

# **Webový ekonomický informační systém**

Web-based Economic Information System

Bc. Jiří Kunčar

---

Diplomová práce  
2013



Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

# **ZADÁNÍ DIPLOMOVÉ PRÁCE**

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jiří Kunčar**  
Osobní číslo: **A11717**  
Studijní program: **N3902 Inženýrská informatika**  
Studijní obor: **Informační technologie**  
Forma studia: **kombinovaná**

Téma práce: **Webový ekonomický informační systém**

## **Zásady pro vypracování:**

- 1. Seznamte se s technologiemi PHP, MySQL, Javascript, jQuery, SSL, Apache, zabezpečení aplikací apod.**
- 2. Analyzujte potřeby ekonomických subjektů, zejména evidenci přijatých a vydaných dokladů, pohledávek, řízení toku peněz a propojení s bankovními účty. Zaměřte se na automatizaci některých procesů v běžném provozu.**
- 3. Navrhněte webový ekonomický informační systém s ohledem na uvedenou analýzu.**
- 4. Realizujte uvedený návrh. Implementujte některá rozšíření dle potřeb konkrétních ekonomických subjektů.**
- 5. Zabezpečte aplikaci proti útokům.**
- 6. Otestujte systém v reálném provozu skutečného ekonomického subjektu.**
- 7. Zhodnoťte přínos práce a navrhněte možný rozvoj do budoucna.**

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **DOSTÁLEK, Libor. Velký průvodce protokoly TCP/IP a systémem DNS. Praha: Computer Press, 2000. ISBN 80-722-6323-4.**
2. **POŠMURA, Vladimír. Apache: Příručka správce WWW serveru. Praha: Computer Press, 2002. ISBN 80-722-6696-9.**
3. **LEE, James a Brent WARE. Open source: vývoj webových aplikací Linux, Apache, MySQL, Perl a PHP. Překlad Jan Gregor. Brno: Mobil Media, 2003. ISBN 80-865-9343-6.**
4. **SCHLOSSNAGLE, George. Pokročilé programování v PHP 5. Brno: Zoner Press, 2004. ISBN 80-868-1514-5.**
5. **GILMORE, Jason W. Velká kniha PHP a MySQL 5: kompendium znalostí pro začátečníky i profesionály. Brno: Zoner Press, 2007. ISBN 80-868-1553-6.**
6. **AJAX a PHP: tvoříme interaktivní webové aplikace profesionálně. Brno: Zoner Press, 2006. ISBN 80-868-1547-1.**

Vedoucí diplomové práce:

**Ing. Petr Šilhavý, Ph.D.**

Ústav počítačových a komunikačních systémů

Datum zadání diplomové práce:

