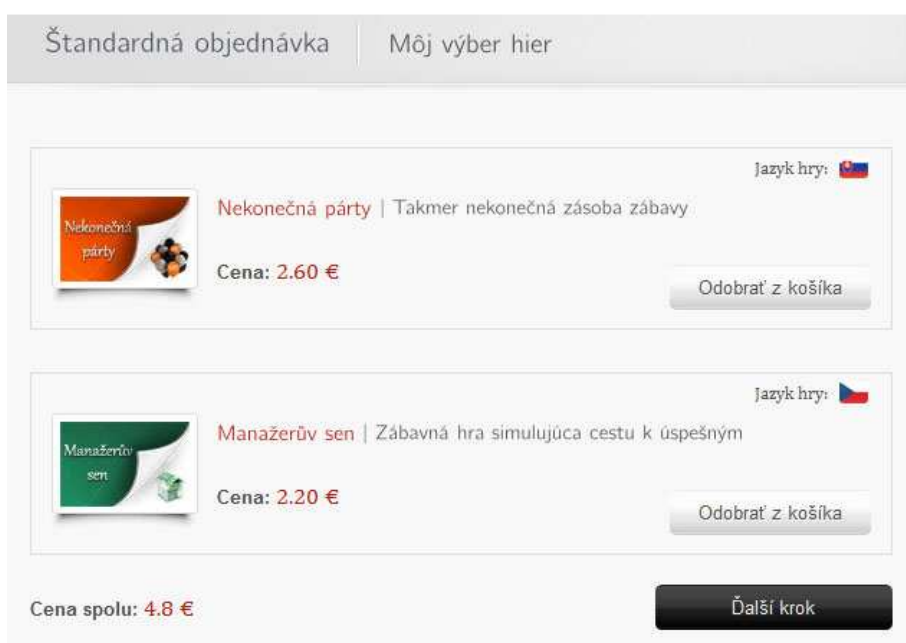
³⁸ Zdroj: vlastný

³⁹ Zdroj: vlastný

- využiť zľavový kupón (užívateľ ním už musí disponovať).

Pre zjednodušenie sa diplomová práca ďalej zaoberá vetvou procesu, ktorá spracováva štandardnú objednávku.

Systém po zobrazení košíka načíta údaje uložené v seassions a zistí všetky produkty, ktoré si užívateľ do košíka vložil. Následne dôjde k sčítaniu finálnej hodnoty objednávky a k vykresleniu informácií užívateľovi prostredníctvom šablóny. Obrázok 15 zobrazuje vykreslená šablóna košíka (prvý krok).



Obrázok 15: Košík (prvý krok)⁴⁰

V tabuľke č. 4 je uvedená ukážka zdrojového kódu, ktorý získava potrebné informácie zo seassions, kalkuluje hodnotu objednávky a posúva potrebné údaje do zobrazovanej šablóny.

```
// Získanie prístupu do seassions.
$namespace=NEEnvironment::getSession('ordered_games');
// Nastavenie hodnoty počítadlo pocet na 0.
$pocet=0;
```

⁴⁰ Zdroj: vlastný

```

// Príprava poľa, do ktorého sa budú vkladať objednané produkty.
$ordered_games[]="";

foreach($namespaceas$key=>$value)
{
    // Rozparovanie poľa a získanie ID produktov.

    if((substr($key,0,8)=="product_")and($value==1))
    {
        // Navýšenie počtu objednaných produktov.

        $pocet++;

        // Príprava rozparovanie poľa s produktom a jeho objednanou jazykovou veriou.

        $idcka=(substr($key,8));

        // Zistenie pozície oddelovača v poli.

        $pos_oddelovac=strpos($idcka,"_");

        // Zistenie ID produktu.

        $product_id=$idcka;

        // Zistenie jazykovej verzie produktu.

        $language_id=(substr($idcka,$pos_oddelovac+1));

        // Uloženie výsledkom do predpripraveného poľa.

        $ordered_games[$product_id]=$language_id;
    }
}

// Odoslanie informácie o tom, že produkty boli objednané do templatu.
if($pocet>0)
{
    // Nastavenie prepínača objednanosti produktov na hodnotu "objednané".

    $this->template->objednane_hry=1;
}
else
{
    // // Nastavenie prepínača objednanosti produktov na hodnotu "neobjednané".

    $this->template->objednane_hry=0;
}

// Kontrola, či sa našlo aspoň jedna objednaná hra pre potreby vykreslenie prehľadu
objednávky (nastavenia celkovej ceny objednávky a pod.).

if($pocet>0)
{
    // Nastavenie celkovej ceny objednávky na hodnotu 0.

    $cena_spolu=0;

    // Prebehnutie celého poľa, ktoré je uložené v seassions a získanie údajov o
    objednaných produktoch.

    foreach($ordered_games as $key=>$value)
    {
        if($key>0)
        {

            // Rozparovanie kľuča a zistenie pozície oddelovača.

```

```

$pos_oddelovac=strpos($key, "_");

// Zistenie ID produktu.

$product_id=(substr($key,0,$pos_oddelovac));

// Databazový dotaz na zistenie údajov o produkte.

$hra=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT ".$mena_db->db_invoice_name.">products.product_id,
".$mena_db->db_invoice_name.">products_prices.price,
".$mena_db->db_invoice_name.">products.default_sell_physical_product_id,
".$mena_db->db_invoice_name.">products.physical_product_parent_id,
".$mena_db->db_invoice_name.">products.physical_category_id,
".$mena_db->db_site_name.">currency.currency_symbol,
(SELECT ".$mena_db->db_site_name.">lang_admin.">$this->language.">.full_text from
".$mena_db->db_site_name.">.physical_products, ".$mena_db->db_site_name.">.lang_admin.">$this->language." where ".$mena_db->db_site_name.">.physical_products.product_id = %i and ".$mena_db->db_site_name.">.physical_products.product_name = ".$mena_db->db_site_name.">.lang_admin.">$this->language.">.constant) as product_name,
(SELECT ".$mena_db->db_site_name.">lang_admin.">$this->language.">.full_text from
".$mena_db->db_site_name.">.physical_products, ".$mena_db->db_site_name.">.lang_admin.">$this->language." where ".$mena_db->db_site_name.">.physical_products.product_id = %i and ".$mena_db->db_site_name.">.physical_products.product_promo = ".$mena_db->db_site_name.">.lang_admin.">$this->language.">.constant) as product_promo,
(SELECT ".$mena_db->db_site_name.">lang_admin.">$this->language.">.full_text from
".$mena_db->db_site_name.">.physical_products, ".$mena_db->db_site_name.">.lang_admin.">$this->language." where ".$mena_db->db_site_name.">.physical_products.product_id = %i and ".$mena_db->db_site_name.">.physical_products.product_description = ".$mena_db->db_site_name.">.lang_admin.">$this->language.">.constant) as description,
(SELECT ".$mena_db->db_site_name.">lang_admin.">$this->language.">.full_text from
".$mena_db->db_site_name.">.physical_products, ".$mena_db->db_site_name.">.lang_admin.">$this->language." where ".$mena_db->db_site_name.">.physical_products.product_id = %i and ".$mena_db->db_site_name.">.physical_products.product_reference = ".$mena_db->db_site_name.">.lang_admin.">$this->language.">.constant) as product_reference,
(SELECT ".$mena_db->db_site_name.">lang_admin.">$this->language.">.full_text from
".$mena_db->db_site_name.">.physical_products, ".$mena_db->db_site_name.">.lang_admin.">$this->language." where ".$mena_db->db_site_name.">.physical_products.product_id = %i and ".$mena_db->db_site_name.">.physical_products.product_reference_name = ".$mena_db->db_site_name.">.lang_admin.">$this->language.">.constant) as product_reference_name
FROM ".$mena_db->db_invoice_name.">products, ".$mena_db->db_invoice_name.">products_prices, ".$mena_db->db_site_name.">currency, ".$mena_db->db_site_name.">.physical_products
where ".$mena_db->db_invoice_name.">products.product_id = ".$mena_db->db_invoice_name.">products_prices.product_id
and ".$mena_db->db_invoice_name.">products_prices.currency_id = %i
and ".$mena_db->db_site_name.">currency.currency_id = %i
and ".$mena_db->db_invoice_name.">products.business_id = %i
and ".$mena_db->db_invoice_name.">products.country_id = %i
and ".$mena_db->db_invoice_name.">products.product_id = %i;", $value, $value, $value, $value, $value, $default_currency, $default_currency, $id_podniku, $country_info->country_id, $product_id)->fetch();

// Každý produkt má svoj ilustračný obrázok. Dotaz pre zistenie ID tohto obrázka.

$promo_photo_id=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT photo_id FROM physical_products_promo_photos where product_id = %i;", $value)->fetch()->photo_id;

$skatka_jazyka=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT language_url FROM languages, physical_products where languages.language_id = physical_products.product_language and physical_products.product_id = %i;", $value)->fetch()->language_url;

// Vytvorenie poľa s objednanými hrami. Pole sa ďalej využije v template.

$games_language_product[$key]=$value;
$games_product_name[$key]=$hra->product_name;
$games_product_promo[$key]=$hra->product_promo;
$games_skratka_jazyka[$key]=$skatka_jazyka;

```

```

$games_cena[$key]=$hra->price;
$games_skratka_mena[$key]=$hra->currency_symbol;
$games_promo_photo[$key]=$promo_photo_id;
$games_category[$key]=$hra->physical_category_id;
$games_ciste_id[$key]=$product_id;

// Spočítanie celkovej ceny všetkých objednaných hier.

$cena_spolu=$cena_spolu+$hra->price;

}
}

// Posunutie poľa objednaných hier do templatu, pre účely jeho vyhreslenia užívateľovi.

$this->template->games_language_product=$games_language_product;
$this->template->games_product_name=$games_product_name;
$this->template->games_product_promo=$games_product_promo;
$this->template->games_skratka_jazyka=$games_skratka_jazyka;
$this->template->games_promo_photo=$games_promo_photo;
$this->template->games_cena=$games_cena;
$this->template->games_skratka_mena=$games_skratka_mena;
$this->template->games_category=$games_category;
$this->template->games_ciste_id=$games_ciste_id;

// Posunutie celkovej ceny do templatu.

$this->template->cena_spolu=$cena_spolu;

// Posunutie skratky relevantnej meny do templatu.

$this->template->mena=$hra->currency_symbol;

// Posunutie poľa objednaných hier do templatu.

$this->template->ordered_games=$ordered_games;
}

```

Tabuľka 4: Vykreslenie košíka - prvý krok (zdrojový kód)⁴¹

Užívateľ pred sebou teraz vidí všetky produkty, ktoré si objednal. Má možnosť skontrolovať svoj výber a prípadne odobrať produkt z košíka. V prípade, že sa užívateľ rozhodne produkt odstrániť z košíka, vykoná sa podobná akcia ako pri pridávaní produktu do košíka s tým rozdielom, že tentokrát bude záznam o produkte zo sessions odstránený a zobrazí sa hláška odstránení produktu z košíka.

4.3.3 Prechod do košíka (druhý krok)

Následne potom, ako užívateľ skontroloval svoju objednávku, môže prejsť na ďalší krok kliknutím na príslušné tlačidlo. V nasledujúcom kroku má systém dve možnosti, ako sa zachovať. Správanie systému závisí na tom, či je užívateľ vykonávajúci objednávku práve prihlásený. V prípade, že užívateľ prihlásený nie je, v ďalšom kroku sa mu zobrazí možnosť registrácie alebo prihlásenia, ktorú ukazuje obrázok č. 16.

⁴¹ Zdroj: vlastný

Nákupný košík | Dokončenie objednávky

V prípade že už máš svoj účet na Secret Games, môžeš sa **prihlásiť tu**.

Prezývka:

E-mail:

Pohlavie: -- Vyber tvoje pohlavie --

Koľko je jedna plus dva? (Prosím, odpovedajte číslom):

☐ Súhlasím s obchodnými podmienkami

☒ Mám záujem o newsletter

[Zobraziť obchodné podmienky](#)

Obrázok 16: Košík - registračný formulár (druhý krok - užívateľ nie je prihlásený)⁴²

V prípade, že užívateľ je prihlásený, ostáva už len potvrdiť súhlas so zmluvnými podmienkami a dokončiť objednávku kliknutím na príslušné tlačidlo. V tejto situácii sa už registračný formulár nezobrazuje.

4.3.4 Registračný formulár

V predchádzajúcej podkapitole bol spomínaný registračný formulár potrebný na vytvorenie účtu, prostredníctvom ktorého sú užívateľom sprístupňované kúpené produkty.

