

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačních technologií**



**Diplomová práce**

**Realizace e-shopu na základě open source řešení**

**Vladimír Pacejka**

**©2017 ČZU v Praze**

## ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Vladimír Pacejka

Informatika

Název práce

**Realizace e-shopu na základě open source řešení**

Název anglicky

**Realization of e-shop on open source base**

---

### Cíle práce

Diplomová práce je tematicky zaměřena na realizaci elektronického obchodu s využitím open source řešení. Hlavním cílem práce je analýza možností vytvoření funkčního řešení e-shopu s konkrétní realizací na základě vybraného open source e-commerce řešení s návazností na vybraný účetní software pro potřeby malých a středních podniků. Dílčími cíli jsou:

- komparace nejpoužívanějších e-shop a e-commerce open source řešení,
- vypracování stručného přehledu a vývoje B2C,
- analýza trhu s účetním SW v České republice pro potřeby malých a středních podniků.

### Metodika

Metodika řešené problematiky diplomové práce je založena na studiu a analýze odborných informačních zdrojů. Praktická část je zaměřena na analýzu možností realizace a samotnou realizaci elektronického obchodu s využitím vybraného open source e-commerce řešení pro potřeby malých a středních podniků (SME). Do komparace open source e-commerce řešení a analýzy trhu s účetním SW budou zahrnuty nejpoužívanější řešení a SW s důrazem na podmínky České republiky a potřeby SME. Na základě syntézy teoretických poznatků a výsledků praktické části budou formulovány závěry diplomové práce.

## Doporučený rozsah práce

60 – 80 stran textu.

## Klíčová slova

Redakční systém, open source, e-shop, e-commerce, PHP, HTML, CSS

---

## Doporučené zdroje informací

- D. Barker.: Web Content Management: Systems, Features, and Best Practices. O'Reilly Media, 2015. ISBN 978-1491908129
- D. Beleščák.: Vytváříme e-shop ve WordPressu pomocí WooCommerce. Computer press. 2014
- E. Castro, B. Hyslop.: HTML5 a CSS3. Computer press. 2012. ISBN: 978-80-251-3733-8
- J. Horton.: Presta Shop. Computer press. 2011. ISBN: 978-80-251-3441-2
- L. Šestáková.: WordPress – Vlastní web bez programování. Computer press. 2013. ISBN: 978-80-251-3832-8
- Marriott J., Waring E. : Official Joomla! Book, The, 2nd Edition. Addison-Wesley Professional. 2012. ISBN: ISBN-13: 978-0-321-82154-6
- MIKULÁŠKOVÁ, P. – SEDLÁK, M. *Jak vytvořit úspěšný a výdělečný internetový obchod*. Brno: Computer Press, 2012. ISBN 978-80-251-3727-7.

---

## Předběžný termín obhajoby

2016/17 LS – PEF

## Vedoucí práce

Ing. Pavel Šimek, Ph.D.

## Garantující pracoviště

Katedra informačních technologií

---

Elektronicky schváleno dne 21. 10. 2016

**Ing. Jiří Vaněk, Ph.D.**

Vedoucí katedry

---

Elektronicky schváleno dne 24. 10. 2016

**Ing. Martin Pelikán, Ph.D.**

Děkan

V Praze dne 12. 03. 2017

### **Čestné prohlášení**

Prohlašuji, že svou diplomovou práci Realizace e-shopu na základě open source řešení jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze, dne 31. 3. 2017

.....

### **Poděkování**

Tímto bych rád poděkoval Ing. Pavlu Šimkovi, Ph.D. za cenné rady a připomínky vedoucí ke zlepšení úrovně práce a jejímu zdárnému dokončení. Poděkovat bych chtěl i Ing. Václavu Lohrovi, Ph.D. za poskytnutí šablony pro vytvoření pluginu. V poslední řadě bych chtěl poděkovat i svým přátelům a rodině.

## **Realizace e-shopu na základě open source řešení**

### **Souhrn**

Diplomová práce je tematicky zaměřená na tvorbu e-shopu na základě open source řešení. Cílem je analyzovat možnosti vytvoření funkčního řešení e-shopu s konkrétní realizací na základě vybraného open source e-commerce řešení s návazností na vybraný účetní software pro potřeby zejména malých a začínajících, ale i středních podniků. Práce za použití jazyků a technologií PHP, HTML, CSS a XML demonstruje realizaci tvorby elektronického obchodu využitím redakčního systému a dalších jeho rozšíření v podobě pluginů a vlastního kódu. Navržené řešení uspokojuje požadavky pro fungování e-shopu, jeho administraci, tvorbu objednávek a jejich fakturaci a synchronizaci skladových položek s účetním softwarem. Řešení je založené na open source bázi vhodné pro začínající a malé podniky a lze ho použít jako návod pro tvorbu ekvivalentního nebo podobného e-shopu.

### **Klíčová slova**

Redakční systém, Open source, E-shop, E-commerce, PHP, HTML, CSS, XML, CMS

## **Realization of e-shop on open source base**

### **Summary**

This diploma thesis is thematically focused on realization of e-shop based on open source solution. The goal is to analyze the possibilities of realization of a working e-shop solution based on chosen open source e-commerce platform with the connection to an accounting software suitable for small, beginning and also midsized companies. With use of PHP, HTML, CSS and XML, this thesis demonstrates the most important aspects of realization of content management system based e-shop, including usage of various plugins and own code extensions. Suggested solution satisfies the requirements for a functional e-shop, its administration, order creation and the creation of invoices and also the synchronization of the stock items with the accounting software. The solution is open source based and it's suitable for beginning and small companies and it could be used as a manual for making an equivalent or similar e-shop.

### **Key words**

Content management system, Open source, E-shop, E-commerce, PHP, HTML, CSS, XML, CMS

# Obsah

<b>1</b>	<b>Úvod</b>	<b>8</b>
<b>2</b>	<b>Cíl práce a metodika</b>	<b>10</b>
<b>3</b>	<b>Teoretická východiska</b>	<b>11</b>
3.1	E-business a E-commerce . . . . .	11
3.1.1	Elektronické podnikání . . . . .	11
3.1.1.1	Kategorizace vztahů v elektronickém podnikání . . . . .	12
3.1.2	Elektronické obchodování . . . . .	13
3.1.2.1	Dimenze elektronického obchodování . . . . .	14
3.1.2.2	E-shop . . . . .	15
3.1.2.3	Funkce e-shopu . . . . .	16
3.1.2.4	Platební modely . . . . .	17
3.1.2.5	CRM v prostředí e-shopů . . . . .	18
3.1.2.6	ERP systémy . . . . .	19
3.1.2.7	Elektronická Výměna dat . . . . .	20
3.1.2.8	Klasické obchodování a obchodování dnes . . . . .	21

3.2	Systémy pro správu obsahu . . . . .	22
3.2.1	Data, Informace a obsah . . . . .	23
3.2.2	Technologie . . . . .	25
3.2.2.1	HTML . . . . .	25
3.2.2.2	Kaskádové styly . . . . .	27
3.2.2.3	PHP . . . . .	28
3.2.2.4	Javascript . . . . .	29
3.2.2.5	Aktuální stav . . . . .	29
3.2.3	Open-source systémy . . . . .	30
3.2.3.1	Wordpress . . . . .	31
3.2.3.2	Joomla . . . . .	32
3.2.3.3	Drupal . . . . .	32
3.2.3.4	PrestaShop . . . . .	33
3.2.4	Zabezpečení . . . . .	34
3.2.5	Šablony a pluginy . . . . .	35
3.2.5.1	Šablona . . . . .	35
3.2.5.2	Plugin . . . . .	35



3.3	Vícekritériální rozhodování . . . . .	36
3.4	Účetní software . . . . .	37
<b>4</b>	<b>Praktická část</b>	<b>39</b>
4.1	Výběr CMS . . . . .	39
4.1.1	Kritéria . . . . .	39
4.1.2	Stanovení vah . . . . .	41
4.1.3	Varianty a jejich ohodnocení . . . . .	42
4.2	Realizace . . . . .	43
4.2.1	Instalace redakčního systému Wordpress . . . . .	43
4.2.2	Prvotní konfigurace redakčního systému . . . . .	45
4.2.3	Volba šablony . . . . .	48
4.2.4	Pluginy . . . . .	48
4.2.4.1	Instalace WooCommerce . . . . .	49
4.2.4.2	Instalace TC WooCommerce Product Slider . . . . .	52
4.2.4.3	Instalace WooCommerce Auto Restore Stock . . . . .	53
4.2.5	Základní vzhled . . . . .	53
4.2.6	Detailní nastavení eshopu . . . . .	54
4.2.7	Úpravy vzhledu stránek . . . . .	55
4.2.7.1	Přizpůsobení šablony . . . . .	55
4.2.7.2	Menu . . . . .	56

4.2.7.3	Tvorba stránek	56
4.2.7.4	Úprava CSS	63
4.2.7.5	Úprava postranní lišty	63
4.2.7.6	Vkládání obsahu	65
4.2.7.7	Další rozšiřující funkcionality	66
4.2.8	Napojení eshopu na účetní software	67
4.2.8.1	Propojení s flexibee	67
4.2.8.2	Tvorba faktury	68
4.2.8.3	Aktualizace skladového hospodářství	69
<b>5</b>	<b>Zhodnocení výsledků a doporučení</b>	<b>72</b>
<b>6</b>	<b>Závěr</b>	<b>73</b>
	<b>Přílohy</b>	<b>78</b>

## Seznam obrázků

1	Hierarchie elektronického podnikání[2] . . . . .	12
2	Úrovně subjektů elektronického obchodování[2] . . . . .	14
3	Dimenze e-commerce [1] . . . . .	15
4	Ukázka up-sellingu a cross-sellingu. Zdroj: Volný . . . . .	17
5	Komunikační kanály v kooperačním CRM[2] . . . . .	19
6	Integrace ERP systému[17] . . . . .	20
7	Vztahový diagram realizace obchodní transakce s komunikací se zákazníkem prostřednictvím ICT [2] . . . . .	21
8	Role v implementaci CMS[18] . . . . .	22
9	Tvorba obsahu Zdroj: Vlastní tvorba . . . . .	24
10	HTML struktura[20] . . . . .	26
11	Hierarchie objektového modelu dokumentu.[20] . . . . .	27
12	Open source CMS v České republice[21] . . . . .	30
13	Hlavní vrstvy jádra redakčního systému Drupal[25] . . . . .	33
14	Vytvořená databáze a přístup do ni Zdroj: Vlastní tvorba . . . . .	44
15	Rozbalení CMS na hostingu Endora Zdroj: Vlastní tvorba . . . . .	44
16	Konfigurace CMS WP Zdroj: Vlastní tvorba . . . . .	45
17	Nastavení přístupu do administrace Zdroj: Vlastní tvorba . . . . .	47
18	Administrace redakčního systému Wordpress Zdroj: Vlastní tvorba . . . . .	48
19	Vyhledání WooCommerce pluginu Zdroj: Vlastní tvorba . . . . .	49
20	Konfigurace lokalizace pluginu WooCommerce Zdroj: Vlastní tvorba . . . . .	50
21	Administrační rozhraní po aktivaci WooCommerce Zdroj: Vlastní tvorba . . . . .	52

22	Úvodní stránka v základním zobrazení po prvotní instalaci Zdroj: Vlastní tvorba . . . . .	53
23	Přizpůsobení šablony Zdroj: Vlastní tvorba . . . . .	56
24	Parametry produktu pro plugin WooCommerce Zdroj: Vlastní tvorba . . . . .	65

## Seznam tabulek

1	Základní vztahy subjektů v e-business.[16] . . . . .	12
2	Porovnání verzí wordpress.org a wordpress.com Zdroj: Vlastní tvorba . . . . .	32
3	Saatyho matice Zdroj: Vlastní tvorba . . . . .	42
4	Vyhodnocení variant Zdroj: Vlastní tvorba . . . . .	42

# 1 Úvod

Elektronické obchodování patří téměř mezi každodenní rutinu v moderním životě člověka. Na českém i zahraničním trhu stále vznikají nové a nové e-shopy, které se stávají oblíbenou nákupní platformou. Elektronických obchodů využívá dnes každá velká firma zabývající se prodejem zboží či služeb. Oblast E-commerce se netýká jen velkých firem, ale i drobných živnostníků, nově vznikajících firem, či startupů, které prodávají své zboží pouze prostřednictvím internetu. Problematika tvorby e-shopů je proto stále aktuální a požadavky a nároky na funkcionalitu těchto e-shopů stále rostou. Z pohledu zájemce o prodej zboží může být tvorba e-shopu jednou z hlavních cest k realizaci prodeje produktů a tržeb s tím spojených. Realizace e-shopů pak musí podléhat analýze potřeb zákazníka spolu s tvorbou řešení na míru.

Předkládaná diplomová práce se zaměřuje na problematiku realizace e-shopu na základě open source řešení. Rozebírá jednotlivé možnosti řešení realizace e-shopu, které jsou podrobeny analýze a odpovídají požadavků na konkrétní tvorbu elektronického obchodu. Dále pak popisuje konkrétní realizaci vybraného řešení založeného na využití redakčního systému. Součástí řešení je i napojení na účetní software sloužící pro fakturaci a správu skladu. Práce je rozdělena do dvou hlavních sekcí a to na teoretickou část a praktickou část.

Teoretická část práce popisuje oblast elektronického podnikání a obchodování a vymezuje pojmy, které se v těchto oblastech vyskytují. Dále rozebírá vztahy, subjekty a hierarchii elektronického podnikání, které dále rozpadá na elektronické obchodování, pod které spadají elektronické obchody. Dále jsou zde rozebrány funkce e-shopů, další pojmy, které s elektronickým obchodováním souvisí a další možnosti rozšíření elektronického obchodování. Teoretická část dále popisuje systémy pro správu obsahu spolu s technologiemi, které se v těchto systémech vyskytují. Dále rozebírá aktuální situaci v oblasti redakčních systémů a popisuje aktuálně nejpoužívanější systémy v České republice a rozebírá i oblast zabezpečení těchto systémů, principy šablon a rozšíření v podobě pluginů. Dále popisuje vícekritériální rozhodování, konkrétně metodu vícekritériální analýzy variant a popisuje oblast zdarma dostupných účetních softwarů. Praktická část analyzuje vybrané varianty spolu se zvolenými kritérii pomocí vícekritériální analýzy variant a na základě této analýzy popisuje realizaci vybraného open source řešení. Samotná realizace se skládá z instalace redakčního systému, jeho nastavení a samotné konfigurace elektronického obchodu. Dále se skládá z rozšíření a tvorby vlastního kódu a pluginu pro podporu požadovaných funkcionalit řešení a napojení na požadovaný účetní software.

Problematika tvorby eshopového řešení je problematikou aktuální, zejména se stále se rozvíjející IT oblastí a neustále narůstajícími tržbami v oblasti elektronického obchodování, kam

postupně směřuje stále se zvětšující podíl zákazníků a podnikání. Dostupné řešení tvorby elektronického obchodu je žádoucí zejména pro začínající podnikatele, živnostníky a start-up projekty v oblasti prodeje zboží, výrobků a služeb. Analýza možnosti řešení dostupných open source řešení je potřebná vzhledem k možným odlišnostem požadavků na funkci e-shopu a další prezentace produktů nebo služeb. Jedním z hlavních problémů moderního IT světa je množství dat, které se zpracovávají a v oblasti e-shopů je správa dat zásadním tématem spolu s jejich bezpečností. Proto je nutné hledat řešení, které splňuje výkonnostní požadavky, respektive dané řešení optimalizovat, aby práce s e-shopem byla svižná a přívětivá. Stránky, které se načítají příliš dlouho a práce s nimi je pomalá, ztrácí své návštěvníky a v případě e-shopu i své potenciální kupce.

V rámci elektronického obchodu je nutné zabezpečit citlivá data, která obchodník prostřednictvím e-shopu získá, tak aby se předešlo jejich zneužití. Téma bezpečnosti dat je rovněž aktuální zejména kvůli různým možnostem kybernetických útoků, které se rovněž jako oblast IT stále zdokonalují a rozrůstají. S nákupem na e-shopu souvisí i obchodní podmínky, které je nutné do realizace zpracovat.

## 2 Cíl práce a metodika

### Cíl práce

Diplomová práce je tematicky zaměřena na realizaci elektronického obchodu s využitím open source řešení. Hlavním cílem práce je analýza možností vytvoření funkčního řešení e-shopu s konkrétní realizací na základě vybraného open source e-commerce řešení s návazností na vybraný účetní software pro potřeby malých a středních podniků. Dílčími cíli jsou komparace nejpoužívanějších e-shop a e-commerce open source řešení s vypracování stručného přehledu a vývoje B2C. Dalšími dílčími cíli jsou analýza trhu s účetním SW v České republice pro potřeby malých a středních podniků.

### Metodika

Metodika řešené problematiky diplomové práce je založena na studiu a analýze odborných informačních zdrojů. Praktická část je zaměřena na analýzu možností realizace a samotnou realizaci elektronického obchodu s využitím vybraného open source e-commerce řešení pro potřeby malých a středních podniků (SME). Do komparace open source e-commerce řešení a analýzy trhu s účetním SW budou zahrnuty nejpoužívanější řešení a SW s důrazem na podmínky České republiky a potřeby SME. Na základě syntézy teoretických poznatků a výsledků praktické části budou formulovány závěry diplomové práce.

## 3 Teoretická východiska

### 3.1 E-business a E-commerce

Pojmy e-business a e-commerce jsou často nesprávně označovány jako ekvivalentní či zaměnitelné jeden s druhým. Ve skutečnosti ovšem představují tyto pojmy dva rozdílné koncepty. E-business jakožto elektronické podnikání zastupuje širší oblast, do které patří mnoho aktivit, které mají za úkol zejména zvýšit efektivitu podnikových procesů. Pod elektronické podnikání spadají i vztahy mezi obchodními partnery, kteří komunikují též prostřednictvím elektronických médií. E-business tedy aplikuje podnikání a jeho náležitosti v prostředí internetu. Rozdíl mezi elektronickým podnikáním a obchodováním je též v míře transformace obchodní činnosti podniku pomocí internetu. Elektronické podnikání využije jakékoliv procesy, které podnikatel nebo firma provádí prostřednictvím internetu včetně zahrnutí procesů jako je zákaznický servis, výuka, správa informačního systému, logistika, zásobování, systémy pro elektronickou výměnu dat atd. E-business byl z počátku chápán pouze jako elektronické obchody a rezervační systémy, ale dnes se tyto aktivity označují pojmem e-commerce tedy elektronické obchodování, které představuje pouze část elektronického podnikání. [1,2]

#### 3.1.1 Elektronické podnikání

ISO/EIC, respektive český normalizační institut definuje elektronické podnikání následovně. Elektronickým podnikáním rozumíme sérii procesů mající jasně srozumitelný účel, zahrnujících více než jednu organizaci, realizovaných prostřednictvím výměny informací a řízených směrem k vzájemně odsouhlaseným cílům, které probíhají během daného časového intervalu.[1, 2]

Elektronické podnikání má v souvislosti s elektronickým obchodováním a internetovými obchody následující hierarchii. E-business je v této hierarchii nejvýše postaven.





Obrázek 1: Hierarchie elektronického podnikání[2]

V rámci elektronického podnikání se definují následující základní vztahy subjektů.

	<b>Podnik (B)</b>	<b>Spotřebitel (C)</b>	<b>Správa (G)</b>
<b>Podnik (B)</b>	B2B Mezipodnikové transakce	B2C Internetové obchody	B2G Nabídka zboží a služeb
<b>Spotřebitel (C)</b>	C2B Spotřebitel prodává firmám	C2C Aukce	C2G Volby, podávání daňových příznání
<b>Správa (G)</b>	G2B Granty, dotace, veřejné zakázky	G2C Portály veřejné správy	G2G Spolupráce mezi státními orgány

Tabulka 1: Základní vztahy subjektů v e-business.[16]

### 3.1.1.1 Kategorizace vztahů v elektronickém podnikání

**Vztah Podnik - Podnik[1, 2, 16]** Tento vztah často funguje automaticky díky výměně dat skrze podnikové systémy (pomocí EDI) a v nejsložitějších případech pak tyto systémy fungují jako komunikační a distribuční síť. Standardně jde o velké objednávky například mezi prodejci a dodavateli. V tomto sektoru se také často využívá uzavřených elektronických tržišť.

**Vztah Podnik - Spotřebitel[1, 2, 16]** V dnešní době se jedná o nejznámější vztah, který je prováděn zcela běžně. Jde o přímý prodej koncovým zákazníkům, jehož základní snahou je

udržovat zákazníka informovaného v oblasti produktů, kdy webová stránka nahrazuje úlohu letáků či katalogů. Do této úrovně patří i různé interaktivní formuláře, přičemž nejvyšší úrovní je samotný internetový obchod.

**Vztah Podnik - Správa[1, 2, 16]** Jedná se o obchodní a komunikační vztahy obchodníků s úřady a orgány státní správy většinou na úrovni místní správy. Rovněž do této kategorie patří komunikace obchodníků s finančními institucemi (elektronické daňové přiznání, elektronické podpisy atd.)

**Vztah Spotřebitel - Podnik[1, 2, 16]** Model obsahující z pravidla tři účastníky. Spotřebitel vystupuje jako prodejce zboží, podnik jako kupující a třetí strana zprostředkovávající komunikaci.

**Vztah Spotřebitel - Spotřebitel[1, 2, 16]** Vztah standardně dvou lidí, využívajících internetu pro prodej zboží ať již formou bazarové inzerce, burzy, nebo aukce. Komunikace často rovněž probíhá pouze elektronicky.

**Vztah Spotřebitel - Správa[1, 2, 16]** Většinou jde pouze o formu elektronické komunikace s úřady a státní správou.

**Vztah Správa - Podnik[1, 2, 16]** Tento model vyjadřuje zejména komunikaci a zprostředkování veřejných zakázek, informací o grantech a dotacích státními organizacemi a úřady.

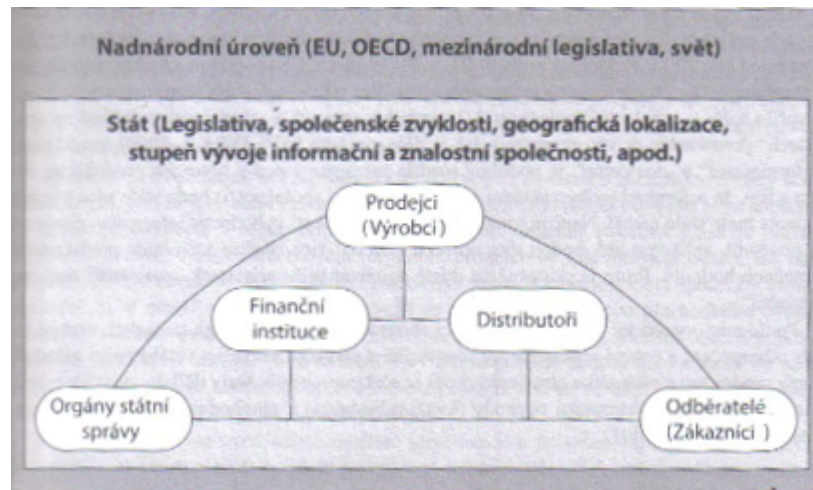
**Vztah Správa - Spotřebitel[1, 2, 16]** Komunikační vztah státní správy a občanů republiky

**Vztah Správa - Správa[1, 2, 16]** Elektronická komunikace různých orgánů veřejné moci, úřadů atd. Jde o čistě nekomerční vztah.

### 3.1.2 Elektronické obchodování

E-commerce, tedy elektronické obchodování, využívá elektronické, respektive internetové prostředky a technologie k realizaci obchodování v rámci podniků, mezi podnikatelskými

subjekty a nejčastěji k podnikání v modelu podnik - spotřebitel. Dnes je již považováno za standard a nabízí skvělé alternativní možnosti prodeje produktů i služeb. E-commerce představuje v současnosti obchodování využívající ke komunikaci zejména informační komunikační technologie. Do této oblasti spadá přímý prodej spotřebitelům online, platby, elektronická komunikace se spotřebiteli (dnes také prostřednictvím sociálních sítí), výměna informací (EDI) atd. [1, 2]

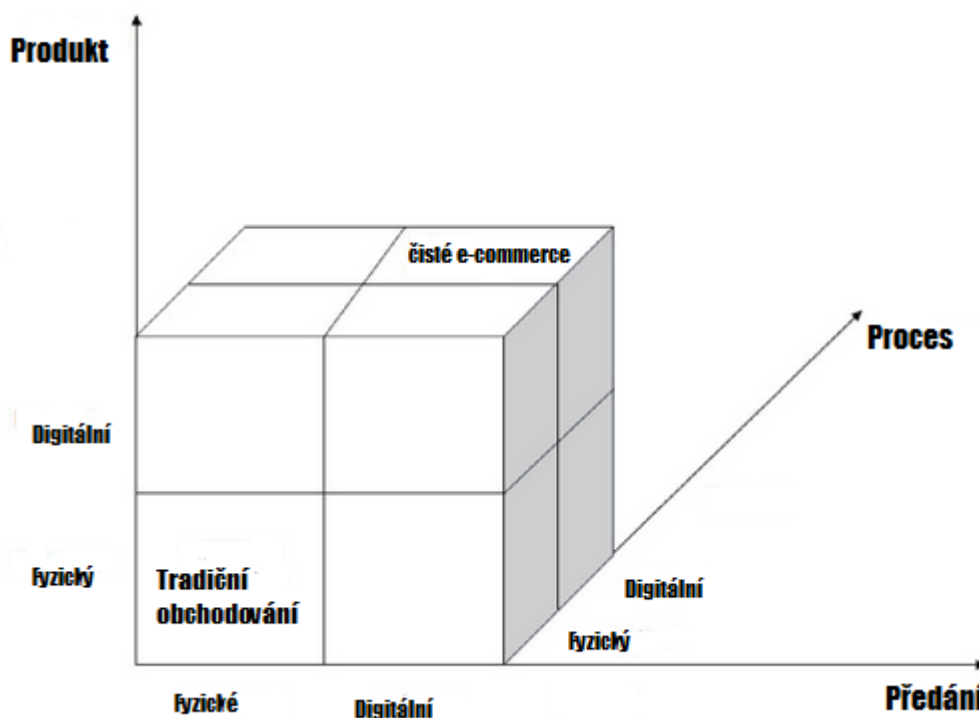


Obrázek 2: Úrovně subjektů elektronického obchodování[2]

Aktuální definice dle ISO/EIC, respektive dle ČSNi charakterizuje elektronické obchodování jako sérii procesů souvisejících s průběhem obchodních transakcí, které probíhají mezi dvěma a více účastníky, jejichž cílem je uzavřít obchod, a které jsou realizovány elektronickými prostředky probíhající v určitém časovém horizontu. Do oblasti e-commerce spadá například distribuce, nákup a prodej, ale i marketing, rovněž i servis a reklamační procedura prodávaných produktů respektive podpora služeb a hlavně komunikace prostřednictvím internetu a elektronických systémů.[1, 2]

### 3.1.2.1 Dimenze elektronického obchodování

Elektronické obchodování se může lišit ve formách, které se odvíjí od míry digitalizace produktu nebo služby, procesu a předání. V roce 1997 vznikl určitý rámec popisující možné scénáře na základě stanovení třech dimenzí. Produkt či služba, proces a předání mohou být fyzické nebo digitální. Tradiční obchodování představuje fyzické předání, procesy i službu. Čistě elektronické obchodování je naopak ve všech třech dimenzích digitální, tedy nákup digitálního produktu pomocí digitálního procesu s digitálním předáním. Ostatní scénáře jsou různými kombinacemi fyzické nebo digitální formy dané dimenze.[1]



Obrázek 3: Dimenze e-commerce [1]

### 3.1.2.2 E-shop

S rozmachem internetu ve světě přišlo již zmiňované elektronické obchodování, respektive obchodování se zbožím a službami pomocí elektronických obchodů, které do jisté míry mohou nahradit kamenné prodejny. Z počátku představoval e-shop poměrně nedůvěryhodnou možnost nákupu zboží, zejména pro nemožnost si zboží vyzkoušet, nebo ho vidět na vlastní oči. Z dalšího hlediska to byl fakt, že zákazník platil za fyzický produkt, který nemohl bezprostředně převzít a ihned začít využívat. V České republice tento problém přetrvával déle než v ostatní státech, nicméně lidé si přivykli pohodlí nákupu z domova, platbě online a při aktuálních zákonem stanovených garancí je nakupování na internetu poměrně bezpečné. Oblíbenou kombinací je dnes nákup na e-shopu a vyzvednutí v kamenné prodejně společnosti či partnera, jelikož uživatel převezme zboží zpravidla zdarma, může si ho překontrolovat, ale zůstává mu pohodlí nákupu, výběru a hlavně záruk plynoucích z nákupu na internetu, jako je například možnost vrátit zboží do čtrnácti dnů bez udání důvodu.[1, 2, 16]

Za e-shop můžeme považovat online obchod se statickým katalogem produktů či služeb, ale i online obchod s rozšířenými možnostmi jako je interaktivní katalog, seznam, recenze a další funkcionality plynoucí ze stále se rozšiřující oblasti informačních technologií, které umožňují neustálé zlepšení těchto služeb poskytovaných spotřebitelům. Hlavní platformou, na které internetový obchod běží, je webová aplikace, což jsou v tomto případě defakto

webové stránky s rozšířenou funkcionalitou pro prodej zboží a služeb. Dále e-shopy využívají webové portály finančních institucí, jako jsou banky pro zprostředkování nejčastější formy platby. Jedná se o internetové služby, které jsou přístupné dnes z počítačů, tabletů i smartphonů tedy mobilních komunikačních zařízení, přičemž tato forma obchodu bývá občasně označována jako m-commerce.[1, 2]

### 3.1.2.3 Funkce e-shopu

Cílem webových stránek bývá nejčastěji prezentace, a proto i na stránkách e-shopu je hlavním cílem produkt či službu prezentovat za účelem prodeje, respektive tvorby zisku. Prezentace produktu či služby by se dala nazvat vědou, jelikož v dnešní době používání internetu se jedná o zcela zásadní a rozhodující část v prostředí elektronického obchodování. Je třeba uživatele zaujmout nejen graficky, ale i smysluplností a věcností textu. Uživatel by se měl dozvědět informace potřebné k nákupu. Stejně tak by proces nákupu na e-shopu měl být přehledný, jednoduchý a uživatelsky přívětivý, čímž se v informačních technologiích zabývá oblast user experience.[15]

Struktura dnešních e-shopů je víceméně standardizovaná a odlišnosti mezi e-shopy co se funkcionalit týče, bývají minimální. Odlišnosti můžeme najít ve volbě prezentace produktů, grafice a službách navíc. Katalog produktů, průvodce nákupem nebo objednávkou a nákupní košík tvoří základ každého internetového obchodu. Proces realizace objednávky by měl být pro zákazníka intuitivní a měl by proběhnout v co nejméně krocích. S procesem objednávky souvisí i jedna z klíčových funkcionalit pro majitele e-shopu. Těmi jsou data o zákaznících. Ať už data byla získána prostřednictvím webové analytiky, které obecně vypovídají o zákaznících, nebo šlo o data, která zákazník sám a dobrovolně o sobě vyplní. Registrace zákazníka v obchodě poskytuje mnohé výhody jak pro něj samotného tak i pro prodejce. Registrovaný zákazník nemusí při každé objednávce znovu vyplňovat údaje a prodejce může zákazníkovi nabízet produkty na míru s ohledem na to, co si u něho zakoupil dříve, popřípadě na základě toho co si dotyčný prohlížel. Na základě těchto informací mu může zasílat, nejčastěji emailem, který od zákazníka získal, různé newslettery a jiná marketingová sdělení.[2]

### Modely registrace zákazníka[2]

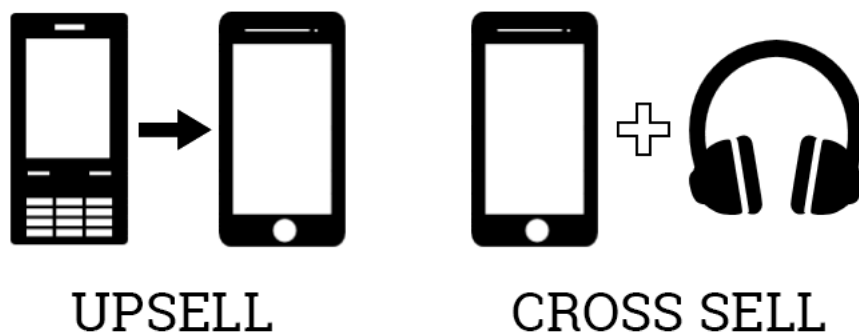
Cílem ověření registrace je možnost nabídnout přihlášenému, tedy registrovanému zákazníkovi například výhodnější ceny a přimět ho tak nakoupit právě u tohoto obchodníka. Nejčastějším modelem v e-shopu je model, kdy se po zákazníkovi nevyžaduje přihlášení a ověření, zdali jde o zákazníka registrovaného, se provádí, až v momentě kdy realizuje svojí objednávku. Další možností je ověření registrovaného zákazníka po vložení zboží do košíku.

Ověření zákazníka před vložením zboží do košíku umožňuje promítnout slevu hned, nikoliv až po přihlášení a zpětném přepočtu. Posledním modelem, který se vyskytuje zejména u e-obchodů s uzavřeným členstvím, nebo na elektronických tržištích, je model vyžadující striktní registraci po příchodu na internetové stránky. Nevýhoda tohoto modelu představuje nemožnost nových zákazníků prohlédnout si jakékoliv zboží na takovém webovém portále.

### **Cross-selling a up-selling[16]**

Cross-selling do češtiny přeloženo jako křížový prodej je aktivita, jejíž účelem je zvýšit celkovou cenu objednávky zákazníka doporučením souvisejícího produktu s produktem, který si zákazník již vybral. Jde tedy o doplňkové zboží neboli komplementy. K nákupu outdoorové kamery je možno doporučit například náhradní baterii, k mobilnímu telefonu zase tvrzené sklo, ochrannou fólii displeje, nebo sluchátka.

Up-selling má za cíl rovněž navýšit celkovou cenu objednávky. Na rozdíl od cross-sellingu se však prodejce snaží prodat zákazníkovi dražší řešení, nebo lepší produkt. Může jít tedy o nabídnutí novější nebo výkonnější verze outdoorové kamery, nebo mobilního telefonu za vyšší cenu.



Obrázek 4: Ukázka up-sellingu a cross-sellingu. Zdroj: Volný

#### **3.1.2.4 Platební modely**

Nedůvěra v elektronické platební systémy představovala hlavní problém pro rozvoj e-commerce, vzhledem k nedůvěře uživatelů. To již v dnešní době neplatí a zejména tuzemské elektronické platby jsou na denním pořádku. Platit v e-shopech je dnes možné elektronicky a to bankovním převodem, převodem elektronické hotovosti, nebo přímo na místě vyzvednutí v hotovosti nebo platební kartou anebo při předání převozci, tzv. dobírkou.[2]

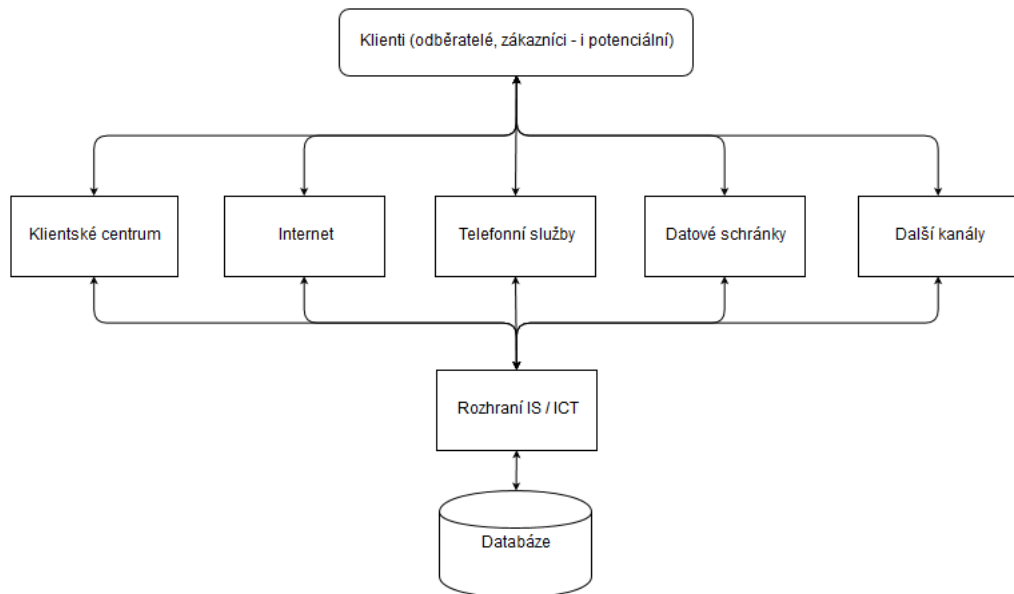
Elektronická platba je technologicky brána jako transakce, tedy posloupnost operací vedoucí k převodu peněz mezi lidmi či společnostmi. Platba bankovním převodem je nejtypičtější v prostředí vztahu podnik - podnik. V e-shopech ve vztahu podnik - spotřebitel se tento způsob platby příliš často nevyužívá. Převod elektronické hotovosti probíhá zprostředkovaně přes banku. Spotřebitel ukládá skutečné peníze do banky, ta je smění za peníze digitální a těmi spotřebitel hradí svůj nákup. Banka v tomto procesu před skutečným předáním digitálních peněz musí převod elektronicky podepsat. Dříve se využívali pro platby i elektronické šeky. Spotřebitel poukáže obchodníkovi šek s elektronickým podpisem a certifikátem, zatímco peníze jsou bezpečně uloženy v bance. Na základě podpisu a certifikátu je obchodník oprávněn k strhnutí dané částky z účtu spotřebitele. Nejpoužívanější metodou dnes je použití platební karty. Princip je podobný jako platba kartou v kamenných prodejnách s tím rozdílem, že platba je realizována elektronicky například skrze paypal či jinou platební bránu.[2]

### 3.1.2.5 CRM v prostředí e-shopů

Ani v prostředí elektronických obchodů není možné zapomínat na péči o zákazníka. Pro tyto účely slouží právě CRM systémy (anglicky Customer relationship management). CRM využívá lidí, procesů a databázové technologie, ve které shromažďuje, zpracovává a používá získané informace o zákaznících (spotřebitelích). Díky těmto informacím je možné pochopit a predikovat potřeby, přání a zvyky zákazníků během nakupování a také zlepšit obousměrnou komunikaci. Rovněž se tyto informace využívají pro tvorbu dlouhodobých byznys plánů.[1, 2]

Hlavním cílem je získat korektní a konzistentní data a ty propagovat do celého informačního systému. Klíčem k dobrému CRM je využití správné informace ve správný čas a na správném místě. Systémy pro řízení vztahů se zákazníky lze rozdělit na kooperační, operativní a analytické.[1, 2]

**Kooperační CRM[2]** Kooperace je založena na komunikaci prostřednictvím rozličných kanálů. Komunikace může probíhat skrze online chaty, fóra, emaily, SMS nebo jiné služby. Informace z kanálů jsou získány pomocí interakce se zákazníkem například konkrétními odděleními daného podniku. Cílem je sdílet všechny informace, které získali jednotlivé kanály.



Obrázek 5: Komunikační kanály v kooperačním CRM[2]

**Operativní CRM[2]** Operativní řízení vztahů se zákazníky podporuje zejména byznys procesy v oblasti prodeje, marketingu a poskytovaných služeb. Podnik díky informacím z CRM tvoří a sleduje marketingové kampaně a prodejní proces, který může být do určité míry automatizován. Do oblasti operativního řízení vztahů spadají procesy, jako jsou správa kontaktů, podpora zákazníků, automatizace a také klientská centra.

**Analytické CRM[2]** Analytika se využívá v prostředí internetu a webových stránek dnes již standardně. Rovněž CRM má svou analytickou část, která se používá k analýze zákaznických dat z důvodů optimalizace, také ke zkoumání chování zákazníků, hledání možností a rovněž pro využití cross-selling a up-selling. Vyhodnocená analýza poskytuje podklady pro rozhodování v oblastech prodeje a cílení produktů.

Hlavními přínosy řízení vztahů se zákazníky je zlepšení vztahů se stávajícími a nově přichozími zákazníky. To může vést k nárůstu prodeje, k zefektivnění procesů a podniku je umožněno monitorovat své cílové skupiny zákazníků.

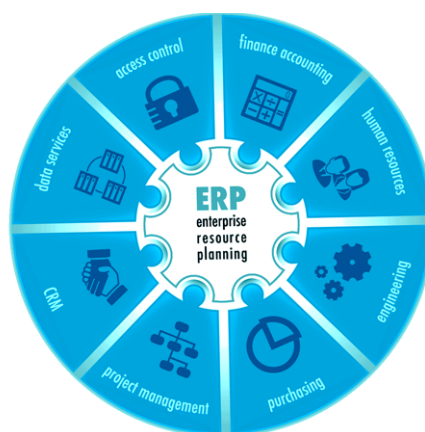
### 3.1.2.6 ERP systémy

Jde o manažerský informační systém pro plánování podnikových zdrojů (z angl. Enterprise resource planning). V ERP je zaintegrovaná řada procesů, které souvisí s produkční činností podniku. Spadá do nich například vývoj, distribuce, logistika, správa majetku, prodej a také fakturace a účetnictví. Propojení ERP a e-business umožňuje spravovat veškeré



podklady na jednom místě v jednotném uživatelském rozhraní. To přináší úsporu času, nákladů, zamezení redundantních dat a data jsou díky tomuto systému lépe propojená. Skrze systém pro plánování podnikových zdrojů je možné řídit a koordinovat veškeré zdroje a aktivity a zajistit tak potřeby podniku. S ERP je možné propojit se systémem pro řízení vztahů se zákazníky a s řízením dodavatelských procesů spolu s procesy pro podporu rozhodování.[2]

Podniky zpravidla požadují, aby ERP systém umožňoval automatizaci a integraci hlavních podnikových procesů. Systém by měl rovněž poskytovat prostředky pro sdílení dat, postupů a ty udržovat standardizované, a také by měl poskytovat informace v reálném čase, nebo umožnit jejich tvorbu. Pro efektivní využití v prostředí e-shopu je třeba dbát na účelné propojení procesů, které elektronický obchod realizuje se stávajícími procesy podniku. ERP představují základní systém každého podniku pro podporu rozhodování a řídicí činnosti s napojením na CRM, HRM (Human resources management - řízení lidských zdrojů), řízení projektu (project management), finanční management, vývoj (engineering), správu dat a správu přístupu (access control).[2, 17]



Obrázek 6: Integrace ERP systému[17]

Hlavními přínosy systému pro plánování podnikových zdrojů je zefektivnění a zlepšení rozvoje strategické a operativní úrovně, zvýšení produktivity, redukce nákladů a celkové dosažení lepší přehlednosti a dostupnosti informací v podniku. ERP systémy jsou většinou dodávány v modulární podobě. Podnik si tedy zvolí z modulů, které pro své podnikání potřebuje. Obvykle je možno zahrnout tyto moduly: účetnictví, řízení majetku, řízení skladu a zásob, řízení výroby, nákup a prodej, závazky a pohledávky, řízení lidských zdrojů, plánování výroby, plánování prodeje, řízení projektů, marketing a mnohé další.[2]

### 3.1.2.7 Elektronická Výměna dat

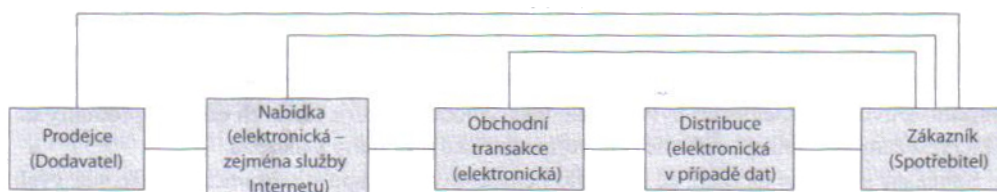
Zkratka EDI z anglického electronic data interchange je definována jako výměna struk-

turovaných dat dle dohodnutých standardů, která probíhá mezi informačními systémy. Tradičně EDI využívá počítačů a globálních komunikačních sítí, nejčastěji internetu. Pro vlastní komunikaci je nejprve proveden tzv. výstup ze systému, například faktura. Tento soubor se následně přenáší pomocí komunikačního softwaru, který soubor přetvoří do standardizovaného formátu pro komunikaci prostřednictvím EDI a soubor odešle pomocí globální sítě zákazníkovi, respektive spotřebiteli. Zákazník s obdobným komunikačním softwarem, který soubor rozklíčuje, přijímá data a ty importuje do spotřebitelova informačního systému. Elektronická výměna dat by měla mít jisté vlastnosti. Zejména jde o integritu, autentičnost, důvěrnost, rychlost, jedinečnost, prokazatelnost, archivaci a spolehlivost. Standardy EDI jsou neustále vylepšovány a tak jsou dnes dostupnější pro vícero výrobců, distributorů a prodejců.[1, 2, 16]

V prostředí e-shopů je tato výměna dat většinou spíše jednosměrná a nejčastěji jde o automatické vygenerování objednávky nebo faktury zákazníkovi na email.

### 3.1.2.8 Klasické obchodování a obchodování dnes

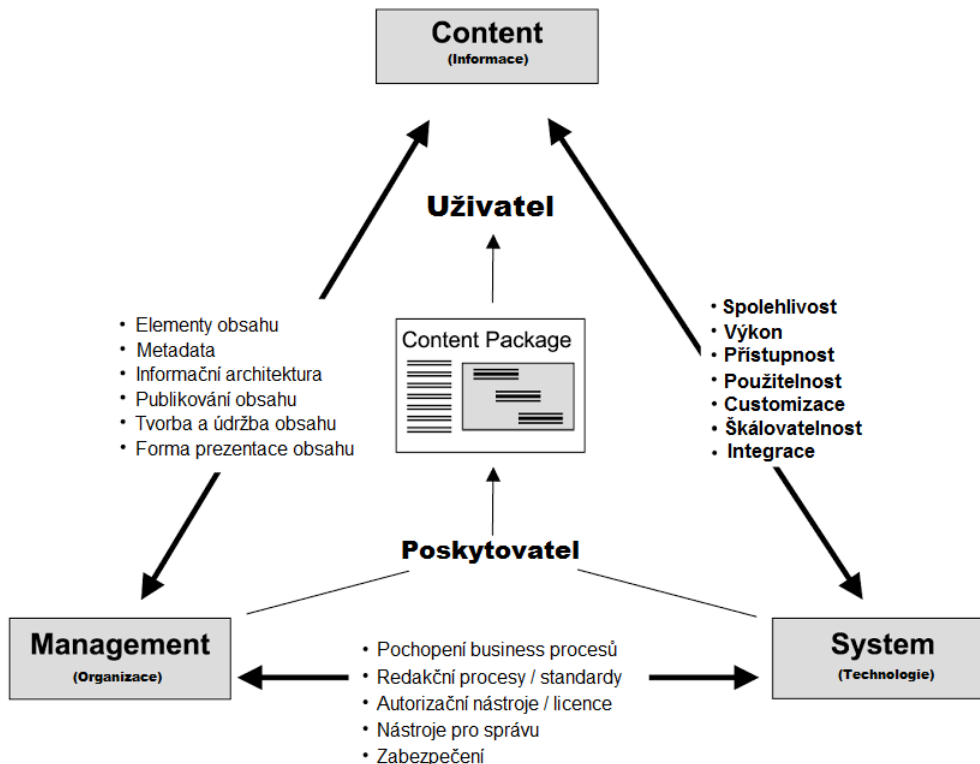
Před příchodem obchodování prostřednictvím internetu se skládal proces obchodu z oslovení spotřebitele prodejcem, kdy na základě nabídky, proběhla obchodní transakce a zboží bylo předáno. Tento proces byl zejména vzhledem k nedostatku elektronické komunikace poněkud zdlouhavý. Dnešní proces je značně zrychlený a při využití elektronické výměny dat a systému automatického skladování a doplňování zásob může být tento proces téměř plně automatizovaný. E-commerce předpokládá, že činnosti potřebné pro realizaci obchodu jsou v co největším rozsahu podporovány informačními a komunikačními technologiemi a probíhají plně nebo částečně automaticky. E-business se dnes též neobejde bez řízení obsahu a k tomu jsou v prostředí e-shopů často využívány systémy pro správu obsahu.[2]



Obrázek 7: Vztahový diagram realizace obchodní transakce s komunikací se zákazníkem prostřednictvím ICT [2]

### 3.2 Systémy pro správu obsahu

Tyto systémy vychází z anglického spojení Content Management (systémy se označují jako Content Management Systems, zkratka CMS), tedy spravování obsahu. Jde o systémy, ve kterých obsah hraje nejzásadnější roli. V češtině se rovněž používá termín redakční systémy. Ať jde o e-shop, blog či jiné stránky, vše jmenované lze skrze tyto systémy spravovat. Dobré stránky představují propletenou síť v podobě propojení s ostatními stránkami a zdroji na webu. Redakční systémy umožňují tvorbu webu skrze obsah snadněji a zejména následná správa takového webu je díky těmto systémům lépe zvládnutelná. Redakční systém tvoří hlavně jeho jádro a na něm se poté dále budují individuální styly (šablony) a vzhledy stránek, popřípadě rozšířené funkcionality. Logicky se tyto systémy skládají z editačního prostředí, úložiště (nejčastěji databázové) a z mechanismu pro publikování. Prostřednictvím CMS lze řídit přístupová práva, správu obsahu a workflow prostřednictvím stavů. Dále má správce systému možnost verzování stránek, článků a dalšího obsahu na webu. Redakční systémy rovněž umožňují znovupoužití i v jiné formě toho, co již bylo vytvořeno. Jako ostatní informační systémy lze díky CMS automatizovat a skládat různý obsah jelikož se nachází v jednom systému. Editorská práce je rovněž značně přehlednější.[4]



Obrázek 8: Role v implementaci CMS[18]

## Historie

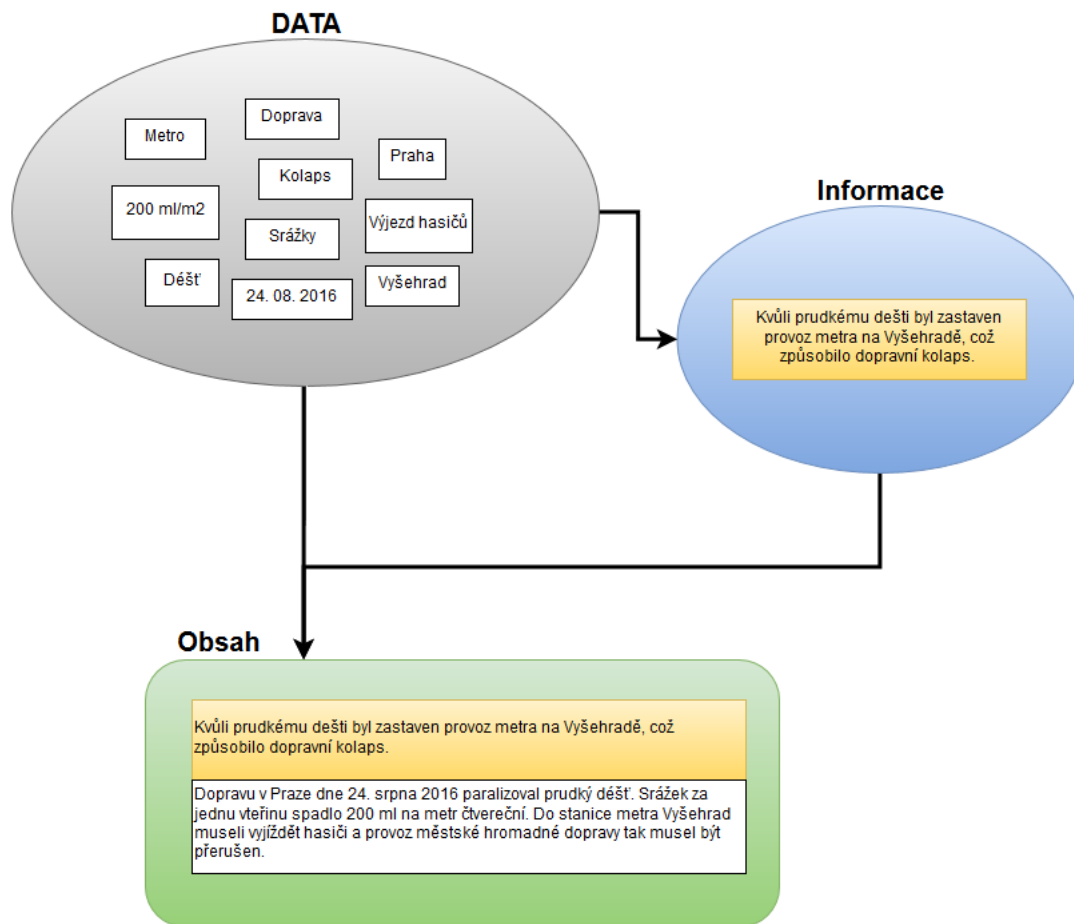
První systémy pro správu obsahu se začaly objevovat zhruba v polovině devadesátých let. Hlavní snahou bylo zprovoznit dynamické stránky pro účely e-commerce. V té době byly obdobné stránky dynamické například pomocí programovacího jazyka Perl. S příchodem PHP započalo programování vlastních systémů pro správu webu a tím se webové stránky stali mnohem přívětivějšími. Následovala komercializace redakčních systémů a příchod prvních firem, které nabízeli své systémy pro správu obsahu s plnou podporou okolo roku 2000. První open source redakční systémy se objevili v roce 2004. Vývoj těchto systémů započal vydáváním vlastních systémů v roce 1997, pokračoval systémy proprietárními v roce 2000 a pokračoval vydáním prvních open-source systémů v roce 2004.[19]

Tyto systémy poskytují uživatelsky přívětivou správu stránek a jejich obsahu, tedy dat a informací respektive znalostí, které jsou nedílnou součástí každého takového systému.

### 3.2.1 Data, Informace a obsah

Správa obsahu představuje ve většině firem nástroj, který firmám pomáhá vypořádat se s přívalem informací, které je třeba transformovat ve znalosti. Ty lze použít na jejich webu a k dalším podnikatelským činnostem. Je třeba ovšem rozlišovat data jako taková a obsah, který může být souvislejšího charakteru.[3]

Počítače zpracovávají data a ta si lze představit například jako čísla, kusy textu, obrázky nebo zvuky. Obsah, anglicky content, se skládá z dat a informací a na rozdíl od dat je podáván i v jistém kontextu. Data jsou nejčastěji uložena v databázích. Náhodná data sami o sobě žádnou hodnotu nepřinášejí, nicméně ve správném kontextu již tvoří informaci, která tvoří určitý obsah. V informatice mají data podobu digitální, která je určena k zpracování počítačem. Zpracovaná data pak představují informaci.[3]



Obrázek 9: Tvorba obsahu Zdroj: Vlastní tvorba

Data jsou tedy malé dílčí údaje, které lidé sbírají a nejčastěji ukládají v databázích. Informace již může být interpretována různě. Nejčastěji jde však o zaznamenané nahrávky komunikace jako jsou texty, články, knihy a jiné. Jde o někým zpracovaná a vhodně interpretovaná data. Informace předaná běžnou formou není obsahem. Obsahem se stává až poté co je informaci dána určitá použitelná forma, využitelná k jednomu nebo více účelům. Hodnota obsahu se odvíjí od kombinace primární účelné formy a aplikací, přístupností a použitelností této informace a do jisté míry i unikátnosti informace. Obsah je zpravidla tvořen využitím informace, kterou máme spolu s daty. Tato data typicky popisují informaci a poskytují jí kontext. Tím je informace použitelná jak počítačem tak rovněž přináší užitek člověku. Příkladem je nadpis sekce, který představuje informaci. Spolu s ostatními nadpisy představuje souhrn informací, které slouží čtenáři pro navigaci v této práci. Díky této navigaci uživatel dohledá kapitoly, které jsou podloženy daty a dosáhne tak uceleného souboru informací a dat předaných určitou formou přinášející hodnotu. Čtenář se touto cestou dostal k obsahu.[3]

Smyslem správy obsahu není přeměna všech dat na obsah. Cílem je dohlédnout na tvorbu a management informací a spravovat integraci této informace s ostatními systémy. Systémy

pro správu obsahu nesou odpovědnost za tvorbu a finální publikaci. Pokud publikace obsahuje data i obsah, pak je na systému, aby zajistil zobrazení dat a obsahu na správných místech a aby publikace působila na webu jednotně a souvisle pro koncového zákazníka, uživatele, nebo čtenáře.

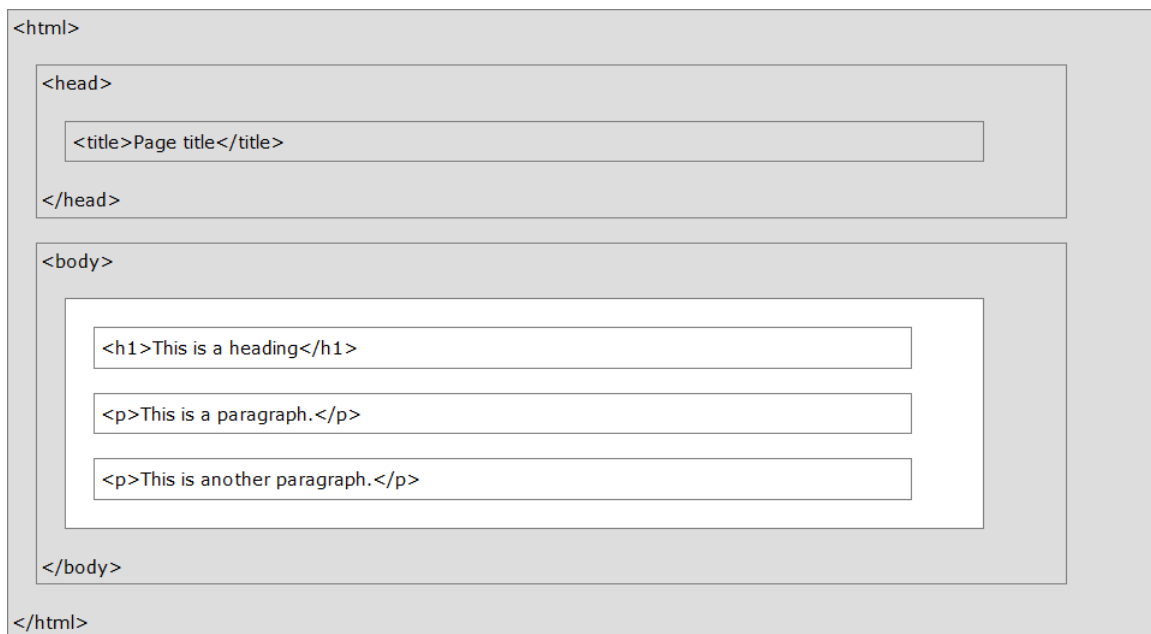
### 3.2.2 Technologie

Tvorba webových stránek se zakládá na technologiích HTML a kaskádových stylů. Rozšířena je pak o PHP, javascript, popřípadě o XML. Stejně tak jako tvorba aplikací jsou i redakční systémy psány v programovacích jazycích a využívají již zmíněných technologií. Wordpress, Joomla, Drupal i PrestaShop mají své jádro psané v PHP, existují však i další systémy psané například v Javě a těmi jsou OpenCMS nebo Ametys a další.[13, 14]

#### 3.2.2.1 HTML

HyperText Markup language neboli hypertextový značkovací jazyk je jazyk sloužící pro výstavbu webových stránek, jenž vycházel z univerzálnějšího značkovacího jazyku SGML. Skrze HTML se tvoří struktura webových stránek a pomocí HTML tagů se formátuje a tvoří obsah nikoliv funkcionalita. HTML patří do technologií klient side, fungující na bázi požadavků. Na základě požadavku prohlížeče se samotný HTML kód překládá u uživatele, kde se zobrazuje požadovaná stránka. HTML obsahuje veškeré funkce základního textového editoru a poskytuje funkce pro práci s grafikou. Hlavním cílem HTML je tvorba webových stránek, které je možno zobrazit kdekoliv na světě.[5, 12]

Původní HTML standard vznikl v laboratořích CERN, ale jeho novější verze po patřičném zanalyzování, probrání a rozhodnutí jsou publikovány konsorciem W3C. HTML stránka má svojí strukturu, která musí být dodržena pro řádný chod webových stránek. Základem je tag HTML, ve kterém jsou uzavřeny sekce hlavička a tělo. Často se také využívá patička, která je zahrnuta v těle.



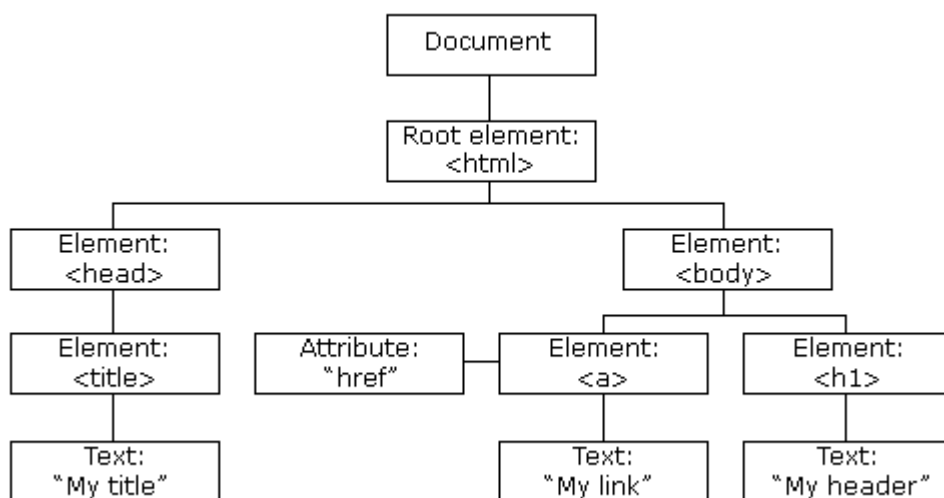
Obrázek 10: HTML struktura[20]

Hlavička HTML stránky obsahuje název a metadata dané stránky. Také jsou v ní namapovány další soubory, například Javascript, kaskádové styly atd. a další vztahy s ostatními stránkami. Samotné tělo stránky představuje obsah, jenž se uživateli zobrazuje. Nachází se zde veškeré texty, grafické prvky, formátování stránky a také patička stránky. Samotná patička představuje záhlaví stránky, dnes již však existuje i patička v tabulce, nebo je možné přiřadit patičku určité sekci v rámci těla stránky. HTML se v průběhu vývoje objevilo i v jiných verzích či obdobách jako DHTML a XHTML. DHTML představuje kombinaci HTML spolu s kaskádovými styly a občasně i s javascriptem pro tvorbu dynamického webu. XHTML je rozšířením HTML standardu, jehož dokumenty jsou založeny na bázi XML. Samotné XML neboli extensible markup language zahrnuje silnou stránku původního SGML se zachováním jednoduchosti HTML standardu. Skrze XML je možno tvořit vlastní značkovací jazyky.[12, 13, 14]

Technologie HTML vstoupila do nové éry s příchodem HTML5. HTML 5 nabízí mnohá zlepšení v oblasti webového designu, rozložení stránek a v oblasti použitelnosti. Webové stránce umožňuje pracovat s grafikou přímo bez nutnosti použití externích pluginů jako Flash, rovněž lze na stránku umístit audio nebo video soubory bez nutnosti přídatných pluginů. Z technologického hlediska představuje programování v HTML5 již spíše programování webové aplikace, jelikož tento standard se mnohem více orientuje na zpracování na úrovni klientské části. Nový standard přináší řadu nových možností, mezi kterými lze například uvést správu geolokací, úloh na pozadí, vylepšenou práci s formuláři a řadu nových HTML tagů pro lepší orientaci v kontextu stránky.[13, 14]

### 3.2.2.2 Kaskádové styly

Kaskádové styly, zkratkou CSS z anglického Cascading style sheets, umožňují popsat prezentaci dokumentu napsaného v HTML nebo XML. CSS, jehož první implementace se datuje na rok 1999, popisuje, jak se jednotlivé elementy zobrazují na obrazovce uživatele. Skrze tyto styly lze upravit například barvy, fonty, odsazení a jiné vizuální detaily webu. Aplikování stylů na dokument funguje díky napojení CSS na DOM (document object model), tedy na objektový model dokumentu. Objektový model dokumentu umožňuje pracovat s jednotlivými elementy HTML stránky jako se samostatnými objekty, které mají své vlastnosti a metody.[5, 14]



Obrázek 11: Hierarchie objektového modelu dokumentu.[20]

Kaskádové styly lze do stránky vložit přímo do hlavičky dané stránky, k čemuž slouží tag `<style>`. Do tohoto tagu lze rovněž kaskádové styly nainportovat z externího souboru. Nej-používanější variantou je však importování kaskádových stylů do hlavičky dokumentu pomocí tagu `<link>`, který umožňuje nainportování nejen kaskádových stylů, ale například i javascriptových souborů. Stylování je možno provést rovněž přímo u elementu, nicméně takovéto stylování vede k nepřehlednosti kódu a složitým úpravám v případě nutnosti provedení změn.[14]

Kaskádové styly se vztahují ke konkrétním identifikátorům nebo ke třídám. Třídy lze používat obecně pro elementy, identifikátory se však používají unikátně a tak kaskádové styly na základě identifikátoru platí pouze pro jednoznačně identifikovaný element v dokumentu. CSS využívá selektorů pro definici stylů. Využívat je možno hierarchie elementů, vztah rodič dítě, nebo obecných selektorů. V CSS lze jednotky vyjadřovat procentuálně, v bodech,



v pixelech, v centimetrech i milimetrech, jednotkách em, jenž odpovídají velikosti aktuálního fontu. Dále také v palcích, v tzv. picas, které odpovídají 12 bodům a v jednotkách exs, které výškou odpovídají malému písmenu x aktuálního fontu.[14]

CSS3, jenž vzniklo současně s HTML5 přináší nové možnosti pro práci se styly. Mezi hlavní rozšíření se řadí práce s pozadím, přechody, ohraničením a možnost transformací prostřednictvím stylování. Zejména přechody a transformace odbourávají nutnost řešit takovéto situace pomocí javascriptu. Díky novým atributům pozadí lze definovat přesnou polohu pozadí a rovněž je umožněno použití vícero pozadí v definici atributem. Díky tomu je možno složit pozadí stránky například ze 4 různých obrázků, bez nutnosti rozdělení stránky na elementy, kde by každý element měl svoje jedinečné pozadí. V rámci ohraničení je možno pracovat se stínováním, zaoblováním a zabarvováním jednotlivých hran. Stínování lze využít i pro text. Žádaným přínosem CSS3 je možnost sloupcování obsahu. Za pomoci definice stylováním rozdělíme obsah elementu na námi definovaný počet sloupců. Transformace skrze kaskádové styly umožňují rotace, zešíkmení, zvětšování či zmenšování ve třech dimenzích. U přechodů se nastavuje jejich celková délka, prodleva a načasování.[14]

### 3.2.2.3 PHP

PHP je programovací jazyk pro tvorbu dynamických a interaktivních webových stránek. PHP programy běží na webovém serveru a poskytují webové stránky uživatelům na základě uživatelského požadavku. Poté co webový server narazí na PHP kód nebo volání nějakého PHP souboru, pak dojde k předání těchto dat PHP procesoru, který standardně běží na hostingu.[5, 6, 14]

Jazyk PHP má své základy v jazycích C a Perl, i když je podobný jazyku JAVA. Základní struktura jazyka dodržuje oddělení jednotlivých příkazů středníkem. Proměnná je definována pomocí symbolu \$. PHP na rozdíl od jiných jazyků nedefinuje proměnou jako integer, string či jiné typy proměnných. Proměnná je určena svým obsahem. K tvorbě polí je však zapotřebí uvést klíčové slovo array. PHP využívá aritmetické operátory, rovněž operátory přiřazení, logické i porovnávací operátory.[6, 14]

Při použití na webových stránkách, lze díky PHP dosáhnout rozdílného obsahu pro různé uživatele a to za pomoci jedné php stránky. Například v oblíbené blogovací platformě Wordpress, má každá velká sekce ve správě webu svůj php soubor a mnoho dalších generických. Právě skrze naprogramovanou logiku v těchto souborech je správa webu zvládnutelná i pro uživatele s nižší respektive žádnou znalostí programování.[6, 14]

Výhodou PHP je řada před-připravených funkcí, které je možno v kódu zavolat a využívat tyto funkce pro účely programování určité logiky. Předem připravené funkce tak například

umožňují snadné propojení s databází a dotazování nad ní, práci s obrázky, či PDF soubory a mnoho dalších funkcí podporující práci s rozličnými soubory či softwary.[6, 14]

#### 3.2.2.4 Javascript

Pro skriptování se na webových stránkách používá skriptovací jazyk Javascript. Ten na rozdíl od PHP funguje na straně klienta a používá se k tvorbě designu stránky a jejího chování na základě spouštěných událostí. Jedno z častých použití javascriptu je dynamická změna obsahu stránky, bez nutnosti jejího znovu načtení. Jelikož Javascript běží přímo v prohlížeči a má tak přímý přístup ke všem elementům webové stránky respektive webového dokumentu, nabízí tak možnosti, kterých by jinak nemohlo být dosaženo jinou cestou.[5, 6, 14]

Tento programovací jazyk se poprvé objevil v prostředí prohlížeče Netscape Navigator v roce 1995. Javascriptu poskytl širší možnosti příchod HTML elementů, které byli strukturované a formální dle objektového modelu dokumentu (DOM). Stejně jako v PHP je i v JavaScriptu syntaxe strukturovaného programování.[13, 14]

JavaScript v HTML je uveden tagem `<script>`. Do tohoto tagu je možno přímo vepsat javascriptový kód nebo prostřednictvím atributů tohoto tagu zavolat javascriptový soubor, ve kterém se skript nachází. Každý příkaz je oddělen novým řádkem a proměnná je volně definována bez nutnosti uvedení `$` jako tomu je v PHP. Operátory využívá Javascript stejně jako PHP a rovněž má svou sadu předem definovaných funkcí, které usnadní tvorbu kódu či programování.[13, 14]

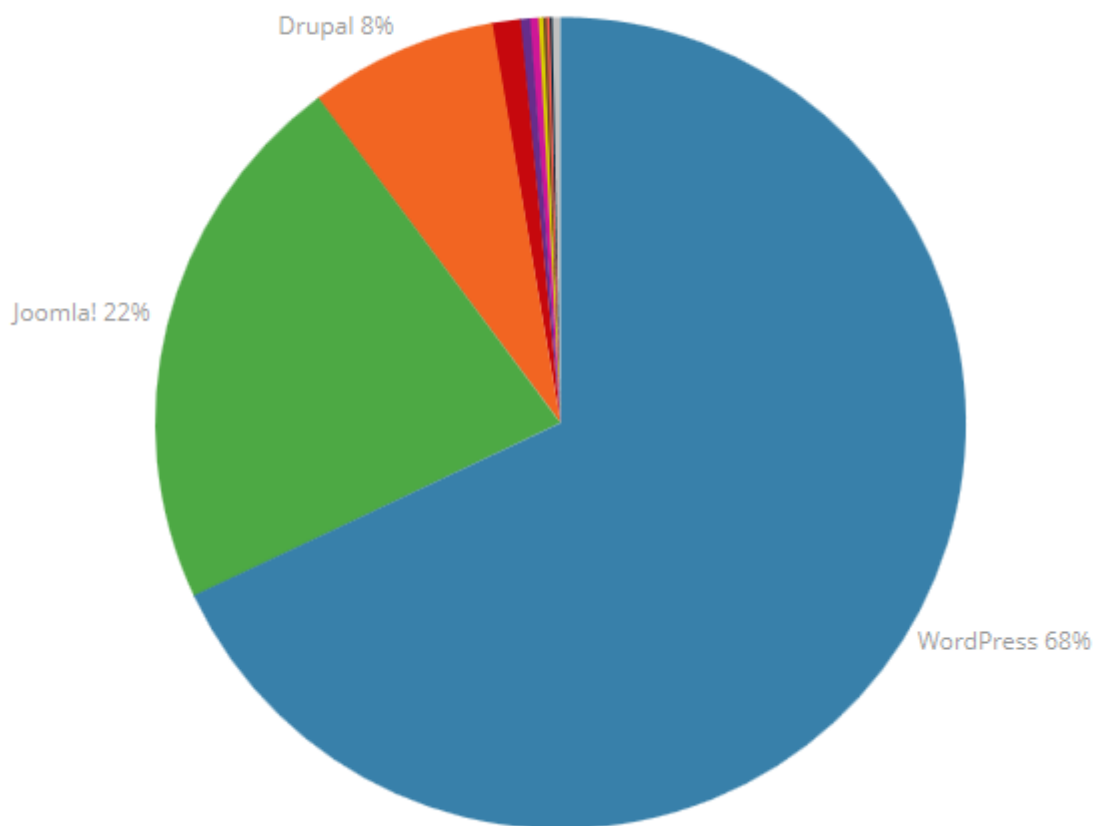
V souvislosti s tímto jazykem se v programování webů a webových aplikací lze setkat s pojmy AJAX a JQUERY. Termín AJAX představuje asynchronní javascript a XML. Jedná se o použití sady metod zabudovaných v javascriptu k přenosu dat mezi prohlížečem a serverem na pozadí. Použití AJAX značně snižuje objem přenesených dat a také umožňuje působit stránce více dynamicky. JQuery je javascriptová knihovna funkcí překlenující rozdíly mezi jednotlivými prohlížeči. Obdobné knihovny funkcí jsou angularJS, MooTools, Prototype, script.aculo.us, YUI a mnoho dalších.[13, 14]

#### 3.2.2.5 Aktuální stav

Aktuálně se v prostředí webových technologií používá HTML 5.1 spolu s CSS3. Nejnovější PHP stabilní verze se označuje jako verze 7.0.11 a u jazyka javascript jde o verzi 1.8. Na novějších verzích zejména PHP a javascriptu se průběžně pracuje, v prostředí HTML se pracuje na verzi 5.2, v CSS se však novější verze v nejbližší době neočekává.

### 3.2.3 Open-source systémy

Open source redakční systémy představují řešení, které je dostupné zdarma. Kód těchto systémů je volně k dispozici a každý do něj může zasahovat dle své potřeby. Nejčastěji se jedná o zásahy do php skriptů, úpravy HTML respektive CSS. Mezi nejznámější redakční systémy zdarma patří Wordpress, Drupal, Joomla a PrestaShop. Všechny tyto systémy je možno volně stáhnout a provozovat na svém serveru, nebo na serveru poskytovatele. Následující graf ukazuje, které systémy jsou v České republice nejčastěji používány.



Obrázek 12: Open source CMS v České republice[21]

Jediný systém přesahující 1% se nazývá TYPO3 a je používán spíše v podnikatelské sféře. Ostatní systémy nedosahují ani na hranici jednoho procenta.

Open source systémy běží standardně na licencích GNU GPL. Licence GNU GPL je všeobecnou veřejnou licenci, která byla navržena pro zajištění volného šíření díla a pro umožnění

jeho úprav. Licence GPL vyžaduje, aby díla odvozená z děl vedených pod touto licencí byla opět volně šiřitelná a dostupná, tedy opět pod všeobecnou veřejnou licencí (GPL GNU).[22]

### 3.2.3.1 Wordpress

Redakční systém Wordpress se proslavil zejména díky blogerství. Předchůdce systému vznikl již v roce 2001, samotný Wordpress pak v roce 2003. Systém Wordpress a jeho architektura je vybudována pomocí PHP a MySQL pod GPL licencí. Wordpress se postupně rozrůstal o nové funkcionality, jako jsou pluginy, themes (témata), prostor pro umístění ovládacích prvků a další. V posledních verzích přišel systém s podporou emoji smajlíků a inline linků.[7, 23]

Mezi hlavní výhody Wordpressu patří jeho silná komunita. Tato komunita vytváří nové šablony, pluginy a zpětnou vazbu vývojářům. Samotnou stránku lze založit dvěma způsoby. Stáhnutím celého systému a nainstalováním na server poskytovatele, respektive na vlastní doménu, nebo v šesti jednoduchých krocích vytvoříte web skrze prohlížeč na stránkách wordpress.com. Druhá varianta umožňuje naprosto běžnému uživateli, neznalému věcí jako je hosting, doména a programování, vytvořit si svojí vlastní stránku během minuty. První varianta s instalací redakčního systému je však rovněž velmi rychle nastavená, nicméně vyžaduje již drobnou znalost, nebo potřebu manuálu v podobě návodu krok za krokem. Samotné nastavení redakčního systému a prvotního nastavení stránky je snadné, rovněž tak běžné používání nevyžaduje velkého úsilí. Další silnou stránkou je možnost rozšiřování základní instalace Wordpressu pomocí pluginů. Tato cesta poskytuje nepřeborné množství možností úpravy funkcí samotného redakčního systému, stejně tak jako ohromné množství rozšíření funkcionalit. Následující tabulka ukazuje drobné rozdíly mezi verzemi samoinstalace redakčního systému z wordpress.org a verzí dostupnou na wordpress.com.[3, 7]

Funkce	wordpress.org	wordpress.com
Cena	Zdarma	Zdarma
Stažení SW	Ano	Ne
Instalace SW	Ano	Ne
Nutnost vlastního webhostingu	Ano	Ne
Customizace kaskádových stylů	Ano	15\$ za rok
Přístup k šablonám	Ano	Jedna vybraná na začátku, dále už ji není možné změnit.
Ovládací prvky v postraní liště	Ano	Ano
RSS syndikace	Ano	Ano
Přístup ke kódu samotného redakčního systému	Ano	Ne
Možnost instalace pluginů	Ano	Ne
Instalace témat	Ano	V omezené míře při počátečním výběru
Přispívání více autory	Ano	Ano
Podpora komunity prostřednictvím fóra	Ano	Ano

Tabulka 2: Porovnání verzí wordpress.org a wordpress.com Zdroj: Vlastní tvorba

### 3.2.3.2 Joomla

Předchůdcem Joomla byl systém Mambo. Samotná Joomla pak vznikla fakticky jako novější verze předešlého systému Mambo v roce 2005. Systém běží na objektově orientované bázi PHP frameworku. Databázová část běží na MySQL a v pozdějších verzích přibyla i podpora jiných databázových serverů než je MySQL. Databáze Joomla rovněž podporuje práci s PostgreSQL. Joomla se postupně vyvíjela a rozšiřovala své funkcionality a jako jeden z prvních velkých redakčních systémů poskytovala responzivní web jako standard.[8]

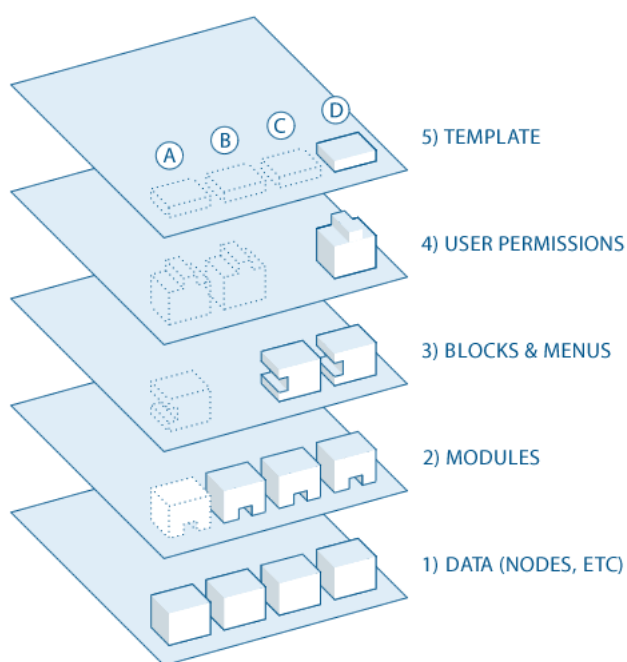
Hlavní předností Joomla je její flexibilita. Jde o robustní systém, který je možné upravit, aby provozoval jakýkoliv typ webu. Joomla též poskytuje interně editor textu a obrázků pro snadné základní úpravy. Rovněž se pyšní širokou nabídkou šablon a pluginů a přehledností při práci v tomto systému. Systém Joomla je však nejvíce opěvován pro svoji správu v oblasti e-commerce. Správa produktů a obsahu probíhá přehledně v jednom místě.[8, 24]

### 3.2.3.3 Drupal

Drupal je dalším z používaných open-source redakčních systémů založený na php v roce 2001. Jádro Drupalu obsahuje základní vlastnosti potřebné pro tvorbu klasického webu, víceuživatelského blogu, internetového fóra a obsahuje též nástroje pro správu obsahu. Pro

potřeby databáze podporuje MySQL, SQLite, nebo PostgreSQL. Primárními komponentami jádra Drupalu pro správu a tvorbu jsou: obsah, download/upload souborů, menu, uživatelské účty, role a práva, taxonomie, diskuzní fóra, pohledy k extrahování a zobrazení obsahu jako jsou seznamy, tabulky a WYSIWYG (akronym z anglického What You See Is What You Get, tedy to co vidíš, to dostaneš) editor obsahu.[9]

Drupal staví na základech modularity, kvality a open-source. Malé, stabilní rychlé jádro, které je možné vlastnoručně modulovat, popřípadě si modul stáhnout na oficiálních stránkách, přidává na flexibilitě systému. S dobře navrženou strukturou jde o bezpečný systém, který je zastřešen pod GNU/GPL licenci.[9, 25]



Obrázek 13: Hlavní vrstvy jádra redakčního systému Drupal[25]

### 3.2.3.4 PrestaShop

Skrze Prestashop, jak již název napovídá, se spravují pouze e-shopové stránky. Tento redakční systém využívá AJAXu, naprogramován je pomocí jazyka PHP, využívá frameworku „SMARTLY“ a podporuje MySQL 5 a vyšší verze.

Hlavní výhodou Prestashopu je fakt, že je postaven přímo pro e-shop a tak se nezatěžuje ničím jiným než funkcionalitou e-shopu. Funkce, které jiné redakční systému musí řešit pomocí pluginů má Prestashop již v sobě. Jeho funkce jsou předem připravené pro spuštění e-shopu a tak se majitel může soustředit zejména na produkty, které prodává. Díky zaměření

právě na elektronické obchodování je systém potenciálně rychlejší, jelikož neobsahuje kusy kódu předpřipravené pro jiné využití systému.[10, 26]

### 3.2.4 Zabezpečení

Zabezpečení dat, informací a webových stránek obecně se vyznačuje v těchto systémech především vybráním kvalitního poskytovatele hostingu, vybráním vhodného redakčního systému potřebného pro provozovaný business, otázkou zálohování a případné následné obnovy. V rámci zabezpečení se také ukládají logovacími soubory, provádí se aktualizace systému, pluginů (a šablon) a eventuálně lze systémy rozšířit o bezpečnostní pluginy.[11]

Hosting je často vybírán dle kritérií velikosti databáze, množství souborů, které je možné uložit, počet cron úloh a jiné. Ač jsou tato kritéria důležitá pro správnou implementaci webu, je třeba dbát i na zabezpečení jaké daný hosting nabízí svým zákazníkům. Je možné vybrat si hosting sdílený, virtuální privátní server, mít vše na vlastním serveru, anebo na cloudovém úložišti. Nejčastější volbou je sdílený hosting, kdy je na jednom serveru uloženo více webových stránek, respektive prostoru pro ně. Důležité je se zaměřit na to, aby byl server zabezpečen jak fyzicky tak i kyberneticky. S výběrem hostingu by se měla pojít i otázka zálohování, nejčastěji databázové části systému, ale i souborů na serveru.[11]

Výběr vhodného redakčního systému se odvíjí od využití daných stránek. Ne každý informační systém je vhodné zvolit pro e-shop, nebo své portfolio. Rozhodujícími faktory jsou podpora systému, jeho životaschopnost, funkcionality a bezpečnost. Logovací soubory v rámci zabezpečení slouží pro kontrolu správcem systému. Obsahují rozličné informace o operacích, přístupech a dalších situacích spojených s redakčním systémem.[11]

Mnoho bezpečnostních děr se snaží sami autoři redakčních systémů záplatovat v podobě aktualizací těchto systémů. Proto je třeba udržovat nainstalovaný redakční systém aktuální. Další možností zabezpečení systému jsou pluginy k tomu určené. Bezpečnostních pluginů je na trhu velké množství jak v placené verzi, tak i ve verzi zdarma. Bezpečnostní pluginy přinášejí v základu ochrany jako je brána firewall a vnitřní kontroly, jenž jsou typicky podobné chováním antivirům při skenování počítače.[11]

Častým narušením zabezpečení bývá neodborná manipulace uživatele se systémem. Proto je třeba člověka, který se systémem pracuje řádně proškolit.[11]

### 3.2.5 Šablony a pluginy

Šablony a pluginy patří ke každému redakčnímu systému. Šablony reprezentují strukturu stránek, kterou lze dále upravovat respektive vzít rovnou k použití. Pluginy v redakčních systémech pak přidávají rozšířené funkcionality webu. V redakčních systémech se často vyskytuje pojem Theme, tedy téma. Téma se skládá z šablon stránek a občas vyžaduje stažení určitých pluginů pro správnou funkcionality celku. [7, 8, 9]

#### 3.2.5.1 Šablona

Všechny zmíněné redakční šablony mají širokou škálu šablon zdarma, ale i placených. Šablony jsou rozděleny dle kategorií použití. Některé jsou připraveny pro e-shopová řešení, některá pro blogy jiná pro portfolia atd. Každá šablona, nebo soubor šablon určuje vzhled webu, který je možno upravit pomocí kaskádových stylů, v určitých případech i zásahem do samotné šablony. Šablona samotná se skládá z klasických HTML prvků s danou strukturou. Tedy má hlavičku, tělo a zápatí. Princip šablon v redakčních systémech je poněkud odlišný od standardní šablony pro jednu webovou stránku. Šablona v redakčním systému zpravidla představuje několik souborů, využívajících například php a funkcí, které jádro CMS nabízí. Programátorskými principy je pak stránka složena z těchto souborů a tvoří výslednou stránku. Tento princip funguje univerzálně na všechny stránky v rámci redakčního systému, není-li naprogramováno jinak.[3, 4]

#### 3.2.5.2 Plugin

Plugin je rozšiřující funkcionality. Zpravidla přidává nějaký mechanismus či novou funkci, jako je například rozšíření redakčního systému o e-shop. Jde o soubory psané v javascriptu, nebo php, které někdo naprogramoval. Občas jsou rozšířeny o HTML, nebo kaskádové styly, pokud rozšiřují i stránky samotné. Pluginy nejčastěji bývají zdarma ke stažení, nicméně jejich plná funkcionality bývá často zpoplatněna.[3, 4]



### 3.3 Vícekriteriální rozhodování

Vícekriteriální rozhodování vychází z teorie rozhodování, což je model konfliktní situace, při kterém se volí optimální rozhodnutí ovlivňované budoucím stavem. Do vícekriteriálního rozhodování patří zejména pojmy varianta a kritérium. Varianta neboli alternativa je jedno z možných řešení, jenž je zvoleno na základě metody dle daných kritérií a jejich ohodnocení. Kritériem je pak faktor, dle kterého danou variantu vybíráme. Vícekriteriální rozhodování se dělí na dva modely dle toho, zdali je počet variant konečný či nikoliv. V případě výběru open-source redakčního systému jde o konečnou množinu variant a pro tyto účely slouží metoda vícekriteriální analýzy variant.[27]

Cílem vícekriteriální analýzy variant je najít variantu, která je kompromisem respektive ideální variantou dle daných kritérií. Informace může být ordinální, nominální či kardinální. Ordinální informace je taková informace, u níž jsme schopni určit pořadí. Kardinální informace je numericky vyjádřena. U nominální informace lze říci pouze to, zda je stejná nebo odlišná od jiné informace. Kvantifikaci informace lze provést metodou pořadí, bodovací metodou, nebo metodou párového porovnávání.[27, 28]

Řešením vícekriteriální analýzy je varianta zvolená na základě kompromisu kritérií. Setkat se lze ale i s variantou ideální a bazální. Ideální varianta představuje situaci, kdy je varianta ohodnocena nejlepšími hodnotami ve všech kritériích. Opakem k tomu je bazální varianta, která je ve všech hodnoceních kritérií nejhorší. Kompromisní řešení má k ideální variantě nejmenší vzdálenost, dle zvolené metody. Při analýze vhodné varianty je možné uvést pro každé kritérium jeho váhu, což rozhodování ovlivní. Vícekriteriální analýzy variant bylo využito pro zvolení vhodného redakčního systému k realizaci. [27]

### 3.4 Účetní software

Účetní software lze definovat jako organizovanou skupinu manuálních a výpočetních účetnických metod, postupů a kontrol určených ke sběru, evidování, klasifikaci, analýze, shrnutí, interpretaci a prezentaci přesných a včasných finančních dat pro manažerské rozhodnutí. V dnešních účetních softwarech se řeší výdaje a příjmy, fakturace a další finanční aktivity podniku. Takovýto software umožňuje podnikateli sledovat veškeré finanční transakce včetně výdajů a příjmů a tedy prodeje a je schopný z těchto dat produkovat obsáhlé statistiky. Historicky šlo o komplexní sérii manuálních a výpočetních metod, které jsou dnes automatizovány.[29, 30]

Historicky první zmínka o účetním záznamu je zhruba 7 tisíc let stará. První účetní stroj byl pak vyvinut v letech 1880 a v dvacátém století s příchodem počítačových technologií se začali účetní softwary automatizovat a to až do dnešních podob. Aktuálně zdarma dostupných softwarů vhodných pro potřeby malých, začínajících či start-up projektů je na trhu hned několik. Mezi hlavní hráče na českém trhu patří podnikatelský software PROFIT, program MONEY S3, ekonomický software FlexiBee a ekonomický a účetní systém Pohoda. Dalšími produkty z oblasti účetních softwarů jsou například ekonomický software Stereo, MPR-K/S, software AceMoney, nebo WinDUO. Většina těchto softwarů cílí na začínající podnikatele a firmy, které poté co se rozrostou, využijí placených programů s rozšířenou funkcionalitou.[30, 31]

#### **PROFIT**

Tento systém umožňuje živnostníkům a všem, kteří potřebují podat daňové přiznání, vedení evidence neomezeného počtu firem v různých světových jazycích. Hlavními moduly jsou adresář, objednávky, fakturace, daňová evidence, peníze (pokladny a účty), majetek, sklady, cesty, reporty a grafy. Systém umožňuje vedení evidence faktur, objednávek či paragonů, tisk dobropisů, dodacích listů či pokladních a skladových dokladů. PROFIT je možno provozovat v lokálním i síťovém prostředí. K dispozici je i placená verze s rozšířenými funkcemi.[31, 32]

#### **MONEY S3**

Program MONEY S3 od společnosti CÍGLER SOFTWARE pro živnostníky, drobné a střední firmy nabízí zpracování podvojného účetnictví a daňové evidence spolu s funkcí pro fakturaci a evidenci skladu a objednávek. V neplacené verzi programu jsou dostupné veškeré funkce, nicméně po překročení limitů pro záznamy v účetním deníku, skladových pohybu apod. je systém bez zakoupení komerční licence nepoužitelný.[31]

**FlexiBee**

Software společnost Abra Software S.R.O. se nabízí ve verzi One, která je dostupná pro jednoho uživatele s ročními omezeními na počet vložených záznamů do libovolného seznamu, či počtem zpracovávaných firem, který lze registrací zdarma navýšit. Tento program by měl do začátku podnikání být dostačující. Software FlexiBee je vhodný pro živnostníky vedoucí daňovou evidenci, tak i firmy vedoucí podvojně účetnictví. Systém umožňuje správu obchodních partnerů, prodejů, nákupů, zboží, pokladních a bankovních dokladů. Program dokáže zpracovat pohledávky, závazky, poptávky, nebo nabídky a objednávky a tvořit účetní výstupy. K dispozici je řešení pomocí cloudových služeb s vystavenými webovými službami umožňující napojení účetního softwaru.[31, 33]

**POHODA**

Obdobně jako konkurence ve verzi start je možné program pohoda využívat do vyčerpání povolených limitů a poté program zakoupit v některé z komerčních verzí. Dalším omezením je práce s daty, která je omezena na 20 úkonů typu mazání, datová komunikace, homebanking a další funkce. Komplexní účetní systém pohoda je založen na propracovaném adresáři, který je doplněn různými moduly pro zpracování faktur, správu majetku, skladů apod. a umožňuje daňovou evidenci živnostníkům i účetnictví firmám.[31]

## 4 Praktická část

### 4.1 Výběr CMS

Pro potřeby realizace stránky s e-shopem je třeba vybrat vhodný redakční systém. Toho bylo dosaženo aplikací metody vícekriteriální analýzy variant. Pro výběr byla stanovena kritéria zohledňující tvorbu e-shopu a rovněž tvorbu prezentačního webu produktu spolu s prezentací firmy. Vzhledem k tomu vyvstává nutnost realizace prezentační stránky, kde bude moci zákazník ověřit solidnost firmy a dozvědět se o ní více informací. Spolu s prezentací firmy bude třeba připravit i e-shop. S ohledem na tyto požadavky byla stanovena následující kritéria.

#### 4.1.1 Kritéria

**Aktivita vývojářů a její trendy:** Toto kritérium pokrývá oblast aktuálnosti redakčního systému. Aktuální redakční systém je základem pro bezpečnost, ale i pro prevenci zastarání či získání rozšířených funkcionalit. Trend vývoje redakčního systému pak určuje určitou míru vhodnosti daného systému k určitému užití.

V rámci ohodnocení toho kritéria byla zohledněna zejména pravidelnost updatu systému, časový odstup dodání nových verzí, přínosy nových verzí oproti verzím původním a směr, kterým se tyto systémy ubírají.

**Bezpečnost:** Garance bezpečnosti je stále aktuálnějším tématem. Spolu se stále rostoucí oblastí IT je nutné systémy chránit před stále novými hrozbami. V oblasti webových stránek jde o různé webové útoky, v případě e-shopů dokonce o únik citlivých dat o zákaznících, firmě a dalších.

V rámci hodnocení tohoto kritéria byly bodovány oblasti podpory schvalování obsahu, verifikace emailů, podpora captcha obrázků, oblast přidělování práv a autentizace, záznamy a verzování a zabezpečení pomocí SSL.

**E-commerce:** E-shop spadá do oblasti elektronického obchodování a proto samotné elektronické obchodování je jedním ze zásadních kritérií výběru vhodného CMS. Některé open-source systémy se soustředí pouze na tvorbu e-shopu, jiné tuto funkcionalitu umožňují pomocí pluginů. Reálná je i možnost využití čistě e-shopového CMS spolu s přidáním statické

stránky. Použití více systémů najednou představuje náročnou situaci v ohledu implementace, konsolidace a výkonu.

Pro ohodnocení tohoto kritéria bylo zjišťováno, zda daný redakční nabízí správu zásob, platby v e-shopu, různé typy dopravy zboží, zdanění, nákupní košík, odběry novin, seznamy přání, crosselling a upselling, správu produktů a evidenci objednávek a faktur.

**Jednoduchost užití:** Intuitivní používání redakčního systému a jednoduchost jeho použití představuje klíč ke snadné správě obsahu, do velké míry i ke správě webu jako takového. Oblast zabývající se mimo jiné i jednoduchostí užití představuje user experience. Určitým pravidlem je, že silnější systémy, umožňující více funkcionalit a robustnější řešení, bývají méně snadné pro užití a správu.

Jednoduchost užití byla vyhodnocena na základě možností tvorby obsahu stylem drag'n drop, možnosti reakce na webu prostřednictvím emailu, zobrazení tzv. přátelských URL, možnosti upravovat obrázky v systému, využívání maker, možnosti většího množství uploadovaných souborů najednou, možnosti prototypování a nastavení jazyků. Dále také bylo vyhodnocováno, zda lze stránku a její stylování nastavit pomocí instalačního procesu, rovněž zda stránka pomáhá se správností slov, zda umožňuje odběr novin ze stránky a možnost zpět, jako je tomu např. u grafických editorů, zdali se jedná o WYSIWYG editor a zdali je možno potřebné soubory archivovat např. do formátu zip.

**Moduly zdarma:** Většina rozšiřujících funkcionalit v oblasti systémů pro správu obsahu je přidávána pomocí pluginů. Vzhledem k open sourcovému řešení této práce zohledňuje toto kritérium šíři nabídky pluginů zdarma vzhledem k pluginům placeným a rovněž vzhledem k jejich dostupnému množství napříč redakčními systémy.

Bodové ohodnocení se odvíjelo od počtu pluginů, které jsou k dispozici zdarma vzhledem k celkovému počtu pro daný systém, rovněž bylo vzato v úvahu jejich reálné použití.

**Podpora CMS:** Podpora redakčního systému stejně tak jako i dalších softwaru je důležitou součástí vývoje stránek a e-shopu na dané platformě. Mezi podporu systému se řadí podpora nejen vývojářů, ale rovněž také podpora komunity.

Vyhodnocení kritéria se odvíjelo od dostupnosti manuálů systému, možnosti tréninkových a výukových programů a dostupnosti těchto programů. Rovněž se také odvíjelo od dostupnosti certifikací, koster kódu, online pomoci, veřejných fór a testovacích frameworků. Do

hodnocení rovněž vstoupila otázka vývojářské a uživatelské komunity a vývoj pluginů třetími stranami. Dále také možnosti profesionálního hostingu a služeb přímo od výrobce daného systému.

**Šablony zdarma:** Tvorba webu v oblasti jeho vizuální stránky zabírá mnoho času a může vyžadovat poměrně velké finanční prostředky. Tyto investice lze zmírnit, respektive minimalizovat a to výběrem vhodné šablony, která vzorem a rozmístěním prvků odpovídá představě zákazníka. Vizuální stránka se tak smršťuje na úpravu detailů například pomocí CSS a nikoliv na výstavbu celého webu. V rámci open-source tvorby je vhodné využít nějakou z dostupných šablon, jenž jsou k dispozici zdarma.

Ohodnocení kritéria se odvíjí od množství šablon, které jsou dostupné zdarma v porovnání se všemi dostupnými šablonami pro daný systém. Výsledné ohodnocení je ovlivněno i vhodností šablon pro tvorbu e-shopu a jejich dostupný počet.

**Výkon:** Vzhledem k narůstajícímu objemu dat na internetu, je otázka výkonu důležitá. Oblast webových stránek není výjimkou, a tak je důležité, aby se webová stránka nenačítala například příliš dlouho, jelikož tak může ztratit potenciální zákazníky a návštěvníky webu. Řešení e-shopu ještě zvyšuje požadavky kladené na výkon systému, jelikož e-shop produkuje mnoho dalších funkcionalit, které musejí být realizovány a uživatelé by neměli nikterak omezit. Pro menší e-shop je hledisko výkonu relativně méně významné, nicméně u velkoobjemových a plně automatizovaných e-shopů je kritérium výkonu zásadní. S pojmem výkon samozřejmě úzce souvisí pojem optimalizace.

V rámci hodnocení tohoto kritéria byl základní výkon systému a dále rozšířené možnosti. Jednalo se o možnosti využití mezipaměti, databázové replikace, rozvržení zátěží při načítání stránek, možnosti exportu a další optimalizace.

#### 4.1.2 Stanovení vah

Ke stanovení vah bylo využito Saatyho matice, ve které se volí preference jednoho kritéria nad druhým, čímž vznikne matice, ze které jsou preference kritéria nad ostatním jasně vidět. Aplikací vzorce  $\sqrt[n]{\prod s_{ij}}$  se vypočítá hodnota  $b_i$  pro každý řádek reprezentující jedno kritérium. Hodnota  $b_i$  je dosazena do vzorce pro výpočet jednotlivých vah kritérií. Váha kritéria  $v_i$  se vypočítá pomocí vzorce  $\frac{b_i}{\sum b_i}$ . Kontrolní součet vah kritérií se pak musí rovnat jedné. Následující tabulka obsahuje preferenci kritérií a výpočet finálních vah kritérií.