V zdrojovom kóde, ktorý sa nachádza nižšie v tabuľke č. 5 je napísané vytvorenie registračného formulára pomocou Nette. Vytvorený formulár je následne možné zavolať z prislúchajúceho templatu. Ako je vidieť zo zdrojového kódu, Nette výrazným spôsobom zvyšuje efektivitu práce programátora. Rôzne verifikáčné funkcie na kontrolu dĺžky zadávaného reťazca, jeho validitu, povolené znaky a podobne by v čistom PHP

⁴² Zdroj: vlastný

bolo potrebné zložiť naprogramovať. Použitím Nette a jeho pravidiel (addRule) sa vytvorenie zabezpečeného a moderného formulára stalo otázkou niekoľkých riadkov.

```
protectedfunctioncreateComponentUserOrderRegistration()
{
    //Jazykové premenné používané vo formulári.

    $mail=$this->mute->getText('REGISTRATION_USERREGISTRATION_MAIL');
    $wrong_mail=$this->mute->getText('REGISTRATION_USERREGISTRATION_WRONG_MAIL');
    $long_mail=$this->mute->getText('REGISTRATION_USERREGISTRATION_LONG_MAIL');
    $licence=$this->mute->getText('REGISTRATION_USERREGISTRATION_LICENCE');
    $licence_filled=$this->mute->getText('REGISTRATION_USERREGISTRATION_LICENCE_FILLED');
    $nick=$this->mute->getText('REGISTRATION_USERREGISTRATION_NICK');
    $nick_filled=$this->mute->getText('REGISTRATION_USERREGISTRATION_NICK_FILLED');
    $long_nick=$this->mute->getText('REGISTRATION_USERREGISTRATION_LONG_NICK');
    $dokoncit=$this->mute->getText('ORDER_USERREGISTRATION_DOKONCIT');
    $final_number_control=$this->mute->
    >getText('REGISTRATION_USERREGISTRATION_FINAL_NUMBER_CONTROL');
    $antispam=$this->mute->getText('REGISTRATION_USERREGISTRATION_ANTISPAM');
    $newsletter=$this->mute->getText('REGISTRATION_USERREGISTRATION_NEWSLETTER');
    $pohlavie=$this->mute->getText('REGISTRATION_USERREGISTRATION_POHLAVIE');
    $vyberte_pohlavie=$this->mute->
    >getText('REGISTRATION_USERREGISTRATION_VYBERTE_POHLAVIE');
    $pohlavie_filled=$this->mute->getText('REGISTRATION_USERREGISTRATION_POHLAVIE_FILLED');

    // Získani prístupu do seassions.

    $namespace=NEEnvironment::getSession('userregistration');

    // Vytvorenie registračného formulára.

    $form=newNAppform;

    $form->addText('nick',$nick,60)
    ->setValue($namespace->nick)
    ->addRule(NForm::FILLED,$nick_filled)
    ->addRule(NForm::MAX_LENGTH,$long_nick,60);

    $form['nick']->getControlPrototype()->class('inputbox');

    $form->addText('email',$mail,128)
    ->setValue($namespace->email)
    ->addRule(NForm::EMAIL,$wrong_mail)
    ->addRule(NForm::MAX_LENGTH,$long_mail,128);

    $form['email']->getControlPrototype()->class('inputbox');

    // Zistenie užívateľom zvoleného jazyka.

    $this->language=$this->getParam('language');

    // Získanie informácií do poľa na vyplnenie číselníka pre pohlavie.

    $gender=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT *
    FROM
    user_gender, lang_". $this->language."
    WHERE user_gender.name = lang_". $this->language." .constant
    order by gender_id;")->fetchassoc('gender_id');
    $gender_array[0]=$vyberte_pohlavie;
    foreach($genderas$gender)
    {
        $gender_array[$gender->gender_id]=$gender->full_text;
    }
}
```

```

$form->addSelect('gender',$pohlavie,$gender_array)
->setValue($namespace->gender)
->addRule(NForm::EQUAL,$pohlavie_filled,0);

$form['gender']->getControlPrototype()->class('inputbox');

$form->addTextArea('antispam',$antispam,38,5)
->addRule(NForm::EQUAL,$final_number_control,3);

$form['antispam']->getControlPrototype()->class('inputbox');
$form['antispam']->getLabelPrototype()->class('textarea_label');

$form->addCheckBox('licence',$licence)
->setValue($namespace->licence)
->addRule(NForm::FILLED,$licence_filled);

$form['licence']->getControlPrototype()->class('checkbox_input');
$form['licence']->getLabelPrototype()->class('checkbox_label');

$form->addCheckBox('newsletter',$newsletter)
->setValue(1);

$form['newsletter']->getControlPrototype()->class('checkbox_input');
$form['newsletter']->getLabelPrototype()->class('checkbox_label');

$form->addSubmit('registration',$dokonciti);
$form['registration']->getControlPrototype()->class('dark_button_order_submit_button');

$form->onSuccess[]=callback($this,'processUserOrderRegistration');

return $form;
}

```

Tabuľka 5: Vytvorenie registračného formulára (zdrojový kód)⁴³

4.3.5 Dokončenie objednávky

Následne potom ako registračný formulár, ktorého kód sa nachádza v tabuľke 5 zabezpečil korektné vloženie údajov od užívateľa, je možné formulár potvrdiť a objednávku dokončiť.

Zdrojový kód, ktorý obsahuje tabuľka č. 6 má na starosti vytvorenie nového užívateľského účtu.

```

function processUserOrderRegistration($form)
{
    // Naplnenie jazykových premenných.

    $neuspesne=$this->mute->getText('REGISTRATION_USERREGISTRATION_NEUSPESNE');
    $uspesne=$this->mute->getText('REGISTRATION_USERREGISTRATION_USPESNE');
    $neuspesne_nick=$this->mute->getText('REGISTRATION_USERREGISTRATION_NEUSPESNE_NICK');
    $zly_nick=$this->mute->getText('REGISTRATION_USERREGISTRATION_ZLY_NICK');

    // Získanie jazyka z URL.

```

⁴³ Zdroj: vlastný


```

$this->language=$this->getParam('language');

// Zistenie ID krajiny, z ktorej sa produkt objednáva.
$country_info=NEnvironment::getSession('country_info');
$country_id=$country_info->country_id;

// Zistenie časovej stopy objednávky.
$timestamp=time();

// Zistenie typu podnikania (Z akého frontendu prišla objednávka).
$id_podnikania=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT business_id FROM
global_settings;")->fetch()->business_id;

// Zistenie mien databáz.
$mena_db=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT * FROM
global_settings;")->fetch();

// Získanie hodnôt z formulára.
$values=$form->getValues();

// Získanie prístupu do sessions.
$namespace=NEnvironment::getSession('userregistration');

// Uloženie údajov o neregistrovanom užívateľovi pre prípad, že by ich bolo potrebné
opätovne získať a aby ich užívateľ nemusel zadávať znova.
$namespace->nick=$values['nick'];
$namespace->email=$values['email'];
$namespace->licence=$values['licence'];
$namespace->gender=$values['gender'];

// Kontrola, či e-mail zadaný pri registrácii nie je už používaný iným užívateľom.
$zistenie_mailu=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT username FROM
users WHERE username = %s;", $namespace->email)->fetch();

if($zistenie_mailu=="")
{
    // Kontrola validnosti nickname
    $validNick="/^[a-zA-Z0-9_-]{1,255}$/";
    if(preg_match($validNick,$namespace->nick))
    {
        // Zistenie, či už daný nickname v databáze existuje, aby nedošlo k duplicite.
        $zistenie_nicku=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT nickname FROM
users WHERE nickname = %s;", $namespace->nick)->fetch();

        if($zistenie_nicku=="")
        {
            $this->flashMessage($uspesne);

            // Príprava hesla pre užívateľa a jeho následný hash.
            $config=NEnvironment::getConfig('security');
            $heslo=substr(md5(rand(1000,99999999)),0,8);
            $heslo_hash=sha1($heslo.$config->salt);

            // Vytvorenie náhodného reťazca, ktorý sa použije pre účel verifikácie e-mailu
            užívateľa.
            $mail_verification_code=rand(1000,99999999);

```

```

$mail_verification_code=md5($mail_verification_code);

// Detekcia jazyka na základe URL a následné získanie ID jazyka.
$env_lang=newDetectLang();
$language_id=$env_lang->urlToLangId($this->language);

// Zistenie prednastavenej meny krajiny, do ktorej bol užívateľ zaradený.
$default_currency=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT currency_id
FROM countries
where country_id = %i;", $country_info->country_id)->fetch()->currency_id;

// Uloženie záznamu o užívateľovi do databázy.
$db=dibi::getConnection(DataBase::MAIN_DB);
$db->query("Insert into users (username, pass, mail_verification_code,
mail_verification, nickname,
language_id, currency_id, registration_id_country, newsletter, gender)
values (%s, %s, %s, %i, %s,
%i, %i, %i, %i, %i);",
$namespace->email, $heslo_hash, $mail_verification_code, 0, $namespace->nick,
$language_id, $default_currency, $country_id, $values['newsletter'], $values['gender']);

// Zistenie posledného vloženého ID užívateľa do databázy.
$add_user_id=mysql_insert_id();

// Uloženie všetkých údajov o zákazníkovi aj do tabuľky so zmenami údajov užívateľov.
$db->query("Insert into changes_users (username, pass, mail_verification_code,
mail_verification, nickname,
language_id, currency_id, change_by_user_id, timestamp, registration_id_country,
newsletter, gender)
values (%s, %s, %s, %i, %s,
%i, %i, %i, %i, %i, %i, %i);",
$namespace->email, $heslo_hash, $mail_verification_code, 0, $namespace->nick,
$language_id, $default_currency, $add_user_id, $timestamp, $country_id, $values['newsletter'],
$values['gender']);

// Pridanie užívateľovi role zákazníka prostredníctvom záznamu v tabuľke user_roles
$db->query("Insert into users_roles (user_id, role_id)
values (%i, 2);",
$add_user_id);

```

Tabuľka 6: Vytvorenie nového užívateľského účtu (zdrojový kód)

Následne ako bol vytvorený nový užívateľský účet, je potrebné uloženie objednávky do systému. Práve spomenutú činnosť zabezpečuje kód obsiahnutý v tabuľke 7.

```

// *** Uloženie objednaných produktov do databázy. ***

// Získanie prístupu do sessions.
$namespace_games=NEnvironment::getSession('ordered_games');

// Nastavenie počítadla pocet na hodnotu 0.
$pocet=0;

```

```

foreach($namespace_gamesas$key=>$value)
{
    if((substr($key,0,8)=="product_")and($value==1))
    {
        $pocet++;

        // Pripravenie rozprarsovaného poľa s produktom a jeho objednanou jazykovou verziou.

        $idcka=(substr($key,8));

        // Zistenie pozície oddeľovača.

        $pos_oddelovac=strpos($idcka,"_");

        // Žistenie ID produktu.

        $product_id=$idcka;

        // Zistenie jazykovej verzie produktu.

        $language_id=(substr($idcka,$pos_oddelovac+1));

        // Získané výsledky sú uložené do poľa.

        $ordered_games[$product_id]=$language_id;
    }
}

// Cena objednávky je nastavená na hodnotu 0.

$scena_objednavky=0;

foreach($ordered_gamesas$key=>$value)
{
    // Rozparsovanie URL adresy.

    $pos_oddelovac=strpos($key,"_");

    // Zistenie ID produktu.

    $product_id=(substr($key,0,$pos_oddelovac));

    // Zistenie jazykovej verzie produktu.

    $language_product_id=(substr($key,$pos_oddelovac+1));

    $ok=0;

    if((is_numeric($product_id))and((is_numeric($language_product_id))))
    {
        $ok=1;
    }

    // Objednanie produktu
    // Zistenie, či daný produkt existuje.
    if($ok==1)
    {
        // Zistenie informácií o produkte.

        $hra=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT ".$mena_db-
        >db_invoice_name.".products.product_id,
        ".$mena_db->db_invoice_name.".products.physical_product_parent_id,
        ".$mena_db->db_invoice_name.".products_prices.price,
        (SELECT ".$mena_db->db_site_name.".lang_admin".$this->language.".full_text from
        ".$mena_db->db_site_name.".physical_products, ".$mena_db-
        >db_site_name.".lang_admin".$this->language." where ".$mena_db-
        >db_site_name.".physical_products.product_id = ".$mena_db-
        >db_invoice_name.".products.default_sell_physical_product_id and ".$mena_db-
        >db_site_name.".physical_products.product_name = ".$mena_db-
        >db_site_name.".lang_admin".$this->language.".constant) as product_name

```

```

FROM ".$mena_db->db_invoice_name.".products, ".$mena_db->db_invoice_name.".products_prices, ".$mena_db->db_site_name.".currency
where ".$mena_db->db_invoice_name.".products.product_id = ".$mena_db->db_invoice_name.".products_prices.product_id
and ".$mena_db->db_invoice_name.".products_prices.currency_id = %i
and ".$mena_db->db_site_name.".currency.currency_id = %i
and ".$mena_db->db_invoice_name.".products.business_id = %i
and ".$mena_db->db_invoice_name.".products.available = 1
and ".$mena_db->db_invoice_name.".products.country_id = %i
and ".$mena_db->db_invoice_name.".products.product_id = %i;",$default_currency,$default_currency,$id_podnikania,$country_info->country_id,$product_id)->fetch();

// Zistenie, či sa daný produkt našiel.
if($hra!="")
{
    // Overenie, či je ID jazyka číslo.

    if(is_numeric($language_product_id))
    {
        // Zistenie, či sa produkt môže predávať v požadovanom jazyku.

        $moznost_predavania=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT product_id
FROM physical_products where product_id = %i and parent_id = %i and available =
1;",$language_product_id,$hra->physical_product_parent_id)->fetch();
        if($moznost_predavania!="")
        {
            // Spočítanie ceny objednávky.

            $scena_objednavky=$scena_objednavky+$hra->price;

            // Uloženie objednaných hier do databázy (priradia sa k užívateľovi).

            dibi::getConnection(DataBase::INVOICES)->query("Insert into customers_products
(product_id, language_product_id, business_id, country_id, user_id, add_user_id,
order_timestamp, default_price, free_product_to_customer)
values (%i, %i, %i, %i, %i, %i, %i, %i, %i);",
$product_id,$language_product_id,$id_podnikania,$country_id,$add_user_id,$add_user_id,$timestamp,1,0);
        }
        else
        {
            // Nič sa nevypisuje.
        }
    }
    else
    {
        // Nič sa nevypisuje.
    }
}
else
{
    // Nič sa nevypisuje.
}
}
else
{
    // Nič sa nevypisuje.
}
}
else
{
    // Nič sa nevypisuje.
}
}

// Uloženie súhrnných informácií o objednávke do databázy.
dibi::getConnection(DataBase::INVOICES)->query("Insert into orders (business_id,
country_id, user_id, add_user_id, order_timestamp, currency_id, price)
values (%i, %i, %i, %i, %i, %i, %f);",
$id_podnikania,$country_id,$add_user_id,$add_user_id,$timestamp,$default_currency,$scena_objednavky);
$order_id=mysql_insert_id();