Saatyho matice	Aktivita vývojářů a její trendy	E-commerce	Bezpečnost	Podpora CMS	Moduly zdarma	Šablony zdarma	Výkon	Jednoduchost užití	bi	Výsledné váhy (vi)
Aktivita vývojářů a její trendy	1	1/5	1	1	1/5	1/5	1/3	1	0,476701031	0,04415588
E-commerce	5	1	5	7	5	5	5	5	4,264467002	0,39500919
Bezpečnost	1	1/5	1	3	2	2	1/5	3	1,046635139	0,09694775
Podpora CMS	1	1/7	1/3	1	1	1	1/3	1	0,595775214	0,05518549
Moduly zdarma	5	1/5	1/2	1	1	3	1/3	3	1,051989506	0,09744372
Šablony zdarma	5	1/5	1/2	1	1/3	1	1/3	3	0,799339167	0,07404121
Výkon	3	1/5	5	3	3	3	1	5	2,118028882	0,19618885
Jednoduchost užití	1	1/5	1/3	1	1/3	1/3	1/5	1	0,44293197	0,04102792
									10,79586791	1

Tabulka 3: Saatyho matice Zdroj: Vlastní tvorba

#### 4.1.3 Varianty a jejich ohodnocení

Jako varianty respektive dostupné alternativy byly zvoleny nejznámější dostupné open-source redakční systémy využívané v ČR a zahraničí. Jde o redakční systémy Wordpress, Joomla, Drupal a PrestaShop. Každá varianta byla bodově ohodnocena v rámci stupnice od 0 do 100 bodů, kde sto bodů reprezentuje maximum v daném kritériu. V tabulce 4 jsou jednotlivá ohodnocení uvedena a rovněž jsou zde uvedeny výsledné body po aplikaci vah jednotlivých kritérií. Na základě těchto hodnot bylo zvoleno vítězné řešení, respektive pořadí variant.

V jednotlivých kritériích byli různé systémy lepší než ostatní, žádný ze systému však nebyl nejlepší ve všech kritériích. V oblasti aktivity vývojářů a trendů vývoje byli nejlépe ohodnoceni vývojáři systému PrestaShop, který je rovněž nejlépe hodnocen v oblasti e-commerce, jelikož se přímo na tuto oblast zaměřuje narozdíl od svých konkurentů. V základních kritériích bezpečnosti bez přídatných pluginů byl ohodnocen nejlépe systém Drupal. Oblast podpory CMS byla u všech systému velmi dobrá. Nejvíce moduly a šablonami zdarma disponoval redakční systém Wordpress. V oblasti výkon byl nejlepší opět systém Drupal, přičemž ostatní systémy byly výkonově ohodnoceny rovnocenně. V jednoduchosti použití byl nejlépe ohodnocen systém Wordpress, který se obecně pohybuje v této oblasti v lepším průměru.

	Aktivita vývojářů a její trendy	E-commerce	Bezpečnost	Podpora CMS	Moduly zdarma	Šablony zdarma	Výkon	Jednoduchost užití	Ohodnocení s váhamy ze saatyho
Wordpress	60	75	65	94	93	86	60	66	73,67
Joomla	70	82	64	91	64	30	60	63	69,52
Drupal	40	70	71	93	82	50	80	62	71,36
PrestaShop	80	98	65	87	29	25	60	51	71,89

Tabulka 4: Vyhodnocení variant Zdroj: Vlastní tvorba

Jako první se umístila varianta redakčního systému Wordpress, následována systémem PrestaShop a Drupal a jako poslední se umístil systém Joomla. Na základě této analýzy bylo vy-

hodnoceno řešení systémem Wordpress jako nejvhodnější pro realizaci portálu, který počítá jak s prezentační stránkou webu, tak i s přímým prodejem prostřednictvím tvorby malého e-shopu pro jejich stránky.

## 4.2 Realizace

Realizace e-shopu a webových stránek vyžaduje prostor, kde budou stránky uschovány. Proto je potřeba pečlivě vyhodnotit poskytovatele hostingu stránek, tak aby byl schopen zajistit hladký chod stránek, a aby poskytoval vše co je k tomuto chodu potřeba. Pro různé stránky můžou být vhodné různé poskytovatelé. Pro případ realizace malého e-shopu na open source bázi je potřeba zajistit přístup do databáze, úložný prostor a hlavně podporu redakčního systému. Redakční systémy jsou dnes již běžnou věcí a tak je většina poskytovatelů podporuje, nicméně je vhodné si podporu CMS u poskytovatele ověřit.

K realizaci tohoto řešení byl použit freehosting od společnosti Endora. Varianta hostingu zdarma poskytuje 2 GB místa pro web, 30GB přenosu dat měsíčně a umožňuje přesměrovat emaily na vlastníkův email. Do webu na tomto hostingu je vložena reklama a poskytovatel vyžaduje pravidelnou aktivitu na webu. Hosting společnosti Endora podporuje všechny v analýze zmíněné redakční systémy.

### 4.2.1 Instalace redakčního systému Wordpress

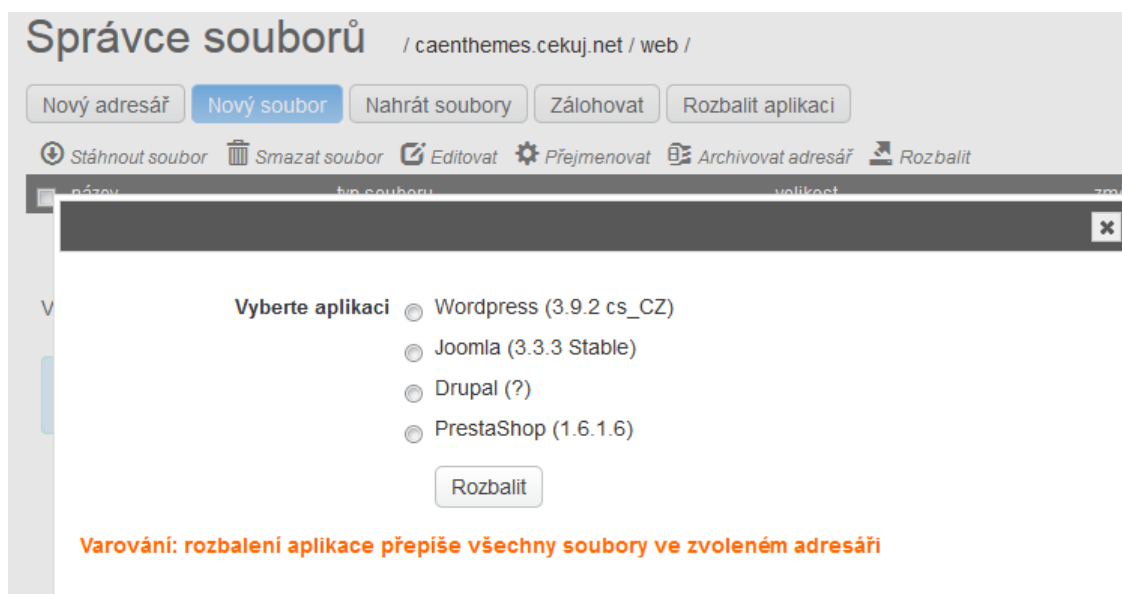
Jako první byla založena jedna společná databáze pro redakční systém a e-shop. Redakční systém si s vytvořenou databází nadále pracuje sám, vytváří si potřebné struktury a tabulky a ty plní daty automaticky. K databázi byl zřízen i přístupový účet s náhodně vygenerovaným heslem. Všechny tyto kroky byly provedeny v administrační části na webu poskytovatele hostingu. Vytvořena byla databáze s názvem eshopcaenthemesdb a s uživatelem eshopd-buser.





Obrázek 14: Vytvořená databáze a přístup do ni Zdroj: Vlastní tvorba

Dalším krokem je instalace samotného redakčního systému. Prvotní nahrání systému je u poskytovatele Endora možné provést dvěma způsoby. Je možné si potřebný redakční systém stáhnout a pomocí správce souborů či přes FTP, daný redakční systém nakopírovat do úložiště na serveru a tam daný redakční systém rozbít, jelikož je většinou stažen jako archiv ve formátu ZIP nebo RAR. Tímto způsobem je možné nahrát nejnovější verzi redakčního systému přímo na web. Druhou možností jak redakční systém u poskytovatele Endora provoznit, je možnost rozbalení aplikace přímo ve správci souborů na webu.



Obrázek 15: Rozbalení CMS na hostingu Endora Zdroj: Vlastní tvorba

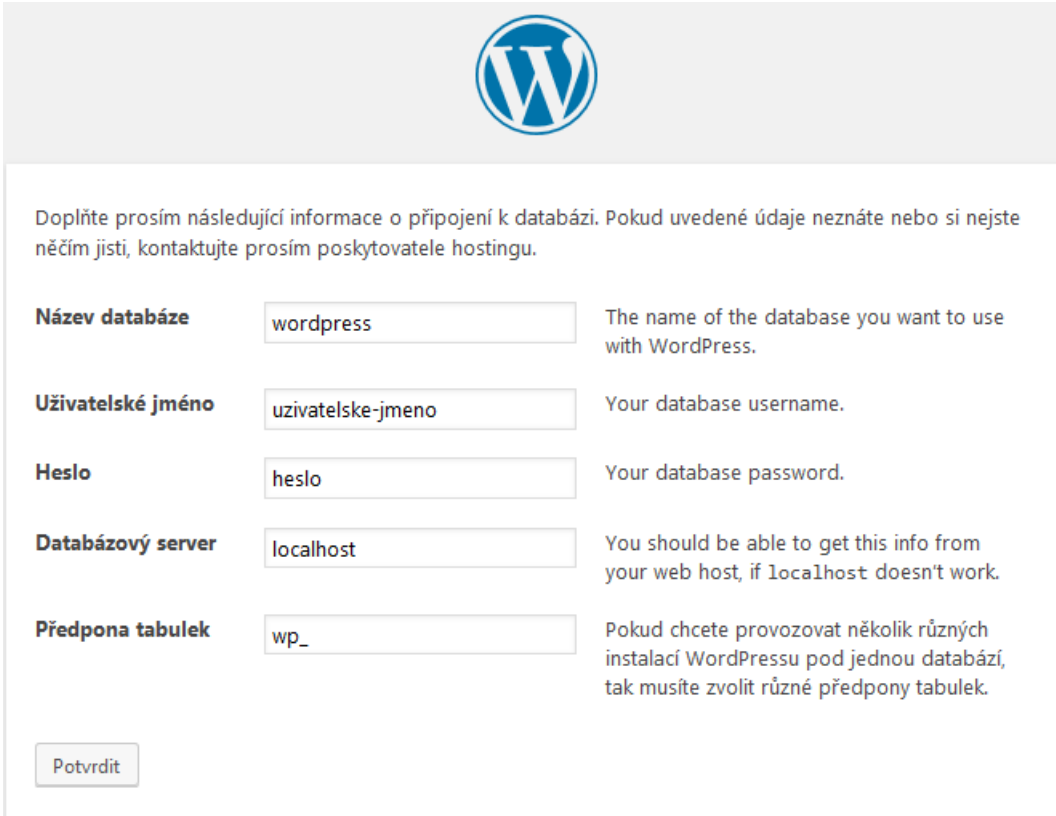
Drobnou nevýhodou druhé možnosti je neaktuálnost instalovaného systému. Uživatel vybere aplikaci, kterou chce rozbít, viz předchozí obrázek, ale tato aplikace je v zastaralé verzi. Proto je nutné po prvotním nastavení redakčního systému provést aktualizaci. Využitím druhé varianty ale naopak odpadá nutnost stahovat redakční systém ze stránek výrobce

a přenášet ho manuálně k poskytovateli. První varianta vyžaduje více práce a pozornosti uživatele oproti variantě druhé, kde uživatel pouze kliká na určená místa.

Poté co je systém rozbalen, ať již první nebo druhou metodou, je třeba redakční systém nakonfigurovat. Při čisté instalaci CMS Wordpress se prvotní konfigurace spustí automaticky otevřením domény v prohlížeči.

#### 4.2.2 Prvotní konfigurace redakčního systému

Jako první se od uživatele vyžaduje volba jazyka. Zvolený jazyk se prolíná nejen skrz průvodce konfigurací, ale i do celé administrační části systému Wordpress. Jazyk je však možné v nastavení systému kdykoliv změnit. Po zvolení jazyka přichází obrazovka s informacemi, kterou budou potřeba pro korektní nastavení. Jde o název databáze a uživatelské jméno pro přihlášení do ní a rovněž heslo k tomuto jménu a databázovému serveru. Tyto informace jsou dostupné u poskytovatele hostingu a bez nich není možné konfiguraci systému dokončit úspěšně.



Doplňte prosím následující informace o připojení k databázi. Pokud uvedené údaje neznáte nebo si nejste něčím jisti, kontaktujte prosím poskytovatele hostingu.

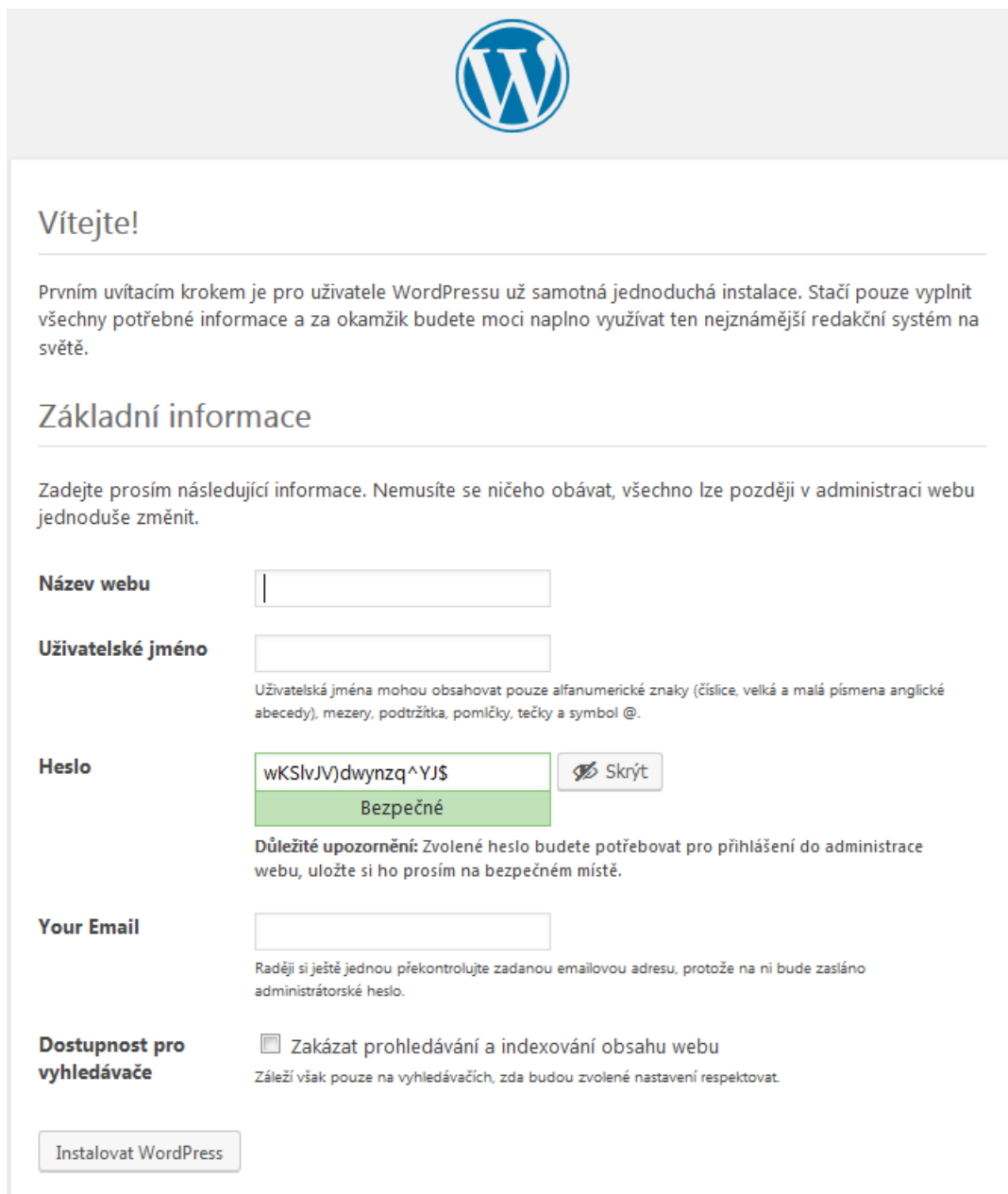
<b>Název databáze</b>	<input type="text" value="wordpress"/>	The name of the database you want to use with WordPress.
<b>Uživatelské jméno</b>	<input type="text" value="uzivatelske-jmeno"/>	Your database username.
<b>Heslo</b>	<input type="text" value="heslo"/>	Your database password.
<b>Databázový server</b>	<input type="text" value="localhost"/>	You should be able to get this info from your web host, if localhost doesn't work.
<b>Předpona tabulek</b>	<input type="text" value="wp_"/>	Pokud chcete provozovat několik různých instalací WordPressu pod jednou databází, tak musíte zvolit různé předpony tabulek.

Obrázek 16: Konfigurace CMS WP Zdroj: Vlastní tvorba

Název databáze je třeba vyplnit dle předem vytvořené databáze a přístup do ní odpovídá

taktéž předem vytvořenému uživateli. Do položky název databáze bylo vyplněno eshopca-enthesdb, do pole uživatelské jméno hodnota eshopdbuser a do položky heslo uživatelem zvolené heslo. Pole databázový server bylo ponecháno s hodnotou localhost, jelikož aplikace se nepřipojuje nikam vzdáleně, nýbrž na místo na serveru, kde je uložena. To jak si přenosy a fyzické uložení dat řeší poskytovatelé, není z hlediska této práce podstatné. Předpona tabulek může a nemusí být prázdná. Nedoporučuje se ponechávat předponu na původní hodnotě wp\_ a to z důvodů prevence před kybernetickými útoky. Struktura databáze těchto systémů je totiž obecně známá a útočník by při případném prolomení přístupu, měl práci s tabulkami ulehčenou. Do pole předpona tabulek tak byla vyplněna hodnota eshop\_. Ta je sice odvoditelná vzhledem k cílům této práce, nicméně jde o demonstrativní hodnotu.

Bylo-li vše správně vyplněno a systému se podařilo propojit se s databází, pak se uživateli zobrazila informativní hláška o tom, že vše proběhlo v pořádku a uživatel dále pokračoval stisknutím tlačítka instalovat. Následovala obrazovka, kde bylo nutné vyplnit základní údaje o webu a nastavit přístupové údaje pro vstup do administrace CMS.



**Vítejte!**

Prvním uvítacím krokem je pro uživatele WordPressu už samotná jednoduchá instalace. Stačí pouze vyplnit všechny potřebné informace a za okamžik budete moci naplno využívat ten nejnámější redakční systém na světě.

### Základní informace

Zadejte prosím následující informace. Nemusíte se ničeho obávat, všechno lze později v administraci webu jednoduše změnit.

**Název webu**

**Uživatelské jméno**

Uživatelská jména mohou obsahovat pouze alfanumerické znaky (číslice, velká a malá písmena anglické abecedy), mezeru, podtržítka, pomlčky, tečky a symbol @.

**Heslo**

**Bezpečné**

**Důležité upozornění:** Zvolené heslo budete potřebovat pro přihlášení do administrace webu, uložte si ho prosím na bezpečném místě.

**Your Email**

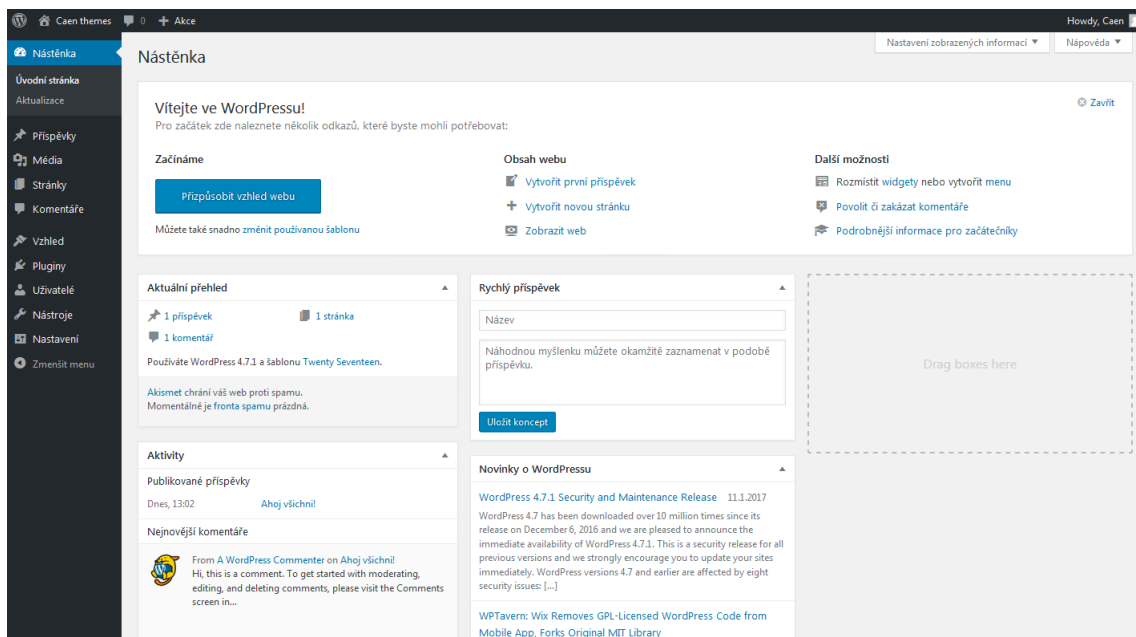
Raději si ještě jednou překontrolujte zadanou emailovou adresu, protože na ni bude zasláno administrátorské heslo.

**Dostupnost pro vyhledávače**  Zakázat prohledávání a indexování obsahu webu

Záleží však pouze na vyhledávačích, zda budou zvolené nastavení respektovat.

Obrázek 17: Nastavení přístupu do administrace Zdroj: Vlastní tvorba

V rámci nastavení je uživateli předem vygenerováno přístupové heslo. Vyžadováno je vyplnění uživatelského jména, názvu webu a emailové adresy kam budou údaje zaslány. Tato emailová adresa funguje i pro zaslání nového hesla v případě jeho ztráty. Vygenerované heslo nemusí být ponecháno a je možno ho změnit. Checkbox pro dostupnost pro vyhledávač je, jak sami autoři uvádějí, ve své podstatě nedůležité, jelikož prohlížeč nemusí brát toto nastavení vůbec v úvahu. Stisknutím tlačítka „Instalovat WordPress“ je prvotní nastavení dokončeno a pokud vše proběhlo korektně, je uživatel přeměrován do webové administrace redakčního systému s přihlašovacími údaji, které na obrazovce v nastavení vyplnil.



Obrázek 18: Administrace redakčního systému WordPress Zdroj: Vlastní tvorba

### 4.2.3 Volba šablony

Po rozbalení systému Wordpress verze 4.7.1 je předinstalovaná pouze šablona twenty-seventeen, twenty-sixteen a twenty-fifteen. Tyto šablony byly vytvořeny přímo vývojáři CMS, nicméně nejsou pro realizaci e-shopu přímo vhodné. Správnou šablonu pro tvorbu e-shopu lze nalézt na různých internetových stránkách, nebo na stránkách wordpress.org. Na těchto stránkách existuje přehledný výběr šablon a pluginů, které lze filtrovat podle daných kritérií. Pro výběr šablony bylo jako filtrační kritérium zvoleno využití pro e-shop, mřížkové rozložení a podpora lokalizace stránky. Výsledkem bylo 72 šablon, ze kterých byla vybrána šablona Storefront. Hlavní výhodou této šablony je to, že byla vyvinuta přímo tvůrci pluginu WooCommerce. Šablona je kvalitně zpracovaná, plně responzivní a je jí možno okamžitě použít, nebo i dopravit. Největší předností je však plná integrace s již zmíněným pluginem. Šablonu je možné nainstalovat z administračního rozhraní systému, nebo ji manuálně nakopírovat do složky `/wp-content/themes`, kde má každá šablona svoji složku. Šablonu je poté nutné aktivovat v záložce „Vzhled“ a její odrážce „Šablony“.

### 4.2.4 Pluginy

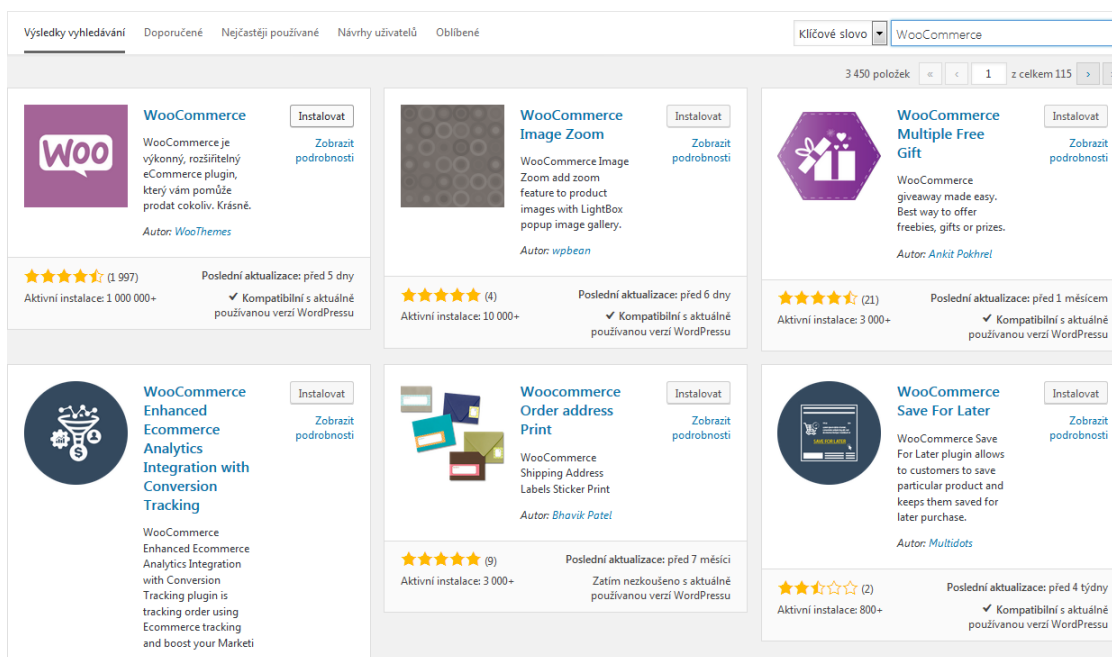
V prvotní instalaci jsou v redakčním systému nainstalovány dva pluginy. Jde o plugin Akismet, který se využívá pro ochranu webu proti spamu. Tento plugin je vhodné aktivovat a nechat ho

aktivovaný i bez zaregistrování. Druhý předinstalovaný plugin Hello Dolly je plugin, který náhodně ukazuje text znělek, pro funkčnost webu je zcela nepodstatný a nepotřebný. Pro realizaci e-shopu byl využit plugin WooCommerce.