```

Tabuľka 7: Uloženie objednávky do systému (zdrojový kód)⁴⁴

⁴⁴ Zdroj: vlastný

4.3.6 Prihlásenie a odoslanie verifikačného e-mailu

Následne potom, ako bola nová objednávka vykonaná a užívateľské konto vytvorené, je užívateľ automaticky prihlásený a presmerovaný na podstránku prezentujúcu základné funkcie účtu a potvrdzujúcu úspešnosť objednávky. Súčasne je potrebné užívateľa informovať o uskutočnených úkonoch. Dôležité je, aby bol užívateľ utvrdený aj v budúcnosti objaviteľným spôsobom v tom, že svoje kroky počas objednávania vykonal správne a systém korektne prijal jeho požiadavku. Práve z tohto dôvodu prebieha proces informovania užívateľa aj prostredníctvom automaticky generovaného e-mailu, ktorý okrem iného obsahuje prihlasovacie údaje k účtu. Zdrojový kód, ktorý odosiela automatický verifikačný e-mail a automaticky prihlasuje užívateľa sa nachádza v tabuľke č. 8.

```
// *** E-mail odosielaný užívateľovi po objednávke ***  
  
// Naplnenie jazykových konštánt.  
  
$predmet_vytvorenie=$this->mute->getText('ORDER_REGISTRATION_VITAJ');  
  
// Vytvorenie templatu a jeho naplnenie premennými.  
  
$template=$this->createTemplate();  
$template->setFile(APP_DIR."/templates_mail/".$this->language."_userOrder.latte");  
$template->meno=$namespace->nick;  
$template->stranka=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT site_url FROM  
global_settings;")->fetch()->site_url;  
$template->heslo=$heslo;  
$template->verifikacny_kod=$mail_verification_code;  
$template->prihlasovacie_meno=$namespace->email;  
$template->jazyk=$this->language;  
$template->slogan=$this->mute->getText('BASE_FOOTER_MAIL-CLIENT_SLOGAN');  
$template->pozdrav=$this->mute->getText('BASE_FOOTER_MAIL-CLIENT_POZDRAV');  
$template->pozdrav_projekt=$this->mute->getText('BASE_FOOTER_MAIL-CLIENT_POZDRAV-  
PROJEKT');  
$template->automaticke_odoslanie=$this->mute->getText('BASE_FOOTER_MAIL-  
CLIENT_AUTOMATICKA-SPRAVA');  
$template->order_id=$order_id;  
  
// Odoslanie e-mailu.  
  
$mail=newNMail;  
$mail->setFrom(dibi::getConnection(DataBase::MAIN_DB)->query("SELECT mail FROM mail  
WHERE mail_type = 1 and country_id = %i;", $country_id)->fetch()->mail)// odosielaťel  
spravy  
->addTo($namespace->email)// Prijemca spravy  
->setSubject($predmet_vytvorenie)// predmet spravy  
->setHtmlBody($template)// Telo e-mailu  
->send();// Odoslanie e-mailu  
  
// *** Automatické prihlásenie užívateľa a jeho persmerovanie na úvodnú stránku pre  
prihláseného užívateľa. ***  
  
$this->getUser()->login($namespace->email,$heslo);  
  
// Získanie prístupu do seassions.  
  
$namespace=NEEnvironment::getSession('userregistration');
```

```
// Vymazanie seassions.

$namespace->remove();
$namespace_games->remove();

// Presmerovanie užívateľa na stránku s úspechovou hláškou o objednávke.

$this->redirect('MyAccount:OrderDone');
}
else
{
$this->flashMessage($neuspesne_nick);
$this->redirect('this');
}
}
else
{
$this->flashMessage($zly_nick);
$this->redirect('this');
}
}
else
{
// Objednávka nebola dokončená.
$this->flashMessage($neuspesne);
$this->redirect('this');
}
}
```

Tabuľka 8: Odoslanie verifikačného e-mailu (zdrojový kód)⁴⁵

Vzhľadom k tomu, že v súčasnej dobe je na elektronickom trhu silná konkurencia, nie je možné zanedbať ani detail, akým je dizajn e-mailu. Bežný textový e-mail bez grafického dizajnu by znižoval dojem, ktorý užívateľ o spoločnosti priebežne získava. Snahou je skĺbiť vysoko profesionálne a priateľské pôsobenie. Pre potreby e-mailového dizajnu je nutné odosielať html e-mail, ktorého ukážku obsahuje tabuľka č. 9. Oranžové reťazce predstavujú premenné posúvané z príslušného presentera.

```
<table
width="510"height=""border="0"cellspacing="0"cellpadding="0"background="mail_shadow.png"
style="padding-bottom:10px; background-repeat:no-repeat; background-position:-5px
bottom;">
<tr><td>

<table width="510"height=""border="0"cellspacing="0"cellpadding="0"style="border:solid
1px #ddd; padding:6px;">
<tr><td>

<table width="500"height=""border="0"cellspacing="0"cellpadding="0"style="background-
repeat:no-repeat; background-position:center bottom; background-color:#fff;">
<tr><td>

<table
width="400"height="58"border="0"cellspacing="0"cellpadding="0"background="mail_logo.png"
style="margin-left:110px; margin-right:110px; margin-top:5px; background-repeat:no-
repeat; display:block;">
<tr><td style="display:block;"><a style="display:block; width:380px; height:58px;
color:#bc2126; text-decoration:none"href="http://www.secret-games.com"></a></td></tr>
</table>
```

⁴⁵ Zdroj: vlastný

```

<table>
<tr><td style="width:600px; text-align:center; font-family:Cambria; font-size: 12pt;
color: #888888; padding-bottom:30px;">{$slogan}</td></tr>
</table>
<table style="margin-left:30px; margin-right:30px;">
<tr>
<td>
<div style="width:550px; font-family:Calibri; color:black; text-align:justify;">

<p>Ahoj, <strong>{$meno}</strong>!</p>

<p>Tvoja objednávka <strong>č. {$order_id}</strong> na <a href="http://secret-
games.com/{$jazyk}/" style="color:#b51d0f; text-decoration:none;">Secret Games</a>
bola <strong>úspešne prijatá.</strong></p>

<p>Všetky informácie o platbách, objednávkach, ako aj hry nájdeš vo svojom
<strong>novom osobnom účte.</strong> Veríme, že sa poriadne zabavíš.<span
style="font-family:Cambria; font-size:21px;">&#9786;</span></p>

<p><strong>Prístup do osobného účtu:</strong></p>
<ul style="list-style-type:none; padding:15px 0px 15px 20px; border:1px solid
#888888; width:270px;">
<li><strong>Stránka:</strong><a href="{$stranka}" style="color:#b51d0f; text-
decoration:none;" title="Osobný účet | Secret Games">{$stranka}</a></li>
<li><strong>Meno:</strong>{$prihlasovacie_meno}</li>
<li><strong>Heslo:</strong>{$heslo}</li>
</ul>
<p style="margin-bottom:40px;"><span style="font-size:10pt; font-style:italic;
color: #888888;">Z bezpečnostných dôvodov si môžete verifikovať svoj e-mail
kliknutím na <a href="{$stranka}/{$jazyk}/my-
account/information/{$verifikacny_kod}" style="color:#b51d0f; text-
decoration:none;" title="Verifikácia e-mailu | Secret Games">odkaz</a>.</span></p>

<p>{$pozdrav}</p>
<p>{$pozdrav_projekt}</p>

<p style="font-size:10pt; font-style:italic; color:
#888888;">{$automaticke_odoslanie}</p>
</div>
</td>
</tr>
</table>
<table
width="400"height="70"border="0"cellspacing="0"cellpadding="0"background="footer.png"sty
le="margin-top:10px; margin-left:100px;">
<tr><td></td></tr>
</table>
<table width="73"height="60"border="0"cellspacing="0"cellpadding="0"background="SG-
facebook_logo.png"style="margin-left:270px; margin-top:-30px;">
<tr><td><a href="http://www.facebook.com/SecretGames" style="display:block; height:60px;
width:73px;"><a></td></tr>
</table>
<table border="0"cellspacing="0"width="220"cellpadding="0"style="margin-left:200px;
margin-bottom:10px;">
<tr><td><p style="text-align:center;"><a style="text-decoration:none; color:#000000;
display:block;" href="http://www.facebook.com/SecretGames"><strong style="font-
family:Cambria; font-size:11pt; color: #bc2126">Secret Games <span
style="color:#000000;">na facebooku </span></strong><span style="color:#888888; font-
size:10pt; font-family:Calibri;"><br/>Súťaž s nami a získaj aktuálne informácie o nových
hrách...</span></a><span style="position:relative; top:20px; font-
size:12pt;"></p></td></tr>
</table>
</tr></td>
</table>
</tr></td>
</table>
</tr></td>
</table>

```

Tabuľka 9: HTML kód verifikačného e-mailu (zdrojový kód)⁴⁶

⁴⁶ Zdroj: vlastný

Náhľad dizajnu spomínaného HTML e-mailu ukazuje obrázok 17.



Obrázok 17: Náhľad verifikačného e-mailu⁴⁷

4.3.7 Zobrazenie užívateľského účtu

Potom, ako bola vykonaná objednávka a vytvorený nový užívateľský účet, je užívateľ presmerovaný práve do svojho nového účtu. V prípade, že sa jedná o prvú objednávku a je tým pádom možné hovoriť o registrácii, tak užívateľ nie je presmerovaný na štandardnú prvú podstránku účtu, ale na podstránku predstavujúcu základné funkcie systému.

Po prihlásení pribudol v pravom hornom rohu užívateľského rozhrania systému odkaz na zobrazenie účtu, ktorý ilustruje obrázok 18.

⁴⁷ Zdroj: vlastný



Obrázok 18: Odkaz na „Môj účet“⁴⁸

Ovládaciú časť primárnej podstránky užívateľského účtu, z ktorej má užívateľ prístup k jednotlivým funkciám systému, ilustruje obrázok 19.



Obrázok 19: Ovládacie prvky užívateľského konta⁴⁹

4.3.8 Užívateľská správa objednávok (zoznam objednávok)

Pomocou ovládacieho prvku „Moje hry“, ktorý bol spomenutý v predošlej podkapitole sa užívateľ dostane do rozhranie, kde môže pracovať so svojím objednanými produktami ako aj samotnými objednávkami.

Zdrojový kód, ktorý vykresľuje zoznam objednávok obsahuje tabuľka č. 10.

```
// Zistenie údajov o užívateľovi.
$user=NEnvironment::getUser();
if($user->isLoggedIn())
{
    // Užívateľ je prihlásený a preto môže byť zoznam objednávok vykreslený.
}
else
{
    // Užívateľ nie je prihlásený a bude presmerovaný na index.
    $this->redirect('Homepage:');
}

// Naplnenie jazykových premenných a ich poslanie do template.
```

⁴⁸ Zdroj: vlastný

⁴⁹ Zdroj: vlastný

```

$this->template->MYACCOUNT_ORDERS_TH_NUMBER=$this->mute-
>getText('MYACCOUNT_ORDERS_TH_NUMBER');
$this->template->MYACCOUNT_ORDERS_TH_TYPE=$this->mute-
>getText('MYACCOUNT_ORDERS_TH_TYPE');
$this->template->MYACCOUNT_ORDERS_TH_PRICE=$this->mute-
>getText('MYACCOUNT_ORDERS_TH_PRICE');
$this->template->MYACCOUNT_ORDERS_TH_STATUS=$this->mute-
>getText('MYACCOUNT_ORDERS_TH_STATUS');
$this->template->MYACCOUNT_ORDERS_TH_MANAGE=$this->mute-
>getText('MYACCOUNT_ORDERS_TH_MANAGE');
$this->template->MYACCOUNT_ORDERS_TH_PAYMENT=$this->mute-
>getText('MYACCOUNT_ORDERS_TH_PAYMENT');
$this->template->MYACCOUNT_ORDERS_H_ORDER_LIST=$this->mute-
>getText('MYACCOUNT_ORDERS_H_ORDER_LIST');
$this->template->MYACCOUNT_ORDERS_SPAN_STANDARD_TYPE=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_STANDARD-TYPE');
$this->template->MYACCOUNT_ORDERS_SPAN_ACTION_TYPE=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_ACTION-TYPE');
$this->template->MYACCOUNT_ORDERS_A_CANCEL=$this->mute-
>getText('MYACCOUNT_ORDERS_A_CANCEL');
$this->template->MYACCOUNT_ORDERS_A_DETAIL=$this->mute-
>getText('MYACCOUNT_ORDERS_A_DETAIL');
$this->template->MYACCOUNT_ORDERS_SPAN_DONE=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_DONE');
$this->template->MYACCOUNT_ORDERS_SPAN_VERIFICATION=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_VERIFICATION');
$this->template->MYACCOUNT_ORDERS_A_PAY=$this->mute->getText('MYACCOUNT_ORDERS_A_PAY');
$this->template->MYACCOUNT_ORDERS_SPAN_NO_MANAGE=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_NO-MANAGE');
$this->template->MYACCOUNT_ORDERS_A_YES_CANCEL_BUTTON=$this->mute-
>getText('MYACCOUNT_ORDERS_A_YES-CANCEL-BUTTON');
$this->template->MYACCOUNT_ORDERS_A_NO_CANCEL_BUTTON=$this->mute-
>getText('MYACCOUNT_ORDERS_A_NO-CANCEL-BUTTON');
$this->template->MYACCOUNT_ORDERS_A_PAYPAL_PAYMENT=$this->mute-
>getText('MYACCOUNT_ORDERS_A_PAYPAL-PAYMENT');
$this->template->MYACCOUNT_ORDERS_A_CARD_PAYMENT=$this->mute-
>getText('MYACCOUNT_ORDERS_A_CARD-PAYMENT');
$this->template->MYACCOUNT_ORDERS_SPAN_ACCOUNT_NUMBER=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_ACCOUNT-NUMBER');
$this->template->MYACCOUNT_ORDERS_SPAN_BANK_ADDRESS=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_BANK-ADDRESS');
$this->template->MYACCOUNT_ORDERS_SPAN_PRICE_SUM=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_PRICE-SUM');
$this->template->MYACCOUNT_ORDERS_SPAN_VAR_SYMBOL=$this->mute-
>getText('MYACCOUNT_ORDERS_SPAN_VAR-SYMBOL');
$this->template->MYACCOUNT_MYGAMES_DESCRIPTION=$this->mute-
>getText('MYACCOUNT_MYGAMES_DESCRIPTION');

// Zistenie parametra akcie z URL, na základe ktorého dochádza k vetveniu algoritmu.

$id_1=$this->getParam('id');
$this->template->id_1=$id_1;

// Zistenie parametra akcie z URL.

$id_2=$this->getParam('id_2');
$this->template->id_2=$id_2;

// Zistenie ID užívateľa a jeho role.

$user_name=$user->getId();
$user_id=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT user_id FROM users WHERE
username = %s;", $user_name)->fetch()->user_id;

// Zistenie krajiny, z ktorej užívateľ prichádza.

$country_info=NEnvironment::getSession('country_info');

// Zistenie predvolenej meny krajiny.
$default_currency=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT currency_id
FROM countries
where country_id = %i;", $country_info->country_id)->fetch()->currency_id;

// Zistenie názvu meny.

```

```

$default_currency_info=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT *
FROM currency
where currency_id = %i;", $default_currency)->fetch();

// Zistenie typu podnikania (Z akého frontendu prišla objednávka).

$id_podnikania=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT business_id FROM
global_settings;")->fetch()->business_id;

// Zistenie mien databáz.

$mena_db=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT * FROM
global_settings;")->fetch();

// Pokiaľ je id_1 prázdne, je požadované zistenie údajov o objednávkach.
if($id_1=="")
{
    // Zistenie počtu objednávok.
    $pocet_objednavok=dibi::getConnection(DataBase::INVOICES)->query("SELECT
count(order_id) as pocet FROM orders where user_id = %i and business_id = %i and
ignored = 0;", $user_id, $id_podnikania)->fetch()->pocet;
    if($pocet_objednavok>0)
    {
        // Do template je posunutá informácia, že objednávky existujú.

        $this->template->objednavky=1;

        // Zistenie všetkých objednávok, ktoré má užívateľ objednané.
        $orders=dibi::getConnection(DataBase::INVOICES)->query("SELECT * FROM orders where
user_id = %i and business_id = %i and ignored = 0 order by paid, order_timestamp
desc;", $user_id, $id_podnikania)->fetchassoc('order_id');

        // Posunutia pola užívateľových objednávok do template.

        $this->template->orders=$orders;
        $this->template->orders_2=$orders;
        foreach($orders as $orders)
        {

            // Zistenie meny objednávky.

            $currency_info=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT currency_symbol
FROM currency
where currency_id = %i;", $orders->currency_id)->fetch()->currency_symbol;

            $currency[$orders->order_id]=$currency_info;
        }
        $this->template->currency=$currency;
    }
    else
    {
        // Do template je posunutá informácia, že objednávky neexistujú.
        $this->template->objednavky=0;
    }
}

```

Tabuľka 10: Zoznamu objednávok (zdrojový kód)⁵⁰

⁵⁰ Zdroj: vlastný

Finálnu grafickú úpravu zoznamu objednávok zobrazuje obrázok 20.

Číslo	Typ	Cena	Stav	Správa	Platba
839	z akčného portálu	3.60 €		Detail / Zrušiť	Overuje sa. (max. 24 h)
809	štandardná	4.80 €		Detail / Zrušiť	Zaplatiť
255	štandardná	13.20 €		Detail / Zrušiť	Zaplatiť
2	štandardná	17.60 €		Detail / Zrušiť	Zaplatiť
828	štandardná	9.20 €		Detail	vykonaná
816	štandardná	2.00 €		Detail	vykonaná

Obrázok 20: Zoznam objednávok⁵¹

Systém rozoznáva tri stavy objednávok (ako je možné vidieť aj na obrázku 20):

- **Objednávka sa overuje** - jedná sa o situáciu, keď bol v objednávke uplatnený zľavový kupón a objednávka už bola v podstate zaplatená, avšak ešte nedošlo k overeniu platnosti kupónu administrátorom (modrý krížik).
- **Objednávka nezaplatená** - platba za objednávku zatiaľ nebola inkasovaná.
- **Objednávka vybavená** - platba za objednávku bola inkasovaná a užívateľovi bol úspešne pridelený prístup k objednaným produktom.

Užívateľ má v zozname objednávok základné možnosti, ktorým sú venované nasledujúce podkapitoly. Základné možnosti sú:

- zobrazenie detailu objednávky,
- zrušenie objednávky,
- zaplatenie objednávky.

⁵¹ Zdroj: vlastný

4.3.9 Detail objednávky

V detaile objednávky môže užívateľ nájsť informácie o jednotlivých produktoch, ktoré boli súčasťou objednávky.

Vzorku zdrojového kódu šablóny pre vykreslenie detailu objednávky ilustruje tabuľka č. 11. Oranžové reťazce predstavujú premenné a logické zápisy frameworku Nette pre šablónu (rozhodovacie bloky, cykly a podobne).

```
{foreach $orders_2 as $orders_2}
{if $id_1 == ""}
{** Jedna sa o zobrazenie objednavok *}
<a href="{link //MyAccount:Information}"title="{ $BASE_ALL_A_WEBNAME} |
{ $BASE_ALL_A_EROTIC_GAMES} - { $BASE_HEADER_LI_ACCOUNT}"class="light_button
back_button">{ $BASE_HEADER_LI_ACCOUNT}</a>

<div class="cufon_form_warning"><h4>Maj svoje objednávky <span class="h_bold">pod
kontrolou</span>.</h4></div>
<br/><br/>

{if $objednavky == 1}

<div id="my_order_list">
<table class="table">
  <tr class="header">
    <th><h3>{ $MYACCOUNT_ORDERS_TH_NUMBER}</h3></th>
    <th><h3>{ $MYACCOUNT_ORDERS_TH_TYPE}</h3></th>
    <th><h3>{ $MYACCOUNT_ORDERS_TH_PRICE}</h3></th>
    <th><h3>{ $MYACCOUNT_ORDERS_TH_STATUS}</h3></th>
    <th style="padding-left:30px !important;"><h3>{ $MYACCOUNT_ORDERS_TH_MANAGE}</h3></th>
    <th><h3>{ $MYACCOUNT_ORDERS_TH_PAYMENT}</h3></th>
  </tr>

  <div style="display:none;">{ $poc=0}</div>

  {foreach $orders as $orders}

  <div style="display:none;">{ $poc++}</div>

  <tr{if $poc % 2 == 1}class="nepar"{/if}>
    <td><span style="width:50px;">{ $orders->order_id}</span></td>
    <td><span style="width:130px;">{if $orders->deal_of_day ==
0}{ $MYACCOUNT_ORDERS_SPAN_STANDARD_TYPE}{else}{ $MYACCOUNT_ORDERS_SPAN_ACTION_TYPE}{/i
f}</span></td>
    <td><span style="width:90px;">{ $orders->price}{ $currency[ $orders-
>order_id]}</span></td>
    {if $orders->paid == 0 and $orders->deal_of_day == 0}<td
href="javascript:;id="status_{ $orders-
>order_id}"OnClick="show_hidden_element('order_status_unpaid');"title="{ $BASE_MYACCOU
NT_A_MYGAMES_A_UNPAID_ORDER}"class="order_status_unpaid
order_status_icon"></td>{elseif $orders->paid == 0 and $orders->deal_of_day == 1}<td
href="javascript:;id="status_{ $orders-
>order_id}"OnClick="show_hidden_element('order_status_processing');"title="{ $BASE_MYA
CCOUNT_A_MYGAMES_A_PROCESSING_ORDER}"class="order_status_paid_action
order_status_icon"></td>{else}<td href="javascript:;id="status_{ $orders-
>order_id}"OnClick="show_hidden_element('order_status_paid');"title="{ $BASE_MYACCOUNT
_A_MYGAMES_A_PAID_ORDER}"class="order_status_paid order_status_icon"></td>{/if}
    <td><span style="width:110px; padding-left:40px !important;">

    {** Ak objednavka nie je zaplatena, tak vykreslim moznosti *}
    {if $orders->paid == 0}

    <a href="{link "///this", 'id'=> 'detail', 'id_2' => $orders->order_id}"
```

```

title="{ $BASE_ALL_A_WEBNAME } | { $BASE_ALL_A_EROTIC_GAMES } -
{ $BASE_MYACCOUNT_A_MYGAMES_INFO_ORDERS } - { $BASE_MYACCOUNT_A_MYGAMES_A_ORDER_DETAIL}"
class="my_link">{ $MYACCOUNT_ORDERS_A_DETAIL}</a>
/ <a href="#order_cancel_verification_{ $orders-
>order_id}"title="{ $BASE_ALL_A_WEBNAME } | { $BASE_ALL_A_EROTIC_GAMES } -
{ $BASE_MYACCOUNT_A_MYGAMES_INFO_ORDERS } -
{ $BASE_MYACCOUNT_A_MYGAMES_A_ORDER_CANCEL}"class="my_link
popup_link">{ $MYACCOUNT_ORDERS_A_CANCEL}</a>

{else}

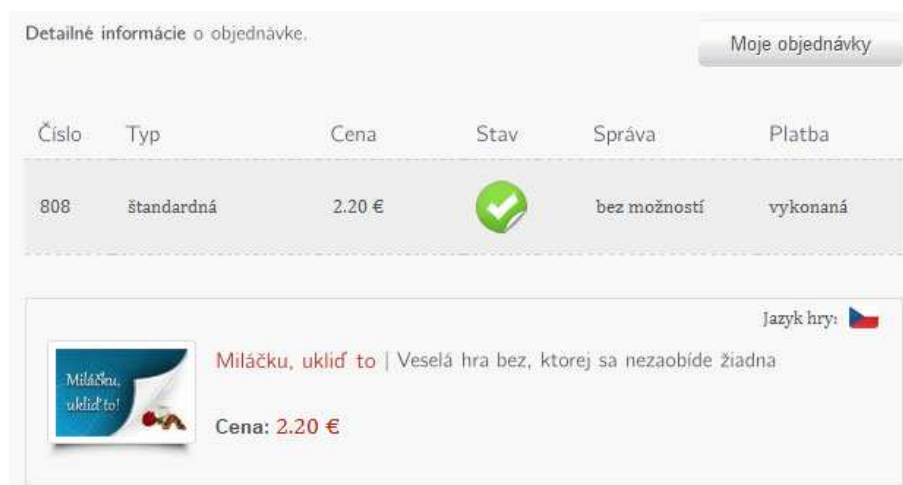
<a href="{link "///this", 'id'=> 'detail', 'id_2' => $orders->order_id}"
title="{ $BASE_ALL_A_WEBNAME } | { $BASE_ALL_A_EROTIC_GAMES } -
{ $BASE_MYACCOUNT_A_MYGAMES_INFO_ORDERS } - { $BASE_MYACCOUNT_A_MYGAMES_A_ORDER_DETAIL}"
class="my_link">{ $MYACCOUNT_ORDERS_A_DETAIL}</a>

{/if}
</span></td>
<td><span class="button_correction {if $orders->paid == 0 and $orders->deal_of_day ==
1}payment_verification_span{/if}"style="width:90px;">{if $orders->paid == 0 and
$orders->deal_of_day == 0}<a href="#order_purchase_{ $orders-
>order_id}"class="dark_button popup_link"title="{ $BASE_ALL_A_WEBNAME } |
{ $BASE_ALL_A_EROTIC_GAMES } - { $BASE_MYACCOUNT_A_MYGAMES_INFO_ORDERS } -
{ $BASE_MYACCOUNT_A_MYGAMES_A_ORDER_PURCHASE}">{ $MYACCOUNT_ORDERS_A_PAY}</a>{elseif
$orders->paid == 0 and $orders->deal_of_day ==
1}{ $MYACCOUNT_ORDERS_SPAN_VERIFICATION}{else}{ $MYACCOUNT_ORDERS_SPAN_DONE}{/if}</span
></td>
</tr>
</table>
{/foreach}

```

Tabuľka 11: Šablóna pre detail objednávky (zdrojový kód)⁵²

Výslednú grafickú podobu detailu objednávky je možné vidieť na obrázku 21.



Obrázok 21: Detail objednávky⁵³

⁵² Zdroj: vlastný

⁵³ Zdroj: vlastný

4.3.10 Zrušenie objednávky

Proces zrušenia objednávky prebieha prostredníctvom podania parametra „cancel“ a ID objednávky od URL adresy. Zdrojový kód, ktorý spracováva rušenie objednávky obsahuje tabuľka č. 12.

```
// V URL je požiadavka na zrušenie objednávky.
if($id_1=="cancel")
{
    // Kontrola, či je paramater, v ktorom je ID objednávky číslo.
    if(is_numeric($id_2))
    {
        // Zistenie, či objednávka existuje.

        $order=dibi::getConnection(DataBase::INVOICES)->query("SELECT * FROM orders where
user_id = %i and business_id = %i and ignored = 0 and paid = 0 and order_id =
%i;", $user_id, $id_podnikania, $id_2)->fetch();
        if($order!="")
        {
            // Zistenie aktuálneho času.

            $timestamp=time();

            // Vypísanie správy užívateľovi o zrušení objednávky.

            $this->flashMessage($this->mute->getText('MYGAMES_FLASHMESSAGE_SUCCESS_CANCEL'));

            // Zrušenie objednávky pomocou príkazu UPDATE.

            dibi::getConnection(DataBase::INVOICES)->query("Update orders set ignored = 1,
completed_timestamp = %i where user_id = %i and business_id = %i and ignored = 0
and order_id = %i;", $timestamp, $user_id, $id_podnikania, $id_2);

            // Uloženie záznamu o zrušení objednávky do príslušnej tabuľky.

            dibi::getConnection(DataBase::INVOICES)->query("Insert into ignored_orders_info
(order_id, user_id, change_timestamp, ignored)
values (%i, %i, %i, %i);",
$id_2, $user_id, time(), 1);
        }
        else
        {
            // Objednávku nebolo možné zrušiť.

            $this->flashMessage($this->mute->getText('MYGAMES_FLASHMESSAGE_UNSUCCESS_CANCEL'));
        }
    }
    else
    {
        // Objednávku nebolo možné zrušiť.

        $this->flashMessage($this->mute->getText('MYGAMES_FLASHMESSAGE_UNSUCCESS_CANCEL'));
    }

    // Presmerovanie užívateľa na stránku so zoznamom užívateľových objednávok.
    $this->redirect('MyAccount:Orders');
}
```