#### 4.2.4.1 Instalace WooCommerce

Plugin WooCommerce je bezplatným pluginem pro zabudování e-commerce do redakčního systému Wordpress. Plugin podporuje prodej nejen produktů fyzických nebo digitálních, ale i prodej služeb. Jeho součástí je i plná kontrola e-shopu skrze administrátorskou část systému Wordpress. WooCommerce podporuje online i offline platby. Existuje rovněž mnoho pluginů, které tento plugin rozšiřují o další funkcionality.

Instalace pluginu je možná přímo z administrační části systému Wordpress, nebo je možné plugin stáhnout a nakopírovat do složky `/wp-content/plugins`, kde každý plugin má svoji složku. Instalace z administrátorské části probíhá zvolením záložky pluginy, poté vybráním položky instalace pluginů a následným vyhledáním požadovaného pluginu. Alternativně lze ještě v této fázi zmáčknout tlačítko nahrát plugin a přiložit zarchivovaný plugin ve formátu zip. Po vyhledání dle klíčového slova WooCommerce se zobrazí relevantní pluginy.



Obrázek 19: Vyhledání WooCommerce pluginu Zdroj: Vlastní tvorba

Po stisknutí tlačítka instalovat u pluginu WooCommerce dojde k jeho instalaci. Poté co je instalace dokončena, je na stejném tlačítku uveden text aktivovat. Opětovným stisknutím tlačítka se plugin aktivuje. Alternativně je možno plugin aktivovat v záložce Pluginy. Po

aktivaci pluginu dojde k přesměrování uživatele na uvítací obrazovku, kde je vyzván k základnímu nastavení e-shopu v rámci průvodce. Tato konfigurace je volitelná a e-shop lze nakonfigurovat kdykoliv později. Stisknutím tlačítka „Jdeme na to“ začíná průvodce nastavení e-shopem. Prvním krokem uživatel stvrzuje vytvoření stránek obchodu, košík, pokladna a můj účet, které plugin pro správnou funkci e-shopu vyžaduje. Následuje nastavení lokalizace.

**Woo** **COMMERCE**

Nastavení stránky   Jazyk obchodu   Doprava & Daň   Platby   Připraven!

### Nastavení lokalizace obchodu

Kde se váš obchod nachází?

Jakou měnu bude používat váš obchod?   
*Pokud není vaše měna v seznamu, můžete ji [přidat později](#).*

Měnové pozice

Oddělovač tisíců

Oddělovač desetinných míst

Počet desetinných míst

Jaká jednotka má být použita pro váhu produktu?

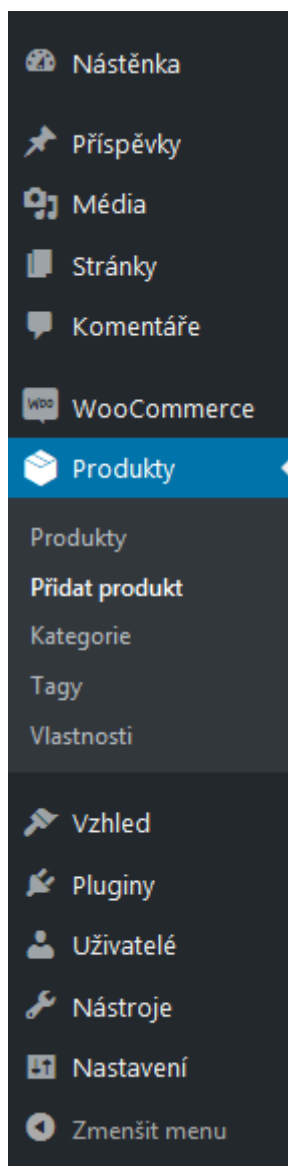
Jaká jednotka má být použita pro rozměry produktu?

Obrázek 20: Konfigurace lokalizace pluginu WooCommerce Zdroj: Vlastní tvorba

Nastavena byla česká lokalizace s uvedením měny vpravo. Jako oddělovač tisíců byla zvolena mezera a v e-shopu se nastavila práce s celými čísly. Hmotnost byla nastavena, aby byla uváděna v gramech a délka v milimetrech.

Další konfigurací byla konfigurace dopravy a daní v e-shopu. Zvolena byla varianta dodávání fyzického zboží zákazníkům a placení daně z prodeje s rozšiřujícím nastavením o uvádění cen s DPH na stránkách elektronického obchodu. Dalším nastavením je otázka plateb. Uživateli je nabídnuto zaškrtnout možnosti, jak půjde v e-shopu platit. Možnosti jsou využití služby PayPal, platby šekem, bankovním převodem, nebo hotově při doručení. Rozšiřující možností je využití platební brány Stripe, která rozšiřuje platby pomocí přídavného pluginu. Přijímaná forma plateb byla nastavena na platby hotově při doručení respektive při převzetí zakoupeného zboží. Dokončení konfigurace přesune uživatele k tvorbě prvního produktu pro e-shop. Administrační rozhraní redakčního systému bylo po instalaci a konfiguraci WooCommerce rozšířeno o záložku „WooCommerce“ a záložku „Produkty“. Záložka „Produkty“ pak obsahuje možnosti pro zobrazení všech produktů, přidání nového, možnosti úpravy kategorií a štítků produktů a nastavení vlastností produktů.





Obrázek 21: Administrační rozhraní po aktivaci WooCommerce Zdroj: Vlastní tvorba

#### 4.2.4.2 Instalace TC WooCommerce Product Slider

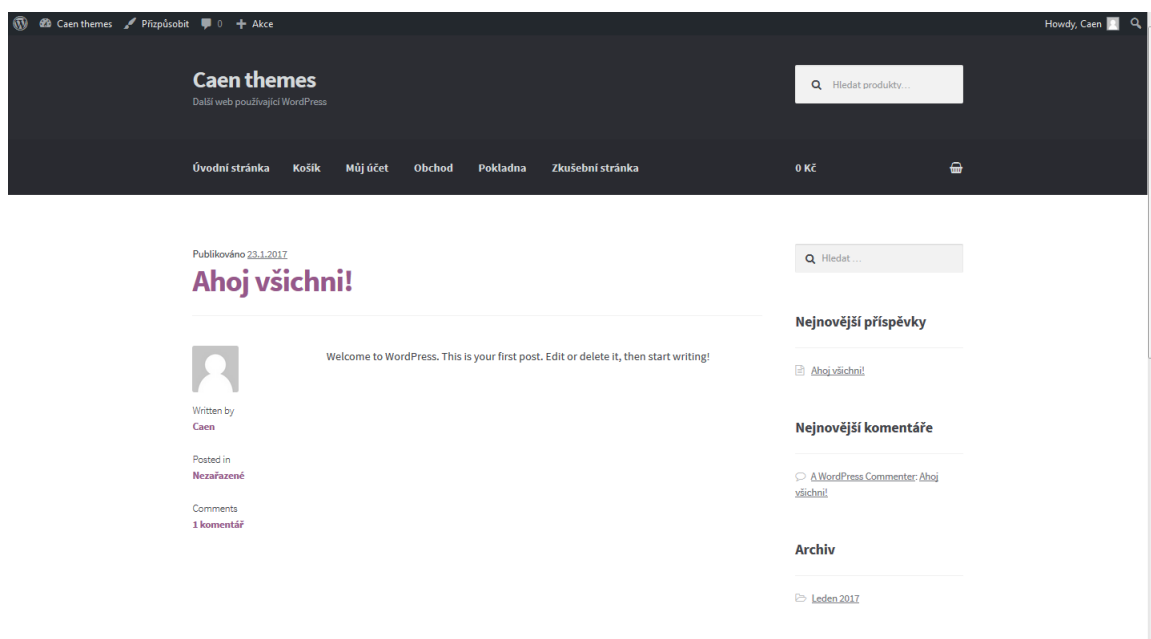
Tento plugin poskytuje možnost zobrazení produktů prodáváných v e-shopu pomocí slideru. Plugin je k dispozici zdarma a po jeho instalaci se objeví v administrační části redakčního systému záložka TC WPS. V této záložce je možné nastavit veškeré parametry tohoto pluginu, jako je navigace slideru a její zobrazení a nastavení barev, které byly sladěny do barevného designu webu. Rovněž se nastavuje různý počet zobrazovaných produktů ve slideru pro různá zařízení. Slider zobrazuje nově přidané produkty do e-shopu v místech, kde je vložen shortcode a je plně responzivní.

#### 4.2.4.3 Instalace WooCommerce Auto Restore Stock

Zdarma dostupný plugin rozšiřující logiku odebírání produktů skladem při objednání. Plugin WooCommerce při založení objednávky alokuje objednané množství kusů zboží. Při zrušení či vrácení objednávky však již kusy znovu do systému nevrátí. To zařizuje plugin WooCommerce Auto Restore Stock. Plugin tedy při změně stavu objednávky na stav zrušená, nebo vrácená, vrátí dané množství alokovaného zboží zpět do redakčního systému. Plugin funkcionalitu zajišťuje automaticky a o změnách na objednávce informuje vložením poznámky v anglickém jazyce a to k upravené objednávce.

#### 4.2.5 Základní vzhled

Po nainstalování e-shopového pluginu WooCommerce a výběru šablony Storefront redakční systém automaticky do menu vložil již vytvořené stránky a stránka respektive hlavní doména pak při prvotním zobrazení vypadala následovně.



Obrázek 22: Úvodní stránka v základním zobrazení po prvotní instalaci Zdroj: Vlastní tvorba

V rámci urychlení úprav stránek, článků a e-shopu zobrazuje redakční systém administrační lištu i na horní části obrazovky a to přihlášeným uživatelům, kteří k tomu mají oprávnění.

#### 4.2.6 Detailní nastavení eshopu

Po prvotním nastavení je možné nastavit chování e-shopu ještě rozsáhleji. WooCommerce má nastavení rozdělené do jednotlivých sekcí. Jako první byla nastavena sekce obecné. V této sekci došlo k nastavení lokality prodeje a byla nastavena pouze na prodej v rámci České a Slovenské republiky a rovněž oblast dopravy zboží byla nastavena ekvivalentně. K dispozici bylo možno i zvolení prodeje a rozvozu do všech zemí kromě správcem vyčleněných zemí. Výchozí lokalita zákazníka byla přednastavena na hodnotu Geolokace. Zbytek údajů v této záložce byl ponechán. Důležitým krokem před přechodem na jinou záložku či operaci je uložení upravených údajů pomocí tlačítka uložit změny, které je umístěno vždy na konci sekce.

V záložce produkty se nacházela již nastavená hodnota o jednotkách produktu a dále jsou zde uvedeny informace o hodnocení produktu. To je standardně nastaveno, aby umožnilo hodnotit produkt v recenzích. V této záložce je možné rovněž upravit zobrazení produktů, skladování a hodnotu o tom jak dlouho bude zboží drženo skladem od přijaté objednávky, což bylo nastaveno na 60 minut. Také lze nastavit možnosti stahování produktu, jde-li o digitální produkty. Nastavení bylo ponecháno v přednastavených hodnotách.

Výpočet daně, tedy DPH je možno počítat na základě zákaznickovy uvedené adresy. Základním nastavením pro výpočet je doručovací adresa. Oproti předvolenému nastavení bylo změněno zobrazení cen v obchodu a košíku na zobrazení ceny včetně daně. V sekci standardní daň je možné nadefinovat běžné daňové sazby pro různé státy. To lze provést manuálně nebo importem CSV souboru. Obdobně lze nastavit i snížené a nulové daňové sazby.

V rámci nastavení doručení lze konfigurovat oblasti dopravy, možnosti doručení a doručovací třídy. V oblasti dopravy je možné nastavit paušální ceny pro určitý typ dopravy. Možnosti doručení definují výchozí cíl přepravy, tedy zákaznickovu fakturační adresu a doručovací třídy umožňují seskupovat například podobné produkty pro nastavení odlišné ceny dopravy.

Záložka pokladna umožňuje nastavení slevových kuponů, povolení nákupu neregistrovaným uživatelům, přiřazení stránky s obchodními podmínkami. Proto byla v redakčním systému vytvořena jednoduchá stránka s obchodními podmínkami e-shopu. Univerzální obchodní podmínky jsou volně dostupné na internetu. Dále se nastavují potřebné údaje pro bankovní převod, jako je číslo bankovního účtu, na který je třeba zaslat finanční obnos a také je možno nastavit textaci instrukcí pro provedení platby, což by mělo zákazníkovi usnadnit provedení bankovního převodu. Jelikož v základním nastavení nebyla platba šekem zvolena jako podporovaný způsob platby, nebyla tomuto nastavení věnována další pozornost, stejně tak možnosti platby pomocí PayPal.

Nastavení stránky účet bylo změněno tak, aby uživatel mohl registraci provádět nejen na stránce pokladna, ale i na stránce můj účet v případě, že není přihlášen. Výchozím nastavením je pak generování přihlašovacího jména z emailu zákazníka, respektive registrovaného uživatele.

Sekce email obsahuje přehled emailových notifikací a jejich příjemce. Rovněž zde lze nastavit odesílatele a šablonu emailu.

#### 4.2.7 Úpravy vzhledu stránek

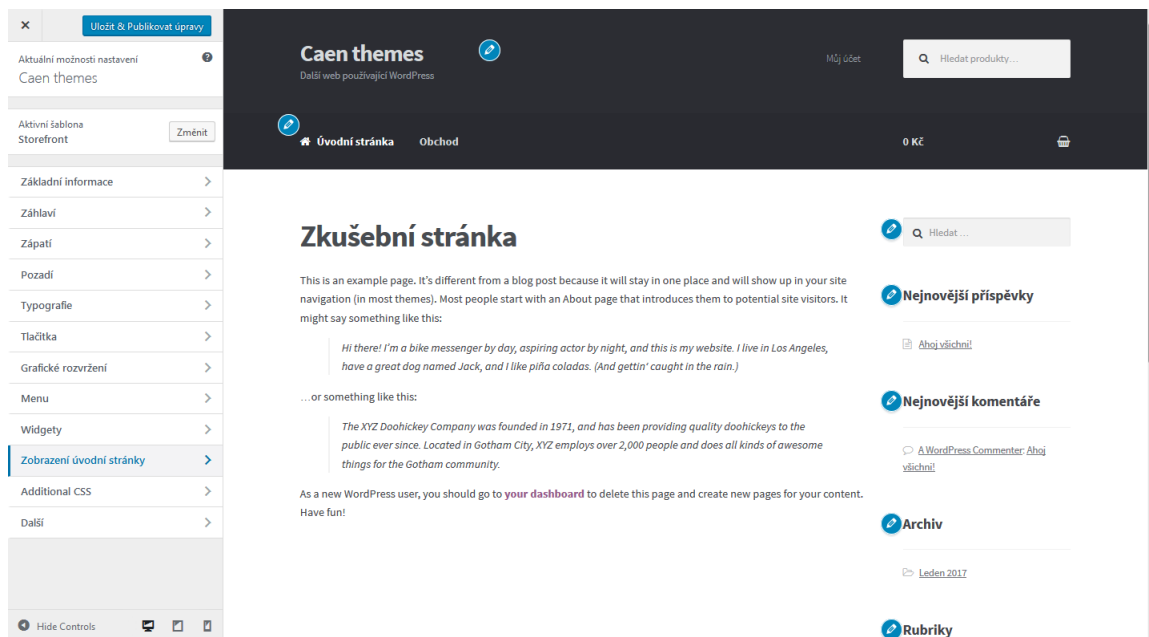
Následující kapitoly popisují detailní úpravy a tvorby dalších stránek a obecné úpravy vzhledu a struktury vytvořeného webu. Důležitým krokem před započítím zásahů do šablony je vytvoření tzv. child theme. Tedy šablony, která přebírá vlastnosti původní, ale při aktualizaci původní nedojde k přemazání vlastního obsahu. Alternativní cestou je přejmenování šablony, respektive přidání například slova upraveno za pomlčku názvu složky. Pro úpravu vzhledu je nutná úprava kaskádových stylů. V rámci zachování původní struktury, je vhodné vlastní stylování umístit do separátního souboru. Proto byl vytvořen soubor *style-custom.css*, který byl umístěn k ostatním souborům šablony storefront. Aby vlastní soubor s kaskádovými styly fungoval, bylo potřeba vložit následující kus kódu do hlavičky stránky, tedy do souboru *header.php*.

```
<link rel="stylesheet" type="text/css" href="<?php echo  
    bloginfo("template_directory");?>/style-custom.css" media=  
    "all">
```

Použitím funkce `bloginfo("template_directory")`, která je jednou z mnoha funkcí v rámci redakčního systému Wordpress, s parametrem `template_directory` bylo docíleno dosažení správného adresáře, ve kterém se šablona nacházela.

##### 4.2.7.1 Přizpůsobení šablony

Základní úpravy lze provést skrze administrační část, konkrétně v sekci vzhled zvolením možnosti přizpůsobit. Tato akce vyvolá okno pro úpravy, kde v levé části jsou možnosti úprav a vpravo náhled na úvodní stránku, kde se úpravy okamžitě projevují.



Obrázek 23: Přizpůsobení šablony Zdroj: Vlastní tvorba

V základu je šablona nastavena, aby na úvodní stránce zobrazovala nejnovější příspěvky. Pro účel stránek bylo toto nastavení změněno na zobrazení statické úvodní stránky a statické stránky pro články. Upraveno bylo i záhlaví a to v podobě změny barvy textu a nastavení banneru, který se zobrazil v pozadí hlavičky stránky. K úpravám došlo i v barvách běžných textů a zvýraznění v sekci typografie a v sekci tlačítka. V nastavení šablony Storefront je možné změnit i rozvržení a to zda se boční panel nachází vlevo anebo vpravo.

#### 4.2.7.2 Menu

Další úpravou byla tvorba vlastního menu, které bude logicky uspořádané tak, aby návštěvník viděl jen ty položky, které jsou nutné a aby byly položky v menu seřazeny dle důležitosti. Primární menu obsahuje odkaz na domovskou stránku, na stránku s obchodem, poté na stránku s články a na závěr na stránku s kontakty. Sekundární menu obsahuje odkaz na správu účtu a obchodní podmínky e-shopu. Menu lze kdykoliv upravit v sekci vzhled zvolením možnosti menu. Existuje-li více než jedno menu je třeba zvolit správné menu, na kterém se mají změny provést.

#### 4.2.7.3 Tvorba stránek

V redakčním systému wordpress lze stránky vytvářet přímo v systému a to pomocí interního editoru, nebo klasickým psaním HTML. Interní editor umožňuje rychlou tvorbu odstavců textu spolu s nadpisy. Dále umožňuje vkládání odkazů a obrázků a tvorbu seznamů.

Tento způsob funguje na WYSIWYG bázi, tedy to co se do editoru napíše, se v prohlížeči zobrazí.

### Kontakty

Stránka kontaktů byla vytvořena tak, aby obsahovala základní kontaktní informace e-shopu a kontaktní formulář. Pro kontaktní formulář byl vytvořen shortcode, což je zkrácený zápis, který se použije místo celého kódu. V šabloně je pak nutné zasáhnout do souboru *functions.php*. Kontaktní formulář je pro přehlednost vytvořen ve vlastním souboru.

```
<?php
$response = NULL;
if ( $_POST[ 'contactMessageSend' ] == 1 )
{
    if ( !empty( $_POST[ 'message_name' ] ) && !empty( $_POST[ 'message_email' ] ) && !empty( $_POST[ 'message_text' ] ) )
    {
        $subject = $_POST[ 'message_name' ] . " | " .
            $_POST[ 'message_email' ];
        $msg = $_POST[ 'message_text' ];
        mail( "eshop@eshop-domena.cz", $subject, $msg )
            ;
        $response = "V co nejkratší možné době Vás
            kontaktujeme na uvedený email.";
    }
    else
    {
        $response = "Všechna pole formuláře musí být
            vyplněna.";
    }
}
?>
<div class="contact_form">
<h2>Kontaktní formulář</h2>
<?php echo $response; ?> <br>
<form role="contact" class="form" action="<?php the_permalink
    (); ?>" method="post">
<div class="row">
```

```
<div class="contact_info">
  <p>
    <label for="name">Jméno: <span>*</span> <br>
    <input type="text" name="message_name"
      required maxlength="100" value="<?php echo
        esc_attr($_POST['message_name']); ?>">
    </label>
  </p>
  <p>
    <label for="message_email">Email: <span>*</
      span> <br>
    <input type="text" name="message_email"
      required maxlength="255" value="<?php echo
        esc_attr($_POST['message_email']); ?>">
    </label>
  </p>
  <input type="hidden" name="contactMessageSend"
    value="1">
  <p>
    <input type="submit" value="Odeslat formulář"
      name="submitBTN" class="submitBTN">
  </p>
</div>
<div class="contact_message">
  <p>
    <label for="message_text">Váš dotaz: <span>
      >*</span>
    <br>
    <textarea type="text" name="message_text"
      required placeholder="Vyplňte váš dotaz
      prosím." rows="7" cols="50" maxlength="500"
    >
    <?php echo esc_textarea($_POST['message_text']
      ); ?></textarea>
    </label>
  </p>
</div>
```

```
</div >  
</form >  
</div >
```

Po stisknutí „odeslat formulář“ dojde k ověření, zda byla všechna pole vyplněna a pokud ano, pak se pomocí php funkce odesílá email na definovanou adresu, kde předmětem emailu je vyplněné jméno spolu s uvedeným emailem. Obsah emailu pak odpovídá vyplněnému textovému poli pro dotaz.

Soubor *functions.php* byl rozšířen o následující kód.

```
function contact_form_func () {  
  ob_start ();  
  get_template_part( 'contactform' );  
  return ob_get_clean ();  
}  
add_shortcode( 'contact_form', 'contact_form_func' );
```

Funkce `add_shortcode` umožní použití zkratky `[contact_form]` v editoru redakčního systému.

Vlastní stránka pak v editoru systému wordpress byla zapsána pomocí html následovně.

```
<section id="eshopinfo">  
  <h2>Eshop sro</h2>  
  <p>Oficiální distributor produktů pro Česko a  
    Slovensko.</br>  
  <strong>Adresa provozovny a odběrného místa:</strong>  
  Kamýcká 129,  
  Praha–Suchdol  
  165 00  
</p>  
  <p><a href="mailto:eshop@eshop-domena.cz" target="_  
    _blank">eshop@eshop-domena.cz</a>  
  tel.: +420 466 260 133</br>  
  Technická podpora eshopu:  
  +420 YYY YYY YYY  
  Servisní oddělení:  
  +420 XXX XXX XXX  
</p>  
</section>
```



[ contact\_form ]

### Domovská stránka

Úvodní stránka domény je jedním z klíčových prvků webové stránky. Zpravidla je to první stránka, kterou zákazník navštíví, a proto je zde třeba zákazníka zaujmout. Pro e-shop byla vytvořena úvodní stránka obsahující slider s nejnovějšími produkty, které jsou například v akci nebo je chce prodejce zdůraznit. Slider je využit pomocí nainstalovaného pluginu TC WooCommerce Product Slider. Dále bylo na stránku umístěno zdůvodnění, proč nakupovat právě v tomto e-shopu, což je prostor pro copywriting. Pod těmito informacemi byla umístěna sekce s několika nejnovějšími příspěvky, respektive články, které byly na webových stránkách uveřejněny. Z hlavní stránky byl rovněž odstraněn úvodní nadpis s názvem stránky, který byl na úvodní stránce přebytečný.

Pro výpis třech nejnovějších článků byl opět připraven shortcode jako v případě kontaktního formuláře.

```
<section id="latest-posts">
<header class="section-header">
    <h2>Nejnovější příspěvky </h2>
</header><!-- .entry-header -->
<div class="row">
    <?php
    $args = array(
        'posts_per_page' => 3,
        'orderby' => 'date',
        'order' => 'DESC',
        'post_type' => 'post',
        'post_status' => 'publish',
        'suppress_filters' => true);
    $latestPost = new WP_Query($args);
    while ($latestPost->have_posts()) : $latestPost->
        the_post();?>
    <article id="post-<?php the_ID(); ?>" <?php
        post_class(); ?>>
        <header class="post-header">
        <?php
        the_title(sprintf('<h3 class="post-title"><a
            href="%s" rel="bookmark">', esc_url(
```

```

        get_permalink ( ) ) , ' </a></h3>' ) ;
?>
</header><!-- .entry-header -->
<span class="post-category">
<?php
    $categories = get_the_category ( ) ;
    $separator = ' ' ;
    $output = '' ;
    if ( !empty ( $categories ) ) {
        foreach ( $categories as
            $category ) {
            $output .= ' <a href="' .
                esc_url ( get_category_link (
                    $category ->term_id ) ) . '"
                alt="' . esc_attr ( sprintf (
                    __( 'View all posts in %s' ,
                        'textdomain' ) , $category
                    ->name ) ) . '" >' . esc_html
                    ( $category ->name ) . ' </a>'
                    . $separator ;
            }
        echo trim ( $output , $separator
            ) ;
    }
?>
</span>
<div class="post-summary">
    <?php the_excerpt ( ) ; ?>
</div>
<div class="article-button">
    <a href=" <?php the_permalink ( ) ; ?>"
        class="button">Přejít na článek </a
    >
</div>
</ article >
<?php
    endwhile ;

```

```

        wp_reset_postdata ( ) ;
    ?>
</div>
</section>

```

Pro zobrazení nejnovějších článků bylo potřeba využít předdefinované funkce redakčního systému. Samotné zobrazení příspěvků pak probíhalo v cyklu, který zpracovává jednotlivé články, které získal z vytvořeného dotazu, který byl zavolán spolu s kritérii jako argumenty funkce. Struktura zobrazeného článku se skládá z jeho nadpisu, kategorie, náhledového obrázku a zkráceného textu spolu se závěrečným tlačítkem odkazujícím na celý článek. Po zpracování článků je doporučeno volat funkci na vyčištění dat získaných v smyčce.

Soubor *functions.php* byl rozšířen o následující kód.

```

function latest_posts_func ( ) {
ob_start ( ) ;
get_template_part ( 'latest-posts' ) ;
return ob_get_clean ( ) ;
}
add_shortcode ( 'latest_posts' , 'latest_posts_func' ) ;

```

Zkratka pro použití v editoru redakčního systému pak je [latest-posts]. Samotný kód úvodní stránky v editoru redakčního systému vypadal následovně.

```

<section class="latest-products">
    <h1 class="entry-title">Nejnovější produkty?</h1>
    [tc-latest-product]
</section>
<section id="why-us">
    <h2 class="entry-title">Proč E-shop?</h2>
    <div class="row why">
        <div class="reason">
            <h3>Důvod 1</h3>
            <p>Protože důvod 1</p>
        </div>
        <div class="reason">
            <h3>Důvod 1</h3>
            <p>Protože důvod 1</p>
        </div>
    </div>

```

```
<div class="reason">
    <h3>Důvod 2</h3>
    <p>Protože důvod 2</p>
</div>
<div class="reason">
    <h3>Důvod 3</h3>
    <p>Protože důvod 3</p>
</div>
<div class="reason">
    <h3>Důvod 4</h3>
    <p>Protože důvod 4</p>
</div>
<div class="reason">
    <h3>Důvod 5</h3>
    <p>Protože důvod 5</p>
</div>
</div>
</section>
[ latest_posts ]
```

#### 4.2.7.4 Úprava CSS

Pro úpravy vzhledu bylo do dříve vytvořeného souboru pro vlastní kaskádové styly využito následujícího kódu. Některé kaskádové styly šablony byly přetíženy pomocí klauzule *!important*. Došlo k úpravám podkladu menu mobilního i standardního. V kódu je rovněž zaveden vzhled pro kontaktní formulář vytvořený v předchozí kapitole. Rovněž došlo k vložení ikony domečku před odkaz na domovskou stránku.

#### 4.2.7.5 Úprava postranní lišty

Postranní lišta umožňuje zobrazování tzv. widgetů, což jsou menší informační či funkční prvky, které se v sidebaru zobrazují. Vzhledem k tomu, že se jedná o e-shopovou stránku, je vhodné do sidebaru vložit i vhodné widgety. Plugin WooCommerce jich přináší do redakčního systému celou řadu. Sidebar se upraví v sekci vzhled výběrem možnosti „Widgety“. Z postranního panelu pomocí metody vzít a zahodit byli zahozeny stávající widgety a vloženy nové. Sidebar se pak skládal z widgetů pro zobrazení kategorií produktů, filtrování dle ceny, zobrazení tří nejlépe hodnocených produktů a vyhledávání produktů. Widgety je také možné umístit do zápatí stránek.

Vzhledem k tomu, že šablona obsahuje jeden postranní panel, který se zobrazuje vždy a to i na stránce s články, bylo vhodné vytvořit postranní lištu pro stránku s články. Proto je nutné rozšířit soubor *sidebar.php* o následující kód, respektive byl na následující kód přepsán.

```
<?php
if (is_home() && !is_front_page())
{
    if (!is_active_sidebar('sidebar-blog')) {
        return;
    }
?>
<div id="secondary" class="widget-area" role="complementary">
    <?php dynamic_sidebar('sidebar-blog'); ?>
</div><!-- #secondary -->
<?php
}
else {
    if (!is_active_sidebar('sidebar-1')) {
        return;
    }
?>
<div id="secondary" class="widget-area" role="complementary">
<?php dynamic_sidebar('sidebar-1'); ?>
</div><!-- #secondary -->
<?php
}
?>
```

Také bylo nutné nový sidebar zaregistrovat v systému, což bylo provedeno přidáním následujícího kódu do souboru *class-storefront.php* a to konkrétně do funkce *widgets\_init*.

```
$sidebar_args['sidebar-blog'] = array(
    'name' => __('Sidebar blog', 'storefront'),
    'id' => 'sidebar-blog',
    'description' => __('Zobrazuje se na stránce s
    příspěvky.', 'storefront')
);
```

Po těchto úpravách se nový sidebar zobrazil v redakčním systému v sekci vzhled položka widgety a využity v něm byly widgety pro zobrazení rubrik článků, nejnovějších článků (příspěvků) a zobrazení štítků článků.

#### 4.2.7.6 Vkládání obsahu

Tvorba produktů nebo článků a jejich vložení do redakčního systému probíhá také na administrační stránce. Pokud má uživatel přidělena daná práva, může články a produkty, vkládat, modifikovat nebo mazat. Vložení produktů, které se na e-shopu prodávají, má řadu nastavitelných parametrů.

Obrázek 24: Parametry produktu pro plugin WooCommerce Zdroj: Vlastní tvorba

Kromě běžného uvedení názvu, zařazení produktu do kategorií a jeho oštitkování, se produktu může přidat i obrázek, respektive další obrázky pro galerii produktu. Rovněž se přidává krátký popis produktu, zpracovaný ideálně copywriterem pro zaujetí a přínos hodnotných informací zákazníkovi. U produktu je možno nastavit zdali je virtuální, nebo ke stažení. Parametrizace samotného produktu je rozdělena pluginem WooCommerce do sekcí obecné, sklad, doručení (není k dispozici pro virtuální produkty), provázané produkty, vlastnost a pokročilé nastavení.

Sekce obecné umožňuje nastavit cenu produktu před a po slevě, jeli nějaká sleva aktuální, dále daňovou třídu produktu, a zda je zdanitelný či nikoliv, nebo zda se daní pouze poštovné. V sekci sklad je možno nastavit katalogové číslo produktu, stav produktu, zdali je skladem či nikoliv, zdali je produkt omezen na koupi jednoho kusu na objednávku a zdali povolit jeho správu v rámci skladu, což rozšiřuje nastavení o možnost úpravy množství skladem a povolení nákupu na objednávku. V sekci doručení se nastavuje hmotnost, rozměry a doručovací třída produktu. Sekce provázané produkty umožňuje nastavení manuálního crossel-

lingu a upsellingu, popřípadě seskupení produktu, patří-li produkt do nějaké skupiny s jinými produkty. Vlastnosti produktu je uživatelem nastavitelná hodnota. Takto je možné produktu nastavit vlastnosti jako je například výkon, spotřeba elektrické energie, nosnost či jiné další atributy. V sekci pokročilý je možno nastavit povolení recenzí produktu, vlastní řazení produktu a poznámku, která bude zákazníkovi odeslána po zakoupení produktu.

Tvorba článků v redakčním systému wordpress se provádí obdobně. Nastavit lze kategorií a štítky článku, u některých šablon lze nastavit i typ příspěvku, tedy zdali se jedná o obrázek, video atd. U psaní článků se často využívá značka „<!--more-->“, která reprezentuje jakousi zarážku. Cokoliv je napsáno před touto značkou, je zobrazeno na stránce s příspěvky. Ostatní texty, obrázky či jiné elementy uvedené za značkou „<!--more-->“ jsou zobrazeny pouze při zobrazení celého článku.

Redakční systém byl naplněn třemi demonstrativními články a pěti demonstrativními produkty.

#### 4.2.7.7 Další rozšiřující funkcionality

Rozšiřující funkcionality byly realizovány tvorbou vlastního pluginu, který kromě napojení účetní softwaru a realizace procesů s tím spojeným rozšiřuje samotný plugin WooCommerce.

E-shop realizovaný pluginem WooCommerce respektive objednávka v něm evidovaná se nachází v různých stavech, které systém přiřazuje. Logika stavů objednávek byla rozšířena o stav ready, který signalizuje připravenost objednávky k vyzvednutí, případně značí předání objednávky přepravci, který jí doručí. Přidání stavu objednávky bylo provedeno následujícím kusem kódu.

```
function register_order_ready_status () {
    register_post_status ( 'wc-ready' ,
        array (
            'label' => 'Ready' ,
            'public' => true ,
            'exclude_from_search' => false ,
            'show_in_admin_all_list' => true ,
            'show_in_admin_status_list' => true ) );
}
add_action ( 'init' , 'register_order_ready_status' );

function add_order_ready_status ( $order_statuses ) {
```

```
$new_order_statuses = array();  
foreach ($order_statuses as $key => $status) {  
    $new_order_statuses[$key] = $status;  
    if ('wc-processing' === $key) {  
        $new_order_statuses['wc-ready'] = '  
            Ready';  
    }  
}  
return $new_order_statuses;  
}  
add_filter('wc_order_statuses', 'add_order_ready_status');
```

## 4.2.8 Napojení eshopu na účetní software

Jako účetní software bylo zvoleno řešení od společnosti Abra Software, konkrétně produkt Abra FlexiBee, který umožňuje správu účetnictví online pro menší podniky. Pro správu účetnictví a jeho propojení s elektronickým obchodem je nutné FlexiBee nainstalovat síťově pro přístup uživatelů k účetnictví. Rovněž byl naimportován přidělený licenční soubor. V rámci napojení e-shopu se jedná zejména o přenos dat vytvořených objednávek, ze kterých je v systému vytvořena faktura a to skrze webové API účetního systému. Samotná data jsou předána strukturovaně pomocí XML.

### 4.2.8.1 Propojení s flexibee

Účetní software FlexiBee vystavuje pro potřeby propojení webové API, které zdokumentovává webová dokumentace na stránkách produktu.[odkaz] Jedná se o REST API založené na http protokolu, které umožňuje vracet a přijímat různé formáty. Přístup k API je možno realizovat pomocí PHP, nebo JSON voláním REST služeb. Zvoleno bylo napojení za pomoci PHP, vzhledem k faktu, že samotný redakční systém je postaven na PHP technologii.

Volání služeb probíhá využitím funkcí z PHP knihovny s názvem client URL, tedy klient-ské URL. Tato knihovna obsahuje veškeré funkce pro inicializaci spojení, předání a nastavení potřebných parametrů a provedení samotného požadavku. Po zpracování požadavku by mělo dojít k ukončení spojení voláním funkce *curl\_close*. Postup napojení na SW FlexiBee začíná inicializací spojení pomocí funkce *curl\_init* a poté následuje nastavení parametrů funkcí *curl\_setopt*. Nastaven byl parametr *CURLOPT\_RETURNTRANSFER* na hodnotu *TRUE*, což značí předání výsledku funkce *curl\_exec* ve formě stringu. Parametr *CURLOPT\_FOLLOWLOCATION* s hodnotou *TRUE*, zajišťuje kompatibilitu pro případně změny



v API a umožňuje následovat přesměrování, je-li nějaké přesměrování nutné. Nastavena byla i http autentizace parametrem *CURLOPT\_HTTPAUTH* a deaktivována byla SSL verifikace, jelikož SW FlexiBee využívá vlastních certifikátů. Nastavení přístupového hesla a uživatele proběhlo využitím parametru *CURLOPT\_USERPWD*. Tyto parametry jsou nutným základem pro odeslání požadavku na FlexiBee API. Další parametry se odlišují, jelikož je volána odlišná služba a odlišný požadavek. Služby se odlišují voláním URL, které se zapisuje do parametru *CURLOPT\_URL* a požadavkem, který se ukládá do parametru *CURLOPT\_CUSTOMREQUEST*. Využito bylo požadavků *put* a *get*. Po nastavení všech potřebných požadavků došlo pomocí funkce *curl\_exec* k provedení požadavku. Pro získání informací o provedeném požadavku zejména pro účely ošetření případných chyb bylo využito funkce *curl\_getinfo*. Volané URL svou strukturou odpovídá následujícímu kódu, kde proměnná *host* představuje webovou adresu k účetnímu softwaru FlexiBee a proměnná *firma* představuje název firmy, kterou chceme v účetním SW upravovat. V následujícím případě jde o získání faktury dle jejího kódového označení.

```
curl_setopt($ch , CURLOPT_URL, $host . "/c/" . $firma . "/"  
    faktura-vydana/" . $code );
```

#### 4.2.8.2 Tvorba faktury

Faktura je tvořena na základě přepnutí stavu objednávky do stavu dokončena. Tento stav značí, že prodejce přijal obnos od zákazníka, který si přišel fakturu vyzvednout. Po přepnutí stavu v systému dojde k vygenerování faktury, která je uložena v účetním softwaru FlexiBee a dojde i k automatickému odečtení skladových položek pro objednané produkty, které zákazník vyzvedl. Této automatické funkcionalitě bylo docíleno rozšířením existující akce *woocommerce\_order\_status\_completed* o vlastní funkci *create\_invoice* obsaženou v pluginu.

```
// Přidání akce po přepnutí objednávky do stavu dokončená  
add_action( 'woocommerce_order_status_completed' , '  
    create_invoice' );
```

Samotná funkce pro tvorbu faktury obsahuje připojení na server účetního softwaru a naplnění XML struktury potřebnými daty z objednávky. K naplnění potřebných dat je nutné předat XML strukturu pomocí parametru *CURLOPT\_POSTFIELDS* php funkce *curl\_setopt*. Naplnění XML struktury, která je definována na stránkách produktu FlexiBee, vyžaduje vložení dat objednávky do příslušných XML značek. Funkce *create\_invoice* je spuštěna vždy, když dojde k přepnutí stavu určité objednávky. Proto je k dispozici její id a na základě id dojde k vytvoření objektu *WC\_ORDER*, což je požadovaná objednávka. Využitím objednávky jako objektu se získá kompletní přístup k jejím atributům a metodám. Využitím funkce

*get\_items* dochází k získání potřebných atributů i o produktech, které jsou s objednávkou spjaty. Do redakčního systému je k objednávce doplněna poznámka o úspěšném či neúspěšném vygenerování faktury. Zákazník je rovněž informován o dokončení objednávky a na jeho email se odesílá kromě základních faktů generovaných systémem i odkaz na fakturu ke stažení. Pro tuto funkci bylo nutné po vytvoření faktury v účetním systému zpětně získat jedinečný identifikátor této faktury a to dle jejího kódu, který má jednotnou definovanou podobu končící stejným číslem jako bylo číslo objednávky. API účetního softwaru pomocí volání URL s vložením kódu vrací výstup, ze kterého bylo pomocí php funkce *substr* a *trimm* vyparsováno jedinečné číslo respektive identifikátor faktury. Jedinečný identifikátor faktury se doplnil do URL a to bylo formou odkazu doplněno do těla emailu odesílaného zákazníkovi po přesunutí objednávky do stavu dokončena. Fakturu je možno exportovat do PDF, XML souboru, nebo ISDOC.

#### 4.2.8.3 Aktualizace skladového hospodářství

Dle nastavení produktů v e-shopu je u každého produktu vedena jeho skladová evidence. Obdobně jsou v účetním softwaru evidovány jednotlivé skladové karty produktů. Při vytvoření faktury dochází v účetním softwaru a v redakčním systému k automatickému snížení skladovaného množství dané položky, která je evidována na faktuře respektive objednávce a to o zakoupené množství. Synchronizaci skladových položek v účetním softwaru s produkty evidovanými v redakčním systému je tak potřeba provádět kvůli dodávkám respektive příjmu zboží na sklad dle účetního softwaru. Toho bylo dosaženo pomocí webové API, která voláním URL s vybranými parametry vrací XML strukturovaná data všech produktů evidovaných v účetním softwaru.

Se zobrazením XML dat v redakčním systému byl zjištěn jeden problém. CMS Wordpress totiž XML značky při výpisu neuvádí a zobrazuje pouze strohé hodnoty jednotlivých XML elementů. Data přijatá z API flexibee tak byla uložena jako SimpleXMLElement, což je jedna z možností jak s XML výstupy pracovat pomocí jazyka PHP.

```
// output přijata data z flexibee  
$flexibeeProducts = new SimpleXMLElement($output) or die("  
    Error: Cannot create object");
```

Získaná data z FlexiBee jsou modifikovatelná pomocí předaných parametrů na konci URL. Tímto způsobem byly získány pouze atributy identifikátor produktu v účetní systému (XML element „id“), kód produktu (XML element „kod“) a množství produktu skladem (XML element „sumStavMj“).

Pro aktualizace položek produktů v e-shopu bylo nutné produkty z redakčního systému získat. To bylo provedeno vytvořením dotazu do redakčního systému s argumentem „product“. Redakční systém wordpress vede objednávky i produkty a další nově přidané položky nepřímou jako příspěvky. To způsobuje drobnou mystifikaci, která ale umožňuje právě využití wordpress dotazování. Parametry pro výpis produktů z redakčního systému proto vypadají následovně.

```
$args = array(
    'post_type' => 'product',
    'post_status' => 'publish',
    'posts_per_page' => -1
);
```

Voláním dotazu s těmito parametry vrací výsledek, který je zpracován v cyklu a voláním metody *the\_post* jsou získána potřebná data o produktu. Dle získaného id produktu byl obdobně jako v případě objednávky vytvořen produkt jako objekt pro přístup k jeho atributům a metodám. Procházením XML struktury pomocí cyklu a zachycením požadovaného kódu produktu, dochází následně k nastavení skladového množství produktu v redakčním systému pomocí předdefinované funkce pluginu WooCommerce.

```
for ( $i = 0; $i <= $flexibeeProducts->count(); $i++)
{
    if ( ( $flexibeeProducts->cenik[ $i ]->kod == $code ) )
    {
        $_product->set_stock( $flexibeeProducts->cenik
            [ $i ]->sumStavMj, 'set' );
        break;
    }
}
```

Synchronizaci skladových zásob je možné provést manuálně z rozhraní pluginu. Dále byla synchronizace nastavena pomocí cron úlohy na každodenní provedení a to ve dvě hodiny ráno. Nastavení cron úloh proběhlo pomocí rozšíření naplánovaných úloh redakčního systému. Rozšíření o cron úlohu bylo zapracováno do rozhraní pluginu tak, aby byl skript k synchronizaci spuštěn každý den v 02:00. Samotná funkce pro synchronizaci se nachází v adresáři pluginu v souboru *functions.php*. Pro naplánování cron úlohy byl použit následující kód.

```
if ( ! wp_next_scheduled( 'stock_synchronize_cron_action' ) )
{
```

```
        wp_schedule_event( strtotime( '01:00:00' ), 'daily', '  
            stock_synchronize_cron_action' );  
    }  
    add_action( 'stock_synchronize_cron_action', '  
        stock_synchronize' );
```

Pro sledování, upravování, přidávání a odstraňování cron úloh je též možno využít plugin WP Cronrol. Správa cron úloh by však měla být pouze v rukou administrátora.

## 5 Zhodnocení výsledků a doporučení

Výsledkem této práce je funkční open source řešení umožňující provoz internetového obchodu spolu s jeho údržbou, které je možné dále rozšiřovat a rozvíjet. Dále umožňuje práci s produkty a objednávkami, které automaticky po jejím dokončení přenáší do účetního softwaru, který provádí fakturaci a odečtení skladových zásob. Rovněž dochází i k pravidelné jednodenní synchronizaci skladových zásob podniku. Řešení je připravené k nasazení do provozu a určené k reálnému použití zejména začínajícími živnostníky, podniky, nebo startupy a malými firmami.

V souvislosti s prací s účetním softwarem by bylo do budoucna možné rozšiřovat funkce e-shopu o další automatizace se softwarem FlexiBee, u kterého by však bylo nutné zakoupit některou z placených licencí. Tím by došlo k rozšíření přístupu v oblasti dostupných funkcionalit, které by bylo možné do redakčního systému dále doplňovat. Rozšiřovat by bylo možné oblasti skladového hospodářství, management vztahu se zákazníky a další oblasti. Samotný e-shop by bylo rovněž možné dále rozvíjet, jelikož ne všechny funkce jsou pluginem WooCommerce vhodně řešeny, a některé pro budoucnost vhodné funkce nemá plugin vůbec. Je však třeba brát ohled na limity, které s redakčními systémy a open source řešeními souvisí. Rozsáhlé přidávání funkcionality a propojování různých nástrojů může narazit na nedostatečnou optimalizaci. Náklady na údržbu a provoz mohou být poté vyšší, než u jiného specializovaného řešení.

## 6 Závěr

Celá práce rozebírá problematiku realizace internetového obchodu od prvotní analýzy dostupných open source řešení, přes realizaci vybraného redakčního systému až k jeho napojení na účetní software. Vymezuje teoretické znalosti a výrazy a prakticky demonstruje realizaci pro reálné využití elektronického obchodu. Výsledkem práce je analýza dostupných open source řešení a postup realizace vybraného řešení, které je vhodné pro praktické nasazení u malých a začínajících firem, prodávajících zboží na internetu. Toto řešení bylo dále napojeno na účetní software vybraný na základě dostupných informací a požadavcích stanovených pro výběr tohoto softwaru.

Použitá metodika zadané diplomové práce se skládala ze studia a analýzy odborných informačních zdrojů a ze zmapování dostupných možností. Stěžejní bylo využití technologií PHP, HTML a CSS a využití metody vícekritériální analýzy variant. Navrhované řešení vychází z analýzy dostupných open source řešení a zohledňuje požadavky klienta, respektive začínajících malých či startupových firem. Na základě syntézy teoretických poznatků a výsledků analýzy jsou formulovány závěry této diplomové práce.

Přínos této práce představuje její použitelnost jako manuál pro realizaci e-shopu v redakčním systému wordpress od prvotní konfigurace, přes instalaci vhodných pluginů a výběr šablony, až k drobným i rozsáhlejšími úpravám kódu a následným propojením s účetním softwarem FlexiBee. Přínosem je i analytický pohled na problematiku výběru vhodného open source redakčního systému pro potřeby malých podniků, nebo začínajících živnostníků a startupových projektů. Samotná analýza ukazuje váhy jednotlivých kritérií a hodnocení variant v těchto kritériích, tedy i vhodnost jednotlivých redakčních systémů v závislosti na jejich parametrech. Celkovou sumarizací všech ohodnocení s ohledem na váhy kritérií vyšel vítězně redakční systém Wordpress. E-shop byl pomocí vlastního pluginu napojen na účetní software FlexiBee, kdy bylo realizováno propojení respektive synchronizace skladového hospodářství a fakturace na základě objednávek vytvořených v e-shopu a rovněž zpětná propagace odkazu na vytvořenou fakturu. Tento odkaz, kde se faktura ke stažení nachází, se zasílá v emailu spolu s dalšími informacemi o dokončené objednávce. Navrhované řešení je realizováno s využitím dostupných open source možností pro tvorbu elektronického obchodu.

Naplnění cíle práce bylo dosaženo využitím redakčního systému Wordpress, který byl doplněn o plugin WooCommerce, jenž rozšiřuje funkcionalitu redakčního systému o funkce pro provoz e-shopu. Redakční systém byl vybrán na základě analýzy využitím vícekritériálního rozhodování, konkrétně metodou vícekritériální analýzy variant. Napojení na účetní software bylo realizováno pomocí předávání požadavků a parametrů volání v rámci URL

vystaveného webového API produktu FlexiBee. Pro tento účel byl v rámci práce vytvořen vlastní plugin. Data jsou předávána v XML struktuře a jejich výměna je zajištěna pomocí jazyka PHP a voláním dostupných funkcí redakčního systému Wordpress.

Během realizace bylo odhaleno několik úskalí, souvisejících s podstatou redakčních systémů. Ty bylo nutno důkladně prostudovat a zjistit příčinu těchto úskalí, aby došlo k odstranění nastalých problémů. Jedním z problémů, při psaní vlastního kódu rozšiřujícího práci s objednávkami, byl přístup k objednávkám v rámci redakčního systému. Jelikož se primárně jedná o nástroje pro správu blogů, je práce s e-shopem pouze určitou nadstavbou nad existující logikou systému. Objednávky a další prvky e-shopu jsou proto systémem řešeny obdobně jako příspěvky nebo články, které jsou rozlišeny atributem typ příspěvku. Po zjištění tohoto faktu byla tato struktura zachována a k datům bylo přistupováno touto cestou. Po získání potřebných identifikátorů byl vytvořen patřičný objekt, který byl využit pro přístup k atributům a metodám. Obdobně byl řešen i problém s přístupem k produktům elektronického obchodu. Dalším specifikem systému Wordpress je automatická interpretace XML značek. Výpis přijatých dat byl proto oddělen pouze mezerou, nikoliv XML značkami. To vedlo k mylnému pochopení a špatné interpretaci výsledků, se kterými se dále pracovalo. Za použití PHP byl problém řešen vytvořením XML elementu, který byl dále zpracován pro potřeby synchronizace skladových zásob. Poměrně časově náročným problémem bylo rozklíčování fungování e-shopových funkcí pluginu WooCommerce, který měl místy nepřesnou dokumentaci, což zapříčinilo řadu chyb a omylů, které museli být následně opravovány. Autoři pluginu poskytují support pro neplacené služby jen v omezené míře a proto bylo využito dalších dostupných zdrojů a uživatelský fór. Přehlednější dokumentaci nabízel účetní software FlexiBee, který disponoval i různými příklady pro práci s ním, nicméně i tato dokumentace vyžadovala mnoho času pro pochopení a nutnost dalších zdrojů, zejména pro vhodnou práci s XML strukturou.