Tabuľka 12: Zrušenie objednávky (zdrojový kód)⁵⁴

⁵⁴ Zdroj: vlastný

Pred definitívnym zrušením objednávky systém užívateľovi zobrazí vyskakovacie okno s overením aktuálnej činnosti. Ukážka zdrojového kódu overovacieho vyskakovacieho okna pred zrušením objednávky je v tabuľke č. 13.

```
{foreach $orders_2 as $orders_2}
{** Zrusenie objednávky *}
<div style="display:none">
  <div id="order_cancel_verification_{$orders->order_id}"class="popup_window">
    <div id="order_cancel_verification">

      <div id="logo">
        <h1 class="logo popup_logo"><a href="{link //Homepage:}"title="{BASE_ALL_A_WEBNAME}
        | {BASE_ALL_A_EROTIC_GAMES}"></a></h1>
      </div>

      <h2>Overenie <span>zrušenie objednávky</span></h2>
      <p>Skutočne si praješ <strong>zrušiť svoju objednávku?</strong></p>
      <a href="{link "//this", 'id'=> 'cancel', 'id_2' => $orders->order_id}"
      title="{BASE_ALL_A_WEBNAME} | {BASE_ALL_A_EROTIC_GAMES} -
      {BASE_MYACCOUNT_A_MYGAMES_INFO_ORDERS} - {BASE_MYACCOUNT_A_MYGAMES_A_ORDER_CANCEL}"
      class="light_button">{MYACCOUNT_ORDERS_A_YES_CANCEL_BUTTON}</a>
      <a style="margin-left:10px;"href="javascript:;"onclick="$.fancybox.close();"title="{BASE_ALL_A_WEBNAM
      E} | {BASE_ALL_A_EROTIC_GAMES} - {BASE_MYACCOUNT_A_MYGAMES_INFO_ORDERS} -
      {BASE_MYACCOUNT_A_MYGAMES_A_ORDER_NO_CANCEL}"class="dark_button">{MYACCOUNT_ORDERS_
      A_NO_CANCEL_BUTTON}</a>
    </div>
  </div>
</div>
{foreach}
```

Tabuľka 13: Zrušenie objednávky (zdrojový kód)⁵⁵

Skutočná podoba vyskakovacieho okna s overením zrušenia objednávky je na obrázku 22.



Obrázok 22: Zrušenie objednávky⁵⁶

⁵⁵ Zdroj: vlastný

⁵⁶ Zdroj: vlastný

4.3.11 Zaplattenie objednávky (obecne)

Odkaz „Zaplatiť“ po svojom aktivovaní vyvolá vyskakovacie okno, ktoré ponúka momentálne podporované možnosti platby. Aktuálne sú podporované dve možnosti platby:

- platba pomocou služby PayPal,
- platba prevodom na účet.

Jednotlivé možnosti platieb sú detailne rozobrané v nasledujúcich podkapitolách. Ukážka zdrojového kódu šablóny, ktorý vykresľuje spomínané vyskakovacie okno je obsiahnutá v tabuľke č. 14.

```
{foreach $orders_2 as $orders_2}
{** Platba za objednávky *}
<div style="display:none;">
  <div id="order_purchase_{ $orders_2->order_id}"class="popup_window">
    <div id="order_purchase">
      <h2>Platba za <span>objednávku číslo { $orders_2->order_id}</span></h2>
      <span class="purchase_price">{ $orders_2->price}{ $currency[ $orders_2->order_id]}</span>

      <h3 class="border_bottom purchase_margin_bottom">Platba pomocou <span
      class="h_bold">PayPal</span></h3>
      <h4>Pri platbe cez <span class="h_bold">PayPal</span> sa hry stávajú aktívne <span
      class="h_bold">okamžite po platbe!</span><span class="color_red">Vzrušujúca zábava
      bez čakania!</span></h4>

      <div class="alignright thumb_shadow payment_picture">
        <a href="{link '//Paypal:Order','id'=> $orders_2->order_id}"
        title="{ $BASE_ALL_A_WEBNAME} | { $BASE_ALL_A_EROTIC_GAMES} -
        { $BASE_MYACCOUNT_A_MYGAMES_A_PAYPAL}"></a>
      </div>

      <a href="{link '//Paypal:Order','id'=> $orders_2->order_id}"
      title="{ $BASE_ALL_A_WEBNAME} | { $BASE_ALL_A_EROTIC_GAMES} -
      { $BASE_MYACCOUNT_A_MYGAMES_A_PAYPAL}" class="dark_button
      purchase_button">{ $BASE_MYACCOUNT_A_MYGAMES_A_PAYPAL}</a>

      <h3 class="border_bottom purchase_margin_top">Platba <span class="h_bold">prevodom
      na účet</span></h3>
      <h4>Pri platbe <span class="h_bold">prevodom na účet</span> môže pripísanie platby
      trvať <span class="h_bold">až 3 pracovné dni</span> pri tuzemskom prevode. Hry sa
      stávajú aktívne <span class="h_bold">až&nbsp;</span>po pripísaní platby.</span></h4>

      <table class="table">
        <tr style="border-top-color:#ddd; border-top-style: dashed; border-top-width:
        1px;"class="nepar"><th><h4>{ $MYACCOUNT_ORDERS_SPAN_ACCOUNT_NUMBER} SK:
        </h4></th><td><strong>XXXXXXXXXX / XXXX</strong></td></tr>
        <tr><th><h4>{ $MYACCOUNT_ORDERS_SPAN_ACCOUNT_NUMBER} CZ:
        </h4></th><td><strong>XXXXXXXXXX / XXXX</strong></td></tr>
        <tr class="nepar"><th><h4>IBAN: </h4></th><td>XXXXXXXXXXXXXXXXXXXX</td></tr>
        <tr><th><h4>SWIFT: </h4></th><td>XXXXXXXXXXXX</td></tr>
        <tr class="nepar"><th><h4>{ $MYACCOUNT_ORDERS_SPAN_BANK_ADDRESS}: </h4></th><td>Fio
        banka, a.s., V Celnici 1028/10, Praha 1</td></tr>
        <tr><th><h4>{ $MYACCOUNT_ORDERS_SPAN_PRICE_SUM}: </h4></th><td><strong>{ $orders_2-
```

Tabuľka 14: Informácií o platbe (zdrojový kód)⁵⁷

Platba za **objednávku číslo 808**

2.20 €

Platba pomocou **PayPal**

Pri platbe cez **PayPal** sa hry stávajú aktívne okamžite po platbe! **Vzrušujúca zábava bez čakania!**

Zaplatiť cez PayPal



Platba **prevodom na účet**

Pri platbe **prevodom na účet** môže trvať pripísanie platby až 3 pracovné dni pri tuzemskom prevode. Hry sa stávajú aktívne až po pripísaní platby.

Číslo účtu SK:	XXXXX
Číslo účtu CZ:	XXXXX
IBAN:	XXXXX
SWIFT:	XXXXX
Adresa banky:	XXXXX
Čiastka:	2.20 €
Variabilný symbol:	808

4.3.12 Platba pomocou služby PayPal

⁵⁷ Zdroj: vlastný

90

- PayPal.php,
- HRequest.php.

Autorom oboch knižníc je Martin Malý a ich využitie podlieha pravidlám popísaným v samotných knižniciach. Pri využívaní spomínaných knižníc došlo k minimálnym, prípadne žiadnym zmenám a nakoľko sa nejedná o vlastný kód ale hotové riešenie, diplomová práca sa v svojej hlavnej časti PayPal knižniciam už detailne nevenuje. V prípade záujmu sú však knižnice k dispozícii k nahliadnutiu na konci diplomovej práce v prílohe č. 3 a prílohe č. 4.

V realizovanom systéme využíva spomínané knižnice presenter Paypal, ktorého časti zdrojového kódu a ich popis sa nachádzajú v ďalšej časti kapitoly.

Tabuľka č. 15 zobrazuje volanie funkcie do ExpressCheckout z knižnice Paypal.php. Funkcia dostáva nasledovné parametre (sú v poradí):

- cena,
- popis produktu,
- poznámka o objednávke (je tam id objednávky a čas)
- mena,
- jazyk, ktorý udáva, na akú stránku Paypalu je potrebné presmerovať užívateľa.

```
$ret=($r->doExpressCheckout($obednavka_info-
>price,$meno,$id_1."_".time(),$obednavka_info->currency_abbreviation,$namespace-
>jazyk));
```

Tabuľka 15: Volanie funkcie doExpressCheckout (zdrojový kód)⁵⁹

Funkcia do ExpressCheckout podáva údaje ďalej do systému PayPal. Užívateľ je presmerovaný na webové stránky Paypalu, ktoré ilustruje obrázok 24.

⁵⁹ Zdroj: vlastný

Secret Games

Your order summary

Descriptions	
Objednávka číslo 809	€4,80
Item total	€4,80
Total €4,80 EUR	

Proceed to pay with PayPal

PayPal securely processes payments for Secret Games.
You can pay using your PayPal account or if you don't have one, you can pay as a PayPal guest.

Have a PayPal account?

Log in to your account to pay

Don't have a PayPal account?

Sign in as a PayPal guest and pay with your debit/credit card

Obrázok 24: PayPal⁶⁰

Následne má užívateľ dve možnosti:

- zrušiť platbu objednávky,
- zaplatiť objednávku.

V prípade nezaplatenia objednávky, je užívateľ presmerovaný na podstránku, ktorá informuje o nezaplatení objednávky (viď. obrázok 25).

Paypal

Platba bola zrušená



Platobný proces bol na žiadosť užívateľa
prerušný.

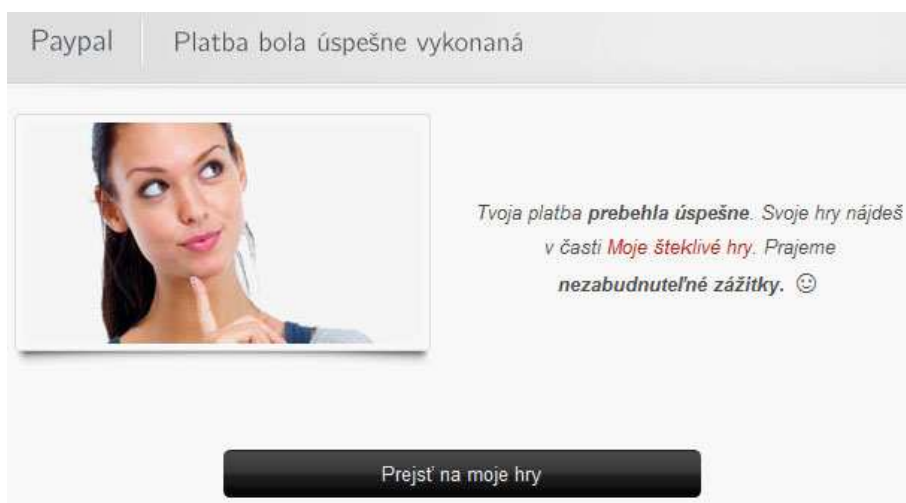
Späť na objednávku

Obrázok 25: PayPal - zrušená platba⁶¹

⁶⁰ *Proceed to pay with PayPal* [online] 1999 [cit. 2012-04-05]. Anglicky. Dostupný z URL <<https://www.paypal.com>>.

⁶¹ Zdroj: vlastný

V prípade zaplatenia objednávky, je užívateľ presmerovaný na podstránku, ktorá informuje o úspešnej platbe za objednávku (viď. obrázok 26).



Obrázok 26: Paypal - úspešná platba⁶²

System PayPal po úspešnej platbe vracia údaje o objednávke a výsledok platby za objednávku. Údaje o výsledku platby objednávky sú zistené pomocou kódu v tabuľke č. 16.

```
// Zistenie ako prebehla PayPal platba.  
$r=newPayPal(true);  
$final=$r->doPayment();  
if($final['ACK']=='Success'){  
    // *** Z bezpečnostných dôvodov nie je uvedený presný kód. ***  
    // Dochádza k zisteniu ID objednávky.  
}
```

Tabuľka 16: Údaje a PayPal platbe za objednávku⁶³

V ďalšom kroku systém aktivuje produkty podľa ID zaplatenej objednávky a odošle užívateľovi informačný e-mail o sprístupnení objednaných a zaplatených produktov. Popísanú činnosť vykonáva zdrojový kód v tabuľke č. 17.

⁶² Zdroj: vlastný

⁶³ Zdroj: vlastný

```

// Zistenie aktuálneho času.
$timestamp=time();

// Aktivácia objednávky.
dibi::getConnection(DataBase::INVOICES)->query("Update orders set paid = 1,
completed_timestamp = %i where business_id = %i and order_id =
%i;", $timestamp, $id_podnikania, $success_order_id);

// Uloženie informácií o objednávke do tabuľky zaplatených objednávok.
dibi::getConnection(DataBase::INVOICES)->query("Insert into paid_orders_info (order_id,
user_id, change_timestamp, ignored)
values (%i, %i, %i, %i);",
$success_order_id, $user_id, $timestamp, 0);

// Pridelenie užívateľského prístupu k produktom z aktivovanej objednávky.
dibi::getConnection(DataBase::INVOICES)->query("Update customers_products set available =
1, start_timestamp = %i where business_id = %i and user_id = %i and order_timestamp =
%i;", $timestamp, $id_podnikania, $order->user_id, $order->order_timestamp);

// Uloženie informácie o prebehnutí PayPal platby.

dibi::getConnection(DataBase::INVOICES)->query("Insert into paypal_payments (order_id,
user_id, completed_timestamp)
values (%i, %i, %i);",
$success_order_id, $user_id, $timestamp);

// Zistenie URL pre daný jazyk podľa ID jazyka z objednávky.

$skatka_jazyka=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT language_url FROM
languages where languages.language_id = %i;", $order->language_id)->fetch()-
>language_url;

// Naplnenie jazykových premenných, ktoré sa budú využívať v e-mailu. Premenné sú
naplnené podľa jazyka, ktorý mal užívateľ zvolený pri registrácii.

$template=$this->createTemplate();
$template->setFile(APP_DIR."/templates_mail/".$skatka_jazyka."_gameActivate.latte");

$template->slogan=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT full_text FROM
lang_".$skatka_jazyka." where constant = %s;", "BASE_FOOTER_MAIL-CLIENT_SLOGAN")-
>fetch()->full_text;
$template->pozdrav=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT full_text FROM
lang_".$skatka_jazyka." where constant = %s;", "BASE_FOOTER_MAIL-CLIENT_POZDRAV")-
>fetch()->full_text;
$template->pozdrav_projekt=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT
full_text FROM lang_".$skatka_jazyka." where constant = %s;", "BASE_FOOTER_MAIL-
CLIENT_POZDRAV-PROJEKT")->fetch()->full_text;
$template->automaticke_odoslanie=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT
full_text FROM lang_".$skatka_jazyka." where constant = %s;", "BASE_FOOTER_MAIL-
CLIENT_AUTOMATICKA-SPRAVA")->fetch()->full_text;
$template->meno=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT nickname FROM
users WHERE user_id = %i;", $order->user_id)->fetch()->nickname;
$template->order_id=$order->order_id;

// Odoslanie e-mailu.
$mail=newNMail;
$mail->setFrom(dibi::getConnection(DataBase::MAIN_DB)->query("SELECT mail FROM mail
WHERE mail_type = 1 and country_id = %i;", $order->registration_id_country)->fetch()-
>mail)// Odosielateľ správy.
->addTo($order->username)// Prijemca správy.
->setSubject(dibi::getConnection(DataBase::MAIN_DB)->query("SELECT full_text FROM
lang_".$skatka_jazyka." where constant = %s;", "ORDERS_ACTIVATE_MAIL_SUBJECT")->fetch()-
>full_text)// Predmet správy.
->setHtmlBody($template)// Telo e-mailu.
->send();// Odoslanie e-mailu.

```

Tabuľka 17: Aktivovanie produktov zo zaplatenej objednávky (zdrojový kód)⁶⁴

⁶⁴ Zdroj: vlastný

4.3.13 Platba pomocou prevodu na účet

V prípade, že sa užívateľ rozhodne zaplatiť za svoju objednávku štandardným prevodom na účet, systém nedokáže celú situáciu vyriešiť automaticky. Potrebný je zásah od administrátora, ktorý manuálne skontroluje stav zaplatenia objednávky pomocou výpisu z bankového účtu. Následne po potvrdení zaplatenia objednávky sa administrátor pomocou svojho administračného rozhrania dostane k vyhľadávaniu objednávok, ktoré má na starosti zdrojový kód v tabuľke č. 18. Podporované je vyhľadávanie podľa:

- čísla objednávky (variabilného symbolu),
- e-mailu užívateľa.

```
// Kontrola či je na vstupe vyhľadávania numerická hodnota. Ak áno, prebieha vyhľadávanie na základe variabilného symbol (čísla objednávky). Ak nie, prebieha vyhľadávanie na základe e-mailu.
```

```
if(is_numeric($id))
{
    // *** Vyhľadávanie na základe variabilného symbolu. ***

    // Zistenie počtu objednávok.

    $pocet_objednavok=dibi::getConnection(DataBase::INVOICES)->query("SELECT count(order_id) as pocet FROM orders where business_id = %i and order_id = %i;", $id_podnikania, $id)->fetch()->pocet;

    if($pocet_objednavok>0)
    {
        // Do templatu sa posúva informácia, že objednávky existujú.

        $this->template->objednavky=1;

        // Zistenie informácií o hľadanej objednávke.

        $orders=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT order_id, business_id, country_id, orders.user_id, order_timestamp, FROM_UNIXTIME(order_timestamp, '%d.%m.%Y %H:%i:%s') as cas_format, deal_of_day, action_code, action_id, orders.currency_id, price, paid, ignored, users.username, users.nickname, (select nickname from ".$mena_db->db_site_name.".deal_of_day, ".$mena_db->db_site_name.".deal_of_day_actions where ".$mena_db->db_site_name.".deal_of_day.deal_of_day_id = ".$mena_db->db_site_name.".deal_of_day_actions.deal_of_day_id and ".$mena_db->db_invoice_name.".orders.action_id = ".$mena_db->db_site_name.".deal_of_day_actions.action_id) as zlavomat
```

```

FROM ".$mena_db->db_invoice_name.".orders, ".$mena_db->db_site_name.".users where
business_id = %i and order_id = %i and users.user_id =
orders.user_id;", $id_podnikania, $id)->fetchassoc('order_id');

// Posunutie informácií o získanej objednávke do templatu.

$this->template->orders=$orders;

foreach($orders as $orders)
{
    // Zistenie meny objednávky.

    $currency_info=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT currency_symbol
FROM currency
where currency_id = %i;", $orders->currency_id)->fetch()->currency_symbol;

    $currency[$orders->order_id]=$currency_info;
}
$this->template->currency=$currency;
}
else
{
    // Do templatu sa posúva informácia, že objednávky neexistujú.

    $this->template->objednavky=0;
}
}
else
{
    // *** Vyhľadávanie na základe e-mailu. ***

    // Predloha validného e-mailu.

    $validmail="/^[a-zA-Z0-9_.-]+@[a-zA-Z0-9-]+.[a-zA-Z0-9-.-]+$/";

    // Kontrola zadaného e-mailu.

    if(preg_match($validmail, $id))
    {
        // Zistenie počtu objednávok.

        $pocet_objednavok=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT
count(order_id) as pocet FROM ".$mena_db->db_invoice_name.".orders, ".$mena_db->
db_site_name.".users where orders.business_id = %i and users.username = %s and
orders.user_id = users.user_id;", $id_podnikania, $id)->fetch()->pocet;

        if($pocet_objednavok>0)
        {
            // Do templatu sa posúva informácia, že objednávky existujú.

            $this->template->objednavky=1;

            // Zistenie všetkých produktov, ktoré má užívateľ objednané.

            $orders=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT order_id,
business_id,
country_id,
orders.user_id,
order_timestamp,
FROM_UNIXTIME(order_timestamp, '%d.%m.%Y %H:%i:%s') as cas_format,
deal_of_day,
action_code,
action_id,
orders.currency_id,
price,
paid,
ignored,

```



```

users.username,
users.nickname,
(select nickname from ".$mena_db->db_site_name.".deal_of_day, ".$mena_db->
db_site_name.".deal_of_day_actions where ".$mena_db->
db_site_name.".deal_of_day.deal_of_day_id = ".$mena_db->
db_site_name.".deal_of_day_actions.deal_of_day_id and ".$mena_db->
db_invoice_name.".orders.action_id = ".$mena_db->
db_site_name.".deal_of_day_actions.action_id) as zlavomat
FROM ".$mena_db->db_invoice_name.".orders, ".$mena_db->db_site_name.".users where
business_id = %i and users.username = %s and users.user_id =
orders.user_id;", $id_podnikania, $id)->fetchassoc('order_id');

// Posunutie poľa do templatu.
$this->template->orders=$orders;

foreach($ordersas$orders)
{
    // Zistenie meny objednávky.

    $currency_info=dibi::getConnection(DataBase::MAIN_DB)->query("SELECT currency_symbol
FROM currency
where currency_id = %i;", $orders->currency_id)->fetch()->currency_symbol;

    $currency[$orders->order_id]=$currency_info;
}
$this->template->currency=$currency;
}
else
{
    // Do templatu sa posúva informácia, že objednávky neexistujú.

    $this->template->objednavky=0;
}
}
else
{
    // Do templatu sa posúva informácia, že objednávky neexistujú.

    $this->template->objednavky=0;
}
}
}

```

Tabuľka 18: Vyhľadávanie objednávok (zdrojový kód)⁶⁵

Jednoduchá výsledná podoba formulára pre vstup do vyhľadávania objednávok je ilustrovaná obrázkom č. 27.

⁶⁵ Zdroj: vlastný

Vyhľadavanie

Vyhľadavanie a správa objednávok

Zadajte buď e-mailovú adresu klienta, alebo variabilný symbol objednávky.

Vyhľadávaný reťazec: Hľadať

Obrázok 27: Formulár vyhľadávania⁶⁶

Po úspešnom nájdení objednávky a potvrdení jej zaplatenia môže administrátor produkty obsiahnuté v objednávke aktivovať pomocou príslušného tlačidla. Výsledok vyhľadávania objednávok zobrazuje obrázok č. 28.

Číslo	Typ	E-mail	Prezývka	Cena	Čas	Stav	Správa
2	štandardná	kosturak@secret-games.com	Ivo	17.60 €	09.12.2011 02:14:15		Detail / Ignorovať / Aktivovať
28	štandardná	kosturak@secret-games.com	Ivo	2.20 €	15.12.2011 10:50:07		Detail
255	štandardná	kosturak@secret-games.com	Ivo	13.20 €	10.01.2012 14:32:01		Detail / Ignorovať / Aktivovať

Obrázok 28: Výsledok vyhľadávania objednávok⁶⁷

Aktivovaním, prípadne zrušením (ignorovaním) objednávky je proces spracovávanía objednávky ukončený.

4.4 Zavedenie realizovaného IS

Ďalším krokom, ktorý nasleduje po úspešnom vývoji a otestovaní IS je jeho zavedenie. Pred samotným zavedením IS musia byť pripravené nasledujúce body:

- Server ju umiestnení v server housing centre.
- Na fyzickom serveri beží nakonfigurovaný:
 - o webový server,
 - o databázový server,
 - o mailový server,
 - o ftp server.

⁶⁶ Zdroj: vlastný

⁶⁷ Zdroj: vlastný

O spomínané body sa stará technický pracovník.

Za stratégiu zavádzania IS je vzhľadom na neexistenciu predošlého informačného systému zvolená nárazová stratégia.

Súčasťou procesu zavádzanie IS v prípade IJM Group nie je školenie rozoberajúce spôsoby užívania systému. Dôvodom je, že minimálne v prvých mesiacoch po zavedení systému budú jeho jediní užívatelia samotní tvorcovia, ktorí z pochopiteľných dôvodov školenia nepotrebujú. Do budúcnosť sa príprava školiacich dokumentov plánuje ako súčasť rozširovania podnikateľskej činnosti spoločnosti.

Zavedenie webového IS prebieha veľmi jednoducho prostredníctvom ftp klienta, ktorým sa nakopírujú súbory systému z testovacieho servera a ostrý webový server. Následne prebehne posledné kontrolné testovanie, ktorým sa overí správna funkčnosť systému.

Detailná charakteristika hardwarového riešenia využitého ako platforma pre rozoberaný informačný systém je nad rámec diplomovej práce. Pre informáciu je v tabuľke 19 uvedená konfigurácia využitého servera, ktorý spoločnosti slúži už od jej vzniku.

Sun Fire X4275 Server:	
Operačný systém	Linux CentOS 5.x
CPU - procesor	Intel Xeon X5560 (2,8 GHz)
RAM - operačná pamäť	4 GB DDR3-1333 MHz Memory
HDD - pevné disky	450 GB 15000 rpm SAS Disk Drive
Bezpečnosť	RAID 1, Pripojenie na UPS

Tabuľka 19: Konfigurácia servera⁶⁸

Klientské stanice využívané v spoločnosti IJM Group sú už zakúpené dlhší čas a ich nákup priamo nesúvisel so zavádzaním nového IS. Realizovaný IS je možné využívať z ľubovoľnej klientskej stanice s pripojením na internet.

⁶⁸ Zdroj: vlastný

4.5 Prevádzka realizovaného IS

Ďalšom dôležitou súčasťou cyklu IS je po jeho zavedení samotná prevádzka systému. V prípade IJM Group je hlavným znakom prevádzky IS jej jednoduchosť. Webový IS je prístupný z akejkoľvek klientskej stanice s pripojením na internet. Klientské stanice používané majiteľmi spoločnosti nie sú systematicky udržiavané v stave schopnom prevádzky. Každý spolumajiteľ si za svoju klientskú stanicu zodpovedá sám.

Údržbu servera má na starosti technický pracovník pomocou vzdialeného prístupu. Údržba servera spočíva najmä v reakcii na e-mailové upozorňovacie správy, ktoré server automaticky odosiela v prípade problémov a občasnom ale pravidelnom sledovaní logov servera.

Základom nízkej náročnosti údržby systému je správna konfigurácia podporných služieb.

4.6 Zhodnotenie realizovaného návrhu

Zhodnotenie projektu štandardne pozostáva z porovnania prínosov projektu s nákladmi na jeho realizáciu.

4.6.1 Náklady na zavedenie IS

Náklady na realizáciu IS je možné v prípade IJM Group orientačným spôsobom odhadnúť prostredníctvom času, ktorý bol potrebný na vytvorenie systému a primeranej hodinovej mzdy ohodnocujúcej odbornú činnosť potrebnú pri realizácii. Odhad nákladov obsahuje tabuľka 20. Nejedná sa o reálne náklady (vyplatené výdaje) spoločnosti, ale o interný odhad, ktorý poskytuje lepšiu predstavu o systéme.

Odhad nákladov realizácie IS		
Počet odpracovaných hodín	Odhadnutá hodinová mzda	Výsledné náklady na IS
2 000	6 Eur	12 000 Eur

Tabuľka 20: Odhad nákladov realizácie IS⁶⁹

⁶⁹ Zdroj: vlastný

4.6.2 Prínosy zavedenia IS

Jednoducho merateľný prínos realizovaného návrh je možné vidieť v znížení časovej náročnosti vybavenia objednávky, čo rovnako predstavuje zníženie nákladov. Spomínanú úsporu zobrazuje tabuľka 21.

Úspora nákladov po zavedení IS					
Priemerná časová náročnosť vybavenia jednej objednávky pred zavedením IS	Priemerná časová náročnosť vybavenia jednej objednávky po zavedení IS	Priemerný počet objednávok za mesiac	Ušetrený čas za mesiac (0,08 hod*600)	Hodinová mzda pracovníka	Ušetrené náklady za mesiac
0,1 hod	0,02 hod	600	48 hod	4 Eurá	192 Eur

Tabuľka 21: Úspora nákladov po zavedení IS⁷⁰

Spoločnosť IJM Group sa zaoberá predajom produktov, ktoré sú výhradne v elektronickej podobe a pred zavedením informačného systému vybavenie objednávky zahŕňalo odosielanie produktov v prílohách na e-mailové adresy zákazníkov. Proces neautomatizovaného odosielanie veľkého množstva e-mailov bol zbytočne náročný na čas aj sústredenie a navyše si vyžadoval neustálu prítomnosť aktuálnych verzií ponúkaných produktov na práve využívanom lokálnom počítači. Ďalšou nevýhodou bola nespoľahlivosť a obmedzenia e-mailovej komunikácie. Prílohy sa často stávali pre zákazníka nečitateľné a v niektorých prípadoch musela byť veľkosť ponúkaných produktov redukovaná na úkor ich kvality kvôli kapacitným obmedzeniam príloh e-mailov. Nedostatok predstavoval aj proces sledovanie objednávok, ktorý bol neprehľadný. Zavedením popisovaného IS sa stal proces odoslania produktu zákazníkovi otázkou jedného kliknutia. Využitie optimalizovanej databázy výrazným spôsobom zvýšilo prehľadnosť vybavovania objednávok. Odstránená bola aj potreba redukovania veľkosti produktov, čím sa zvýšila ich kvalita a všetky produkty majú teraz navyše svoje aktuálne verzie uložené centrálné na serveri.

Napriek tomu, že realizovaný systém bol pôvodne navrhovaný hlavne pre interné potreby konkrétnej spoločnosti dosahuje podoby, ktorá umožňuje jeho predaj externým

⁷⁰ Zdroj: vlastný

subjektom s podobným zameraním ako má spoločnosť IJM Group. Uplatnenie systému pre ľubovoľné internetové obchody si vyžaduje výraznejší zásah do systému, avšak možnosť prispôsobenia systému obecnému rozsahu požiadaviek nie je vylúčená. Potenciálne výnosy z predaja IS externým subjektom môžu byť chápané ako ďalší prínos systému a s využitím odhadu tržnej ceny a počtu predaných kusov ich ukazuje tabuľka 22.

Potenciálne výnosy z predaja systému		
Finálna cena systému	Počet predpokladaných predaných kusov ročne	Ročný výnos
1800 Eur	7	12 600 Eur

Tabuľka 22: Potenciálne výnosy z predaja systému⁷¹

Počet predpokladaných kusov systému ročne nepredstavuje skutočný odhad, ale demonštráciu množstva predaných systémom, ktoré je potrebné na navrátenie nákladov spojených s vytvorením IS. Pre relevantný odhad počtu predaných kusov je potrebná samostatná analýza trhu informačných systémom, ktorá je však vzhľadom na zameranie diplomovej práce a sekundárnu dôležitosť potenciálnych výnosov z predaja systému nad jej rámec.

Pri určení ekonomického prínosu zavedenia informačných systémov je podstatné uvedomenie, že sa jedná o zložitý problém. Hlavný prínos v prípade IJM Group spočíva v zabezpečení plánovanej podnikateľskej činnosti a nie je ho možné v ekonomických ukazovateľoch uspokojivo vyjadriť. Bez zavedenia rozoberaného IS by spoločnosť nebola schopná realizovať svoje podnikateľské zámery.

4.6.3 Výsledok ekonomického porovnania prínosov a nákladov zavedenia IS

Z predchádzajúcich kapitol vyplýva, že v priebehu času sa náklady spojené s vytváraním a zavádzaním IS do spoločnosti IJM Group bezpečne vrátia. V prípade úspešného predaja systému externým subjektom môže návratnosť investície nadobúdať hodnotu len niekoľkých mesiacov.

⁷¹ Zdroj: vlastný

Záverom zhodnotenia nemožno zabudnúť na zásadný faktor, ktorý čiastočne znižuje význam štandardnej ekonomickej analýzy v špecifickom prípade spoločnosti IJM Group. Jedná sa o neexistenciu iných reálne možných alternatív k navrhovanému riešeniu. Spoločnosť by bez IS nebola schopná konkurencie a ani základnej prevádzky. Vzhľadom na neštandardné požiadavky a dostačujúce programátorské schopnosti bola jednoznačne zvolaná možnosť vytvorenia vlastného IS. Spoločnosť sa rozhodla, že v porovnaní so kúpením zákazkové IS je výhodnejšie zvoliť vytvorenie vlastného IS a využiť vlastný čas a energiu, ktorými výkonná zložka spoločnosti na rozdiel od voľných finančných prostriedkov disponuje.

Záver

Cieľom diplomovej práce bolo rozobratie rôznych techník využívaných pri návrhu, realizácii a implementácii IS. Práve zo spomenutého dôvodu sa diplomová práca vo svojej úvodnej časti venuje teoretickým východiskám problematiky informačných systémov, metodológií ich tvorby a technológiám používaným pri vytváraní webových informačných systémov.

Ďalším cieľom diplomovej práce bolo zvoliť riešenie realizácie informačného systému, ktoré by najviac vyhovovalo spoločnosti IJM Group a zabezpečiť maximálnu možnú zhodu predstáv o výslednom systéme a jeho finálnej podobe. Zabezpečenie spomínanej zhody bolo úlohou analytickej časti diplomovej práce. Analytická časť práce sa sústreďuje na problematiku spoločnosti IJM Group s dôrazom na využitie populárnej SWOT analýzy a špecifikáciu požiadaviek na výsledný informačný systém za účelom prekrytia systému a podnikových procesov spoločnosti. Záver analytickej časti sa venuje výberu vhodného frameworku ako nástroja pre nezanedbateľné uľahčenie a zefektívnenie programátorských prác na projekte.

Najdôležitejšou časťou diplomovej práce je návrhová časť, ktorá vo svojom úvode rámcovým spôsobom predstavuje celý realizovaný IS a jeho vlastnosti. Práca vo svojej ďalšej časti detailne rozoberá systémovú podporu jedného z hlavných podnikových procesov spoločnosti, ktorým je spracovanie objednávky. Popis spracovania objednávky obsahuje príslušné časti zdrojového kódu, ktoré sú doplnené o vysvetľujúce komentáre, ako aj ilustračného obrázku výslednej podoby systému.

Konečným výstupom diplomovej práce je funkčný informačný systém spĺňajúci definované požiadavky spoločnosti, ktorý podporuje hlavné podnikové procesy a umožňuje spoločnosti úspešne vykonávať podnikateľskú činnosť obsiahnutú v jej strategických plánoch.

Záverom je možné konštatovať, že diplomová práca naplnila všetky svoje ciele.

Zoznam použitej literatúry

Nasledujúca kapitola obsahuje sumár zdrojov použitých v diplomovej práci.

Použité monografie

BANÝ, P a kol. *Základní metody a nástroje systémové analýzy a projektování informačních systémů*. In: Sborník Mendelovy zemědělské a lesnické univerzity v Brně, 1999, č. 2, s. 209-216.

BRUCKNER, T a kol. *Tvorba informačních systémů: principy, metodiky, architektury*. 1. vyd. Praha: Grada Publishing, 2012. 357 s. ISBN 978-80-247-4153-6.

DARIE, C. a kol. *AJAX a PHP tvoříme interaktivní webové aplikace PROFESIONÁLNĚ*. Brno: Zoner Press, 2006. 320 s. ISBN: 80-86815-47-1.

GILMORE, W. J. *Velká kniha PHP a MySQL 5: kompendium znalostí pro začátečníky i profesionály*. 1. vyd. Brno: Zoner press, 2005. 712 s. ISBN 80-86815-20-X.

GUTMANS, A., BAKKEN, S.S., RETHANS, D. *mistrovství v PHP 5*. 2. vyd. Brno: Computer Press, 2007. 655 s. ISBN: 978-80-251-1519-0.

HORNÝ, S. *Analýza a návrh systémů*. 1. vyd. Praha: Vysoká škola ekonomická, 1999, 98 s. ISBN: 978-80-245-0007-2.

KOCH, M. *Dátové a funkční modelování*. Brno: Akademické nakladatelství Cerm, 2006. 108 s. ISBN: 80-214-2724-8.

KOFLER, M a ÖGGL, B. *PHP 5 a MySQL 5: průvodce webového programátora*. 1. vyd. Brno: Computer Press, 2007. 608 s. ISBN 978-80-251-1813-9.

MOLNÁR, Z. *Moderní metody řízení informačních systémů*. Praha: Grada Publishing, 1992. 347 s. ISBN 978-80-856-2307-9.

TVRDÍKOVÁ, M. *Zavádění a inovace informačních systémů ve firmách*. 1. vyd. Praha: Grada Publishing, 2000. 116 s. ISBN 80-7169-703-6.

ULLMAN, L. *PHP pokročilé programování pro World Wide Web*. 1. vyd. Praha: SoftPress, 2002. 512 s. ISBN 80-8649-73-64.

VYMĚTAL, D. *Informační systémy v podnicích*. 1. vyd. Praha: Grada Publishing, 2009. 144 s. ISBN 978-80-247-3046-2.

VLASÁK, R. - BULÍČKOVÁ, S. *Základy projektování informačních systémů*. 1. vyd. Praha: Karolinum, 2003. 144 s. ISBN 80-246-0727-1.

Použité elektronické zdroje

Dokonalé zabezpečení webových aplikací | Nette Framework [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL< <http://nette.org/cs/hlavni-prednosti/bezpecnost>>.

Implementing PayPal Express Checkout [online] 2008 [cit. 2012-04-05]. Anglicky. Dostupný z URL <<http://www.php-suit.com/paypal>>.

Obchodný register na internete [online]. 2011 [cit. 2012-01-28]. Slovensky. Dostupný z URL <<http://orsr.sk/vypis.asp?ID=200687&SID=3&P=0>>.

PHP frameworky – Programujte.com [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL<<http://programujte.com/clanek/2008022000-php-frameworky>>.

Proceed to pay with PayPal [online] 1999 [cit. 2012-04-05]. Anglicky. Dostupný z URL <<https://www.paypal.com>>.

Quick Start | dibi [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL<<http://dibiphp.com/cs/quick-start>>.

Velký test PHP frameworků – Root.cz [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL<<http://www.root.cz/clanky/velky-test-php-frameworku-2008>>.

Vytvoření Presenteru | Nette Framework [online] 2008 [cit. 2012-03-11]. Česky. Dostupný z URL<<http://doc09.nette.org/cs/quickstart/vytvoreni-presenteru>>.

Prílohy

- Príloha č. 1: Výpis z obchodného registra
- Príloha č. 2: Knižnica Paypal.php
- Príloha č. 3: Knižnica HRequest.php
- Príloha č. 4: Ilustračné obrázky administrátorského rozhrania realizovaného IS
- Príloha č. 5: Ilustračné obrázky užívateľského rozhrania Secret Games
- Príloha č. 6: Ilustračné obrázky užívateľského rozhrania Forex Accounts

Príloha č. 1: Výpis z obchodného registra

Zoznam výpisov č.:

**VÝPIS
Z OBCHODNÉHO REGISTRA
Okresného súdu Banská Bystrica**

Oddiel: Sro
Vložka číslo: 19726/S

I. OBCHODNÉ MENO

IJM Group, s.r.o.

II. SÍDLO

Názov ulice (iného verejného priestranstva) a orientačné číslo (príp. súpisné číslo):
Sokolská 2518/48
Názov obce: Zvolen
PSČ: 960 01

III. IČO: 46 050 868

IV. DEŇ ZÁPISU: 24.02.2011

V. PRÁVNÁ FORMA: Spoločnosť s ručením obmedzeným

VI. PREDMET PODNIKANIA (ČINNOSTI)

1. kúpa tovaru na účely jeho predaja konečnému spotrebiteľovi (maloobchod) alebo iným prevádzkovateľom živnosti (veľkoobchod)
2. sprostredkovateľská činnosť v oblasti obchodu
3. sprostredkovateľská činnosť v oblasti služieb
4. činnosť podnikateľských, organizačných a ekonomických poradcov
5. reklamné a marketingové služby
6. prieskum trhu a verejnej mienky
7. počítačové služby
8. služby súvisiace s počítačovým spracovaním údajov
9. vykonávanie mimoškolskej vzdelávacej činnosti

VII. ŠTATUTÁRNY ORGÁN: KONATELIA

Meno a priezvisko: Martin Volko

Bydlisko:

Názov ulice (iného verejného priestranstva) a orientačné číslo (príp. súpisné číslo):
Sokolská 2518/48

Názov obce: Zvolen

PSČ: 960 01

Dátum narodenia: 28.03.1988

Rodné číslo: 880328/8582

Vznik funkcie: 24.02.2011

Meno a priezvisko: Ivan Kosturák

Bydlisko:

Názov ulice (iného verejného priestranstva) a orientačné číslo (príp. súpisné číslo):

J. Jesenského 2128/73
Názov obce: Zvolen
PSČ: 960 01
Dátum narodenia: 05.07.1988
Rodné číslo: 880705/8601

Vznik funkcie: 24.02.2011

Spôsob konania štatutárneho orgánu v mene spoločnosti s ručením obmedzeným:
Konatelia konajú a podpisujú v mene spoločnosti vždy spoločne.

VIII. SPOLOČNÍCI

Meno a priezvisko: Martin Volko
Bydlisko:
Názov ulice (iného verejného priestranstva) a orientačné číslo (príp. súpisné číslo):
Sokolská 2518/48
Názov obce: Zvolen
PSČ: 960 01

Výška vkladu: 1 700,000000 EUR
Rozsah splatenia: 850,000000 EUR

Meno a priezvisko: Ivan Kosturák
Bydlisko:
Názov ulice (iného verejného priestranstva) a orientačné číslo (príp. súpisné číslo):
J. Jesenského 2128/73
Názov obce: Zvolen
PSČ: 960 01

Výška vkladu: 1 700,000000 EUR
Rozsah splatenia: 850,000000 EUR

Meno a priezvisko: Juraj Kováčik
Bydlisko:
Názov ulice (iného verejného priestranstva) a orientačné číslo (príp. súpisné číslo):
Horná Ves 43/D
Názov obce: Kremnica
PSČ: 967 01

Výška vkladu: 1 700,000000 EUR
Rozsah splatenia: 850,000000 EUR

IX. VÝŠKA ZÁKLADNÉHO IMANIA

5 100,000000 EUR

X. ROZSAH SPLATENIA ZÁKLADNÉHO IMANIA

2 550,000000 EUR

Banská Bystrica, 24.02.2011

Správnosť výpisu sa potvrdzuje

Za správnosť výpisu: Jana Cavarová


.....
(podpis oprávnenej osoby)


.....
(odtlačok úradnej pečiatky)

Strana 3

```

<?php
/**
 * Class PayPal
 *
 * @version 1.0
 * @author Martin Maly - http://www.php-suit.com
 * @copyright (C) 2008 martin maly
 * @see http://www.php-suit.com/paypal
 * 2.10.2008 20:30:40
 */

/*
 * Copyright (c) 2008 Martin Maly - http://www.php-suit.com
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are met:
 *
 * * Redistributions of source code must retain the above copyright
 *   notice, this list of conditions and the following disclaimer.
 *
 * * Redistributions in binary form must reproduce the above copyright
 *   notice, this list of conditions and the following disclaimer in the
 *   documentation and/or other materials provided with the distribution.
 *
 * * Neither the name of the <organization> nor the
 *   names of its contributors may be used to endorse or promote products
 *   derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY MARTIN MALY 'AS IS' AND ANY
 * EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
 * WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL MARTIN MALY BE LIABLE FOR ANY
 * DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
 * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
 * SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

class PayPal extends BasePresenter
{
    //these constants you have to obtain from PayPal
    //Step-by-step manual is here: https://cms.paypal.com/us/cgi-bin/?cmd=_render-
    //content&content_ID=developer/e_howto_api_NVPAPIBasics

    const API_USERNAME="UZIVATEL";
    const API_PASSWORD="HESLO";
    const API_SIGNATURE="PODPIS";

    // Z bezpečnostných dôvodov nie sú uvedené reálne prístupové údaje k
    // obchodníckemu PayPal účtu.

    private $endpoint;
    private $host;
    private $gate;

    function __construct($real=false){
        $this->endpoint='/nvp';
        if($real){
            $this->host="api-3t.paypal.com";
            $this->gate='https://www.paypal.com/cgi-bin/webscr?';
        }else{
            //sandbox
            $this->host="api-3t.sandbox.paypal.com";
            $this->gate='https://www.sandbox.paypal.com/cgi-bin/webscr?';
        }
    }

    /**
     * @return HRequest
     */
    private function response($data){

```

```

        $r=newHttpRequest($this->host,$this->endpoint,'POST',true);
        $result=$r->connect($data);
        if($result<400)return$r;
        returnfalse;
    }

    privatefunctionbuildQuery($data=array()){
        $data['USER']=self::API_USERNAME;
        $data['PWD']=self::API_PASSWORD;
        $data['SIGNATURE']=self::API_SIGNATURE;
        $data['VERSION']='52.0';
        $query=http_build_query($data);
        return$query;
    }

    /**
     * Main payment function
     *
     * If OK, the customer is redirected to PayPal gateway
     * If error, the error info is returned
     *
     * @param float $amount Amount (2 numbers after decimal point)
     * @param string $desc Item description
     * @param string $invoice Invoice number (can be omitted)
     * @param string $currency 3-letter currency code (USD, GBP, CZK etc.)
     *
     * @return array error info
     */
    publicfunctiondoExpressCheckout($amount,$desc,$invoice='', $currency='USD', $jazyk)
    {
        $data=array(
            'PAYMENTACTION'=>'Sale',
            'AMT'=>$amount,

            // Definovanie stránky kde sa užívateľ vráti v prípade úspešného
            // zaplatenia objednávky.

            'RETURNURL'=>"http://secret-games.com/".$jazyk."/paypal/success",

            // Definovanie stránky kde sa užívateľ vráti v prípade neúspešného
            // zaplatenia objednávky.

            'CANCELURL'=>"http://secret-games.com/".$jazyk."/paypal/cancel",
            'DESC'=>$desc,
            'NOSHIPPING'=>"1",
            'ALLOWNOTE'=>"1",
            'CURRENCYCODE'=>$currency,
            'METHOD'=>'SetExpressCheckout');

        $data['CUSTOM']=$amount.'|'.$currency.'|'.$invoice;
        if($invoice)$data['INVNUM']=$invoice;

        $query=$this->buildQuery($data);

        $result=$this->response($query);

        if(!$result)returnfalse;
        $response=$result->getContent();
        $return=$this->responseParse($response);

        if($return['ACK']=='Success'){
            header('Location:'. $this->gate.'cmd=_express-
            checkout&useraction=commit&token='.$return['TOKEN'].'');
            die();
        }
        return($return);
    }

    publicfunctiongetCheckoutDetails($token){
        $data=array(
            'TOKEN'=>$token,
            'METHOD'=>'GetExpressCheckoutDetails');
        $query=$this->buildQuery($data);
    }

```



```

        $result=$this->response($query);

        if(!$result)returnfalse;
        $response=$result->getContent();
        $return=$this->responseParse($response);
        return($return);
    }
    publicfunctiondoPayment(){
        $token=$_GET['token'];
        $payer=$_GET['PayerID'];
        $details=$this->getCheckoutDetails($token);
        if(!$details)returnfalse;
        list($amount,$currency,$invoice)=explode('|',$details['CUSTOM']);
        $data=array(
            'PAYMENTACTION'=>'Sale',
            'PAYERID'=>$payer,
            'TOKEN'=>$token,
            'AMT'=>$amount,
            'CURRENCYCODE'=>$currency,
            'METHOD'=>'DoExpressCheckoutPayment');
        $query=$this->buildQuery($data);
        $result=$this->response($query);

        if(!$result)returnfalse;
        $response=$result->getContent();
        $return=$this->responseParse($response);

        /*
        * [AMT] => 10.00
        * [CURRENCYCODE] => USD
        * [PAYMENTSTATUS] => Completed
        * [PENDINGREASON] => None
        * [REASONCODE] => None
        */

        return($return);
    }
    privatefunctiongetScheme(){
        $scheme='http';
        if(isset($_SERVER['HTTPS'])and$_SERVER['HTTPS']=='on'){
            $scheme.='s';
        }
        return$scheme;
    }
    privatefunctionresponseParse($resp){
        $a=explode("&",$resp);
        $out=array();
        foreach($aas$v){
            $k=strpos($v,'=');
            if($k){
                $key=trim(substr($v,0,$k));
                $value=trim(substr($v,$k+1));
                if(!$key)continue;
                $out[$key]=urlencode($value);
            }else{
                $out[]=$v;
            }
        }
        return$out;
    }
}

```

Tabuľka: Knižnica Paypal.php (zdrojový kód)¹

¹Implementing PayPal Express Checkout [online] 2008 [cit. 2012-04-05]. Anglicky. Dostupný z URL <<http://www.php-suit.com/paypal>>.

Príloha č. 3: Knížnica HRequest.php

```
<?php
/**
 * HTTP Request class
 *
 * @version 1.0
 * @author martin maly
 * @copyright (C) 2008 martin maly
 */

/**
 * Class HTTPRequest
 *
 * @version 1.0
 * @author martin maly
 * @copyright (C) 2008 martin maly
 * 2.10.2008 20:10:40
 */
// HTTPRequest

class HRequest extends BasePresenter
{

    private $host;
    private $path;
    private $data;
    private $method;
    private $port;
    private $rawhost;

    private $header;
    private $content;
    private $parsedHeader;

    function __construct($host, $path, $method='POST', $ssl=false, $port=0){
        $this->host=$host;
        $this->rawhost=$ssl?("ssl://" . $host):$host;
        $this->path=$path;
        $this->method=strtoupper($method);
        if($port){
            $this->port=$port;
        }else{
            if(!$ssl)$this->port=80;else $this->port=443;
        }
    }

    public function connect($data='') {
        $fp=fsockopen($this->rawhost, $this->port);
        if(!$fp) return false;
        fputs($fp, "$this->method$this->path HTTP/1.1\r\n");
        fputs($fp, "Host: $this->host\r\n");
        //fputs($fp, "Content-type: $contenttype\r\n");
        fputs($fp, "Content-length: ".strlen($data)."\r\n");
        fputs($fp, "Connection: close\r\n");
        fputs($fp, "\r\n");
        fputs($fp, $data);

        $responseHeader='';
        $responseContent='';

        do
        {
            $responseHeader.=fread($fp, 1);
        }
        while(!preg_match('/\r\n\r\n$/',$responseHeader));

        if(!strstr($responseHeader, "Transfer-Encoding: chunked"))
        {

```

```

        while(!feof($fp))
        {
            $responseContent.=fgets($fp,128);
        }
    }
    else
    {
        while($chunk_length=hexdec(fgets($fp)))
        {
            $responseContentChunk='';

            $read_length=0;

            while($read_length<$chunk_length)
            {
                $responseContentChunk.=fread($fp,$chunk_length-$read_length);
                $read_length=strlen($responseContentChunk);
            }

            $responseContent.=$responseContentChunk;

            fgets($fp);

        }

    }

    $this->header=chop($responseHeader);
    $this->content=$responseContent;
    $this->parsedHeader=$this->headerParse();

    $code=intval(trim(substr($this->parsedHeader[0],9)));

    return $code;
}

function headerParse(){
    $h=$this->header;
    $a=explode("\r\n",$h);
    $out=array();
    foreach($a as $v){
        $k=strpos($v,':');
        if($k){
            $key=trim(substr($v,0,$k));
            $value=trim(substr($v,$k+1));
            if(!$key) continue;
            $out[$key]=$value;
        }
        else
        {
            if($v) $out[]=$v;
        }
    }
    return $out;
}

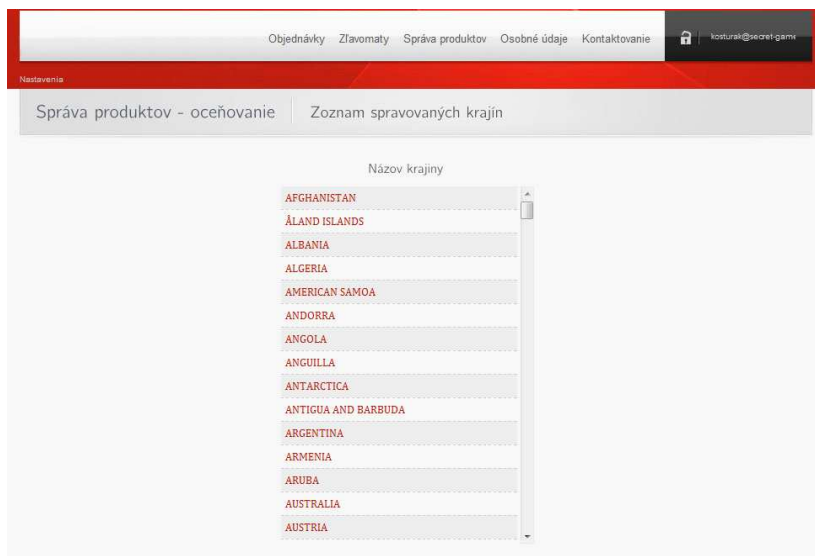
public function getContent(){return $this->content;}
public function getHeader(){return $this->parsedHeader;}
}

```

Tabul'ka: Knižnica HRequest (zdrojový kód)¹

¹ *Implementing PayPal Express Checkout* [online] 2008 [cit. 2012-04-05]. Anglicky. Dostupný z URL <<http://www.php-suit.com/paypal>>.

Príloha 4: Ilustračné obrázky administrátorského rozhrania realizovaného IS



Obrázok: Ilustračný obrázok administrátorského rozhrania¹



Obrázok: Ilustračný obrázok administrátorského rozhrania²

¹ Zdroj: vlastný

² Zdroj: vlastný

Príloha 5: Ilustračné obrázky užívateľského rozhrania Secret Games (časť realizovaného IS)



Obrázok: Ilustračný obrázok Secret Games¹

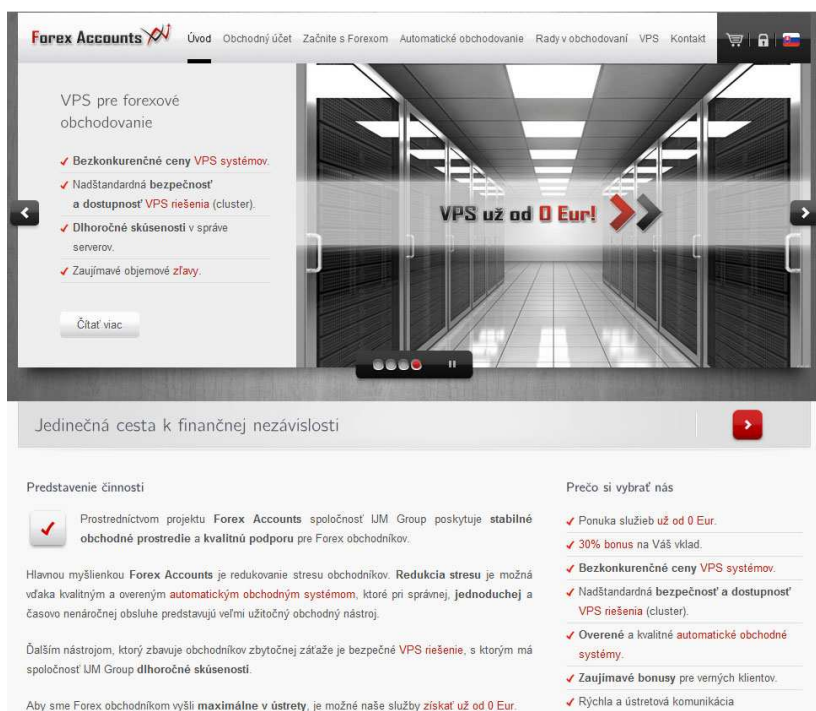


Obrázok: Ilustračný obrázok Secret Games²

¹ Zdroj: vlastný

² Zdroj: vlastný

Príloha 6: Ilustračné obrázky užívateľského rozhrania Forex Accounts (časť realizovaného IS)



Obrázok: Ilustračný obrázok Forex Accounts¹



Obrázok: Ilustračný obrázok Forex Accounts²

¹ Zdroj: vlastný

² Zdroj: vlastný