Cílem práce byla realizace open source řešení e-shopu. Na základě vícekritériální analýzy byl vybrán redakční systém Wordpress s rozšířením Woocommerce, který byl napojen na účetní software Flexibee. Výsledné řešení je připravené na nasazení v reálném provozu a speciálně vhodné pro menší a začínající firmy.

## Literatura

- [1] MANZOOR Amir. E-Commerce: An Introduction. LAP LAMBERT Academic publishing, 2010. ISBN 978-3-8433-7030-1.
- [2] SUCHÁNEK Petr. E-commerce. Ekopress, 2012. ISBN 978-80-86929-84-2.
- [3] BOIKO Bob. Content Management Bible, 2nd Edition. Indianapolis: Wilie Publishing, Inc., 2004. ISBN 0-7645-7371-3.
- [4] BARKER Deane. Web Content Management. O'Reilly Media, 2016. ISBN 978-1491908129.
- [5] MOZILLA DEVELOPER NETWORK. Online Web Technology For Developers [cit. 2016-10-23]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web>.
- [6] DOYLE Matt. Beginning PHP 5.3. Wrox, 2009. ISBN 978-0470413968.
- [7] SABIN-WILSON Lisa. Wordpress For Dummies 6th Edition. John Wiley & Sons, Inc 2014. ISBN 978-1-118-79161-5.
- [8] RAHMEL Dan. Professional Joomla!. Wiley Publishing, Inc., 2007. ISBN 978-0-470-13394-1.
- [9] TOMLINSON Todd. Beginning Drupal 8. Apress 2015. ISBN-13 978-1-4302-6581-8.
- [10] HORTON J., TIZON A. J. PrestaShop 1.5. Beginners guide, Packt Publishing Ltd. 2013. ISBN 978-1-78216-106-6.
- [11] CANAVAN Tom. CMS Security Handbook. Wiley Publishing, Inc., 2011. ISBN 978-0-470-91621-6.
- [12] BANGIA Ramesh. Web Technology. Laxmi Publications, 2006. ISBN 9788170088974.
- [13] HALES Wesley. HTML5 and Javascript Web Apps. O'Reilly Media, 2012. ISBN 978-1-449-32051-5.
- [14] NIXON Robin. Learning PHP, MySQL & JavaScript with JQUERY, CSS & HTML5. O'Reilly Media, 2014. ISBN 978-1-491-91866-1.
- [15] ŠENKAPOUN Pavel. Webcopywriting pro samouky : průvodce moderního marketéra a podnikatele po webovém obsahu. Zoner Press, 2015. ISBN 978-80-7413-176-9.



- [16] MOODLE CZU [online]. Přednášky z předmětu E-business. [cit. 2016-10-17]. Dostupné z: <https://moodle.czu.cz/>
- [17] CALIBR [online]. E.R.P Integration . ©2016 [cit. 2016-10-17]. Dostupné z: <http://www.calibr-feofus.com/erp-integration/>
- [18] EMERALD INSIGHT [online]. Library Hi Tech Journal - Content management systems. 2016 [cit. 2016-10-17]. Dostupné z: <http://www.emeraldinsight.com>
- [19] LAMINACK Brent [online]. A Brief History of Content Management Systems. 2001 [cit. 2016-10-23]. Dostupné z: <http://laminack.com/index.php/technology/11-a-brief-history-of-content-management-systems>
- [20] W3SCHOOLS [online]. HTML, CSS, JavaScript. 2016 [cit. 2016-10-23]. Dostupné z: <http://www.w3schools.com/>
- [21] BUILT WITH [online]. Open Source usage in Czech Republic. 2016 [cit. 2016-10-23]. Dostupné z: <http://trends.builtwith.com/cms/open-source/country/Czech-Republic>
- [22] GNU [online]. GNU General Public License. 2016 [cit. 2016-10-23]. Dostupné z: <https://www.gnu.org/licenses/gpl-3.0.html>
- [23] WORDPRES.ORG [online]. History. 2016 [cit. 2016-10-23]. Dostupné z: <https://codex.wordpress.org/History>
- [24] TECHWYSE [online]. Big Benefits Of Using Joomla To Manage Your Website. 2010 [cit. 2016-10-23]. Dostupné z: <https://www.techwyse.com/blog/website-design/big-benefits-of-using-joomla-to-manage-your-website/>
- [25] DRUPAL [online]. O systému Drupal. 2012 [cit. 2016-10-23]. Dostupné z: <https://www.drupal.cz/>
- [26] BRAINWIRE [online]. PrestaShop Technology Expertise. 2016 [cit. 2016-10-23]. Dostupné z: <http://www.brainvire.com/prestashop/>
- [27] ČZU PEF [online]. Vícekriterální analýza variant. Brožová 2016 [cit. 2016-11-2]. Dostupné z: <http://pef.czu.cz/~BROZOVA/CASESTUDY/VAV1.html>
- [28] VŠE IASTAT [online]. Typy proměnných. 2016 [cit. 2016-11-2]. Dostupné z: [http://iastat.vse.cz/typy\\_promennych.html](http://iastat.vse.cz/typy_promennych.html)
- [29] BUSINESS DICTIONARY [online]. What is accounting system? Definition and meaning 2016 [cit. 2016-12-27]. Dostupné z: <http://www.businessdictionary.com/definition/accounting-system.html>

- [30] DEBITOOR DICTIONARY [online]. Accounting system - What is an accounting system? 2016 [cit. 2016-12-27]. Dostupné z: <https://debitoor.com/dictionary/accounting-system>
- [31] PODNIKATEL.CZ [online]. Vybíráme za vás: Účetní programy zadarmo i kvalitní placené 2011 [cit. 2016-12-27]. Dostupné z: <http://www.podnikatel.cz/clanky/vybirame-za-vas-ucetni-programy-zadarmo/>
- [32] SLUNECNICE.CZ [online]. PROFIT 2016 [cit. 2016-12-27]. Dostupné z: <https://www.slunecnice.cz/sw/podnikatelsky-system-profit/>
- [33] ABRA FLEXIBEE [online]. ABRA FLEXIBEE 2016 [cit. 2016-12-27]. Dostupné z: <https://www.flexibee.eu/>

# Přílohy

## A Vlastní kaskádové styly

```
/*My custom style

*/

.entry-title { color: #fe9900; }
.page-title { display: none; }
.text-center, .post-14 .entry-header {
    text-align: center;
}
.post-14 { margin: 0 0 1rem 0; font-size: 14px; }
#eshopinfo h2 { margin: 0 0 1rem 0; font-size: 20px; }
.menu-item-home a::before {
    font: normal normal normal 1em/1 FontAwesome;
    display: inline-block !important;
    content: "\f015" !important;
}
.handheld-navigation { background: rgba(0,0,0,.4) !important; }
.handheld-navigation a { padding: 0.875em 5% !important; }
.handheld-navigation a:hover { font-weight: 800; background:
    rgba(0,0,0,.75) !important; }
.row {
    display: flex;
    flex-direction: column;
    align-content: flex-start;
    justify-content: flex-start;
}

/*CSS pro proč eshop*/
.reason { padding: 1rem; flex-grow: 1; }
.reason h3 { border-bottom: 1px solid rgba(254, 153, 0, 0.5); }
.reason p { font-weight: 500; }
.why { background-color: rgba(254, 153, 0, 0.25); border: 2px
    solid #fe9900; }
```

```

/*CSS pro kontaktní formulář*/
.submitBTN{margin-top: 1em;}
.contact_info{order: 1; -webkit-order: 1; padding:0 5%; flex-grow: 1;}
.contact_message{order: 2; -webkit-order: 2; padding:0 5%; flex-grow: 1;}
.contact_message textarea, .contact_info input {width: 100%;}
.contact_form > p:first-of-type{text-align: center !important;}
.contact_form h2{color: #fe9900;}

/* Skrytí nadpisu úvodní stránky*/
.home .entry-header .entry-title {display: none;}
#why-us .entry-title, .latest-products .entry-title, #latest-posts .section-header h2{color: #fe9900;}

/*CSS pro shortcode nejnovější příspěvky*/
#latest-posts{margin-top: 2rem;}
.post-title a{color: #2b2e33;}
#latest-posts .row article{flex-grow: 1; padding: 0 2em; margin-bottom: 1em;}
#latest-posts .row{margin-top: 2rem;}

/* Styly pro plugin TC WooCommerce Product Slider*/
.tcwps-wrap {margin: 0 !important;}
.tc-wps-overlay-block{color: #2b2e33;}
.tc-cart-button{margin-top: 1rem;}
@media (max-width:767px){
    .tc-wps-overlay, .owl-controls{display: none;}
}
@media (min-width:768px){
    .storefront-primary-navigation{background: rgba(0,0,0,.4) !important;}
    .secondary-navigation .menu a {font-weight: 600 !important;}
    .row{flex-direction: row};
    .contact_info{order: 1; -webkit-order: 1; flex-grow:

```

```
        1; }
        .contact_message { order: 2; -webkit-order: 2; flex-grow: 3; }
    }
```

## B Kód pro registraci vlastního pluginu

```
<?php
    require_once( plugin_dir_path( __FILE__ ) . '/include/
        functions.php' );

    /*
    Plugin Name: Caen – Flexibee
    Plugin URI:
    Description: Version: 1.0
    Author: Caen Ragestorm
    Author URI: www.CaenRagestorm.cz
    License: GPLv2+
    Text Domain: wp-caen-flexibee
    */

class WP_Caen_FlexiBee {
    // Constructor
    function __construct() {

        // Přidání akce po přepnutí objednávky do stavu
        dokončená
        add_action( 'woocommerce_order_status_completed', '
            create_invoice' );

        function register_order_ready_status() {
            register_post_status( 'wc-ready',
                array(
                    'label' => 'Ready',
                    'public' => true,
                    'exclude_from_search' => false,
                    'show_in_admin_all_list' => true,
                    'show_in_admin_status_list' => true
                )
            );
        }
    }
}
```

```

        ));
    }
    add_action('init', 'register_order_ready_status');
    // Add to list of WC Order statuses
    function add_order_ready_status($order_statuses) {
        $new_order_statuses = array();
        // add new order status after processing
        foreach ($order_statuses as $key => $status)
        {
            $new_order_statuses[$key] = $status;
            if ('wc-processing' === $key) {
                $new_order_statuses['wc-ready
                '] = 'Ready';
            }
        }
        return $new_order_statuses;
    }

    if ( ! wp_next_scheduled( 'stock_synchronize_cron_action'
    ) ) {
        wp_schedule_event( strtotime('01:00:00'), 'daily', '
        stock_synchronize_cron_action' );
    }
    add_action( 'stock_synchronize_cron_action', '
    stock_synchronize' );
    add_filter('wc_order_statuses', 'add_order_ready_status')
    ;
    add_action('admin_menu', array($this, 'wpa_add_menu'));
    register_activation_hook(__FILE__, array($this, '
    wpa_install'));
    register_deactivation_hook(__FILE__, array($this, '
    wpa_uninstall'));
}

/*Actions perform at loading of admin menu*/
function wpa_add_menu() {
    add_menu_page('Nápojení Flexibee', 'Flexibee Plugin',
    'manage_options', 'caen-flexibee-dashboard',

```

```

        array(__CLASS__, 'wpa_page_file_path')
        , plugins_url('images/wp-analytics-logo.png',
        __FILE__), '2.2.9');
    }

    /*Actions perform on loading of menu pages*/
    function wpa_page_file_path() { include "caen-flexibee-
        dashboard.php"; }

    /*Actions perform on activation of plugin*/
    function wpa_install() {}

    /*Actions perform on de-activation of plugin*/
    function wpa_uninstall() {}

    }
    new WP_Caen_FlexiBee();

```

## C Kód pro zobrazení v záložce admin rozhraní pluginu

```
<h2>Caen Flexibee plugin Dashboard</h2>
```

```
<p>
```

```
<?php
```

```
    if (get_locale() == 'cs_CZ') {
```

```
        echo "Tento plugin byl vytvořen pro
            účely diplomové práce. Pokud by
            jste chtěli získat další informace
            , navštivte mé osobní
            stránky na: ";
```

```
    } else {
```

```
        echo "This plugin had been made for
            the purpose of Diploma thesis. If
            you want know more visit my
            personal pages on: ";
```

```
    }
```

```
?>
```

```
<a href="www.CaenRagestorm.cz" src="external">www.
    CaenRagestorm.cz</a>
```

```
</p>
```

```
<?php
```

```
//must check that the user has the required capability
```

```

if (!current_user_can('manage_options')) { wp_die(__('You do
not have sufficient permissions to access this page.')); }
?>
<h2>
    <?php
        if (get_locale() == 'cs_CZ') {
            echo "Manuální synchronizace";
        } else {
            echo "Manual synchronization";
        }
    ?>
</h2>
<p>
    <?php
        if (get_locale() == 'cs_CZ') {
            echo "Pro manuální překlopení
                objednávek ve stavu zpracovává se
                do stavu k vyřízení (objednávky, u
                kterých není zboží skladem)
                stiskněte následující tlačítko: ";
            $buttonText['state'] = "Překlopit stav
                ";
            $buttonText['stock'] = "Synchronizovat
                stav skladových zásob";
            $resultText = "Id objednávek, kterým
                byl změněn stav";
        } else {
            echo "For manual change of order state from
                processing to on hold (orders with
                products not in stock), please click the
                following button: ";
            $buttonText['state'] = "Change state";
            $buttonText['stock'] = "Synchronize
                products";
            $resultText = "Id of orders with
                changed state";
        }
    }

```



```

?>
</p>
<form action="" method="post">
    <input type="submit" name="changeStateBtn" value="<?
        php echo $butonText['state']; ?>">
    <input type="submit" name="changeStockBtn" value="<?
        php echo $butonText['stock']; ?>">
</form>
<?php
    if (isset($_POST['changeStateBtn'])) {
        include_once(plugin_dir_path(__FILE__).' /
            include/change-order-state.php');
    }
    else if (isset($_POST['changeStockBtn'])) {
        include_once(plugin_dir_path(__FILE__).' /
            include/change-products-stock.php');
    }
?>

```

## D Kód funkci pluginu

```

<?php
function create_invoice($order_id) {
    include_once(plugin_dir_path(__FILE__) . 'connection-
        initiation.php');
    $order = new WC_Order($order_id);

    // Požadavek odešleme ve formátu XML
    curl_setopt($ch, CURLOPT_URL, $host . "/c/" . $firma
        . "/faktura-vydana.xml");
    // Nastavení samotné operace
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "PUT");
    if ($order->billing_company != "") {
        $company = $order->billing_company;
    } else {
        $company = $order->
            get_formatted_billing_full_name();
    }
}

```

```

}
// Nastavení parametrů pro založení jednoduché
  faktury
$invoiceXML = '
    <winstrom version="1.0">
    <faktura-vydana>
        <kod>VF1-' . $order_id . ' </kod>
        <last_update>' . date('Y-m-d H:i:s')
            . ' </last_update>
        <typDokl showAs="FAKTURA: Faktura -
            daňový doklad">code:FAKTURA</
            typDokl>
        <mena showAs="CZK: Česká koruna">code
            :CZK</mena>
        <generovatSkl>>true </generovatSkl>
        <cisObj>' . $order_id . ' </cisObj>
        <datObj>' . date("Y-m-d", strtotime(
            $order->order_date)) . ' </datObj>

        <!-- přímo uvedené kontaktní údaje.
            Není nutné vyplnit, pokud se mají
            použít hodnoty z adresáře -->
        <nazFirmy>' . $company . ' </nazFirmy>
        <kontaktJmeno>' . $order->
            get_formatted_billing_full_name()
            . ' </kontaktJmeno>
        <kontaktEmail>' . $order->
            billing_email . ' </kontaktEmail>
        <kontaktTel>' . $order->billing_phone
            . ' </kontaktTel>
        <ulice>' . $order->billing_address_1
            . ' ' . $order->billing_address_2
            . ' </ulice>
        <mesto>' . $order->billing_city . ' </
            mesto>
        <psc>' . $order->billing_postcode . '
            </psc>

```

```

        <stat>' . $order->billing_state . '</
            stat>
        <poznám>' . $order->customer_note . '
            </poznám>
        <polozkyFaktury>';
$order_item = $order->get_items();
foreach ($order->get_items() as $key => $lineItem) {
    $_product = new WC_Product($lineItem[
        'product_id']);
    $invoiceXML .= '
        <PolFak>
        <typCenyDphK>typCeny.sDph</
            typCenyDphK>
        <cenik>code:P ' . $lineItem[
            'product_id'] . '</cenik>
        <název>' . $lineItem['name'] . '</
            název>
        <typPolozkyK>typPolozky.katalog</
            typPolozkyK>
        <sklad>code:SKLAD</sklad>
        <cenaMj>' . $_product->get_price() .
            '</cenaMj>
        <mnozMj>' . $lineItem['qty'] . '</
            mnozMj>
        <sumCelkem>' . $lineItem['qty'] *
            $_product->get_price() . '</
            sumCelkem>
        </PolFak>';
    }
    $invoiceXML .= '
        </polozkyFaktury>
        <!-- údaje o financích -->
        <stavUhrK>stavUhr.uhrazenoRucne</stavUhrK>
        <sumCelkZakl>' . $order->order_total . '</
            sumCelkZakl>
        </faktura-vydana>
        </winstrom>';

```

```

curl_setopt($ch, CURLOPT_POSTFIELDS, $invoiceXML);
//chybový výstup
$output = curl_exec($ch);
$info = curl_getinfo($ch);

if (curl_getinfo($ch, CURLINFO_HTTP_CODE) != 200 &&
curl_getinfo($ch, CURLINFO_HTTP_CODE) != 201) {
    $order->add_order_note(' Při
    přenosu faktury do systému došlo k chybě:' .
    $output);
} else {
    $order->add_order_note('Objednávka číslo ' .
    $order_id . ' byla vyfakturována.');
```

```

}
// close curl resource to free up system resources
curl_close($ch);
}

function get_invoice_pdf($order_id) {
    include_once(plugin_dir_path(__FILE__) . 'connection-
    initiation.php');
    $code = 'code:VF1-' . $order_id;
    curl_setopt($ch, CURLOPT_URL, $host . "/c/" . $firma
    . "/faktura-vydana/" . $code);
    // Nastavení samotné operace
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "GET");
    //chybový výstup
    $output = curl_exec($ch);
    $info = curl_getinfo($ch);
    $trim = substr($output, strpos($output, "/faktura-vydana/"
    ") + 1);
    $invoice_id = substr($trim, strpos($trim, "/" ) + 1);

    if (curl_getinfo($ch, CURLINFO_HTTP_CODE) != 200 &&
curl_getinfo($ch, CURLINFO_HTTP_CODE) != 201) {
        return 'Došlo k chybě při předání faktury. Za
```

```

        tuto nepříjemnost se omlouváme. Prosíme
        kontaktujte eshop.';
    } else {
        return $host . "/c/" . $firma . "/faktura-
            vydana/" . trim($invoice_id, "x") . ".pdf"
            ;
    }
}

function stock_synchronize(){
    include_once(plugin_dir_path(__FILE__) . 'connection-
        iniciation.php');
    curl_setopt($ch, CURLOPT_URL, $host . "/c/" . $firma
        . "/cenik/(exportNaEshop=true).xml?detail=custom:
        kod,sumStavMj&limit=0");
    // Nastavení samotné operace
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "GET");
    // chybový výstup
    $output = curl_exec($ch);
    // output přijata data z flexibee
    $flexibeeProducts = new SimpleXMLElement($output) or
        die("Error: Cannot create object");
    global $woocommerce;
    $args = array(
        'post_type' => 'product',
        'post_status' => 'publish',
        'posts_per_page' => -1
    );
    $loop = new WP_Query($args);
    while ($loop->have_posts()) : $loop->the_post();
        $product_id = $loop->post->ID;
        $code = 'P' . $product_id;
        $_product = new WC_Product($product_id);
        // průchod xml strukturou
        for ($i = 0; $i <= $flexibeeProducts->count();
            $i++) {
            if (($flexibeeProducts->cenik[$i]->

```

```

                kod == $code)) {
                    $_product->set_stock(
                        $flexibeeProducts->cenik[
                            $i]->sumStavMj, 'set');
                    break;
                }
            }
        endwhile;
    }

```

## E Kód navázání spojení

```

<?php
// konfigurace serveru
$host = "http://demo.flexibee.eu";
$firma = "demo";

$ch = curl_init(); // create curl resource
curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE); // return
    content as a string from curl_exec
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, TRUE); // follow
    redirects (compatibility for future changes in FlexiBee)
curl_setopt($ch, CURLOPT_HTTPAUTH, TRUE); // HTTP
    authentication
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, FALSE); // FlexiBee
    by default uses Self-Signed certificates
// curl_setopt($ch, CURLOPT_VERBOSE, TRUE); // For debugging
curl_setopt($ch, CURLOPT_USERPWD, "winstrom:winstrom"); //
    set username and password

```

## F Kód manuální změny objednávky ze stavu zpracovává se na stav ready

```

<?php
$order_ready_check = true;
global $woocommerce;

```

```

$args = array(
    'post_type' => 'shop_order',
    'post_status' => 'publish',
    'posts_per_page' => -1,
    'tax_query' => array( array(
        'taxonomy' => 'shop_order_status',
        'field' => 'slug',
        'terms' => array('processing')
    ))
);
echo "<h3>" . $resultText . "</h3>";
$loop = new WP_Query($args);
while ($loop->have_posts()) : $loop->the_post();
    $order_id = $loop->post->ID;
    $order = new WC_Order($order_id);
    $order_item = $order->get_items();
    foreach ($order->get_items() as $key => $lineItem) {
        $_product = new WC_Product($lineItem[ '
            product_id' ]);
        if (($_product->get_total_stock()) <= 0) {
            $order_ready_check = false;
        }
    }
    if ($order_ready_check == true) {
        $order->update_status('wc-ready', '');
        echo "Objednávka " . $order_id . " je
            připravena k vyzvednutí.</br>";
    } else {
        $order->update_status('wc-on-hold', '');
        echo "Jeden z produktů u objednávky " .
            $order_id . " není skladem.</br>";
    }
    $order_ready_check = true;
endwhile;

wp_reset_postdata(); ?>

```

## G Kód manuální synchronizace zboží s výpisem

```
<?php
global $woocommerce;
$args = array(
    'post_type' => 'product',
    'post_status' => 'publish',
    'posts_per_page' => -1,
);

$loop = new WP_Query($args);

if (get_locale() == 'cs_CZ') {
    echo "<h2>Stav zásob po synchronizaci</h2>";
} else {
    echo "<h2>Products stock state after synchronization
    </h2>";
}

include_once(plugin_dir_path(__FILE__) . 'connection-
    initiation.php');

curl_setopt($ch, CURLOPT_URL, $host . "/c/" . $firma . "/"
    cenik/(exportNaEshop=true).xml?detail=custom:kod,sumStavMj
    &limit=0");
// Nastavení samotné operace
curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "GET");
// chybový výstup
$output = curl_exec($ch);
// output přijata data z flexibee
$flexibeeProducts = new SimpleXMLElement($output) or die("
    Error: Cannot create object");
$info = curl_getinfo($ch);
if (curl_getinfo($ch, CURLINFO_HTTP_CODE) != 200 &&
    curl_getinfo($ch, CURLINFO_HTTP_CODE) != 201) {
    echo 'Došlo k chybě.</br>';
}
```



```

echo "</br><table><thead><tr>";
if ( get_locale () == 'cs_CZ' ) {
    echo "<th>Id produktu</th><th>Název produktu</th><th>
    Stav zásob před</th><th>Stav zásob po</th><th>Cena
    za jednotku</th></tr></thead><tbody>";
} else {
    echo "<th>Product id</th><th>Product name</th><th>
    Stock before</th><th>Stock after</th><th>Price per
    unit</th></tr></thead><tbody>";
}

while ( $loop->have_posts () ) : $loop->the_post () ;
    $product_id = $loop->post->ID;
    $code = 'P' . $product_id;
    $_product = new WC_Product ( $product_id );
    echo '<tr><td>' . $product_id . '</td>';
    echo '<td>' . $_product->get_title () . '</td>';
    echo '<td>' . $_product->get_stock_quantity () . '</td
    >';
    //průchod xml strukturou
    for ( $i = 0; $i <= $flexibeeProducts->count (); $i++ ) {
        if ( ( $flexibeeProducts->cenik [ $i ]->kod ==
        $code ) ) {
            $_product->set_stock (
                $flexibeeProducts->cenik [ $i ]->
                sumStavMj , 'set' );
            break ;
        }
    }
    echo '<td>' . $_product->get_stock_quantity () . '</td
    >';
    echo '<td>' . $_product->get_price () . '</td></tr>';
endwhile ;
echo '</tbody></table></br>';
wp_reset_query () ;

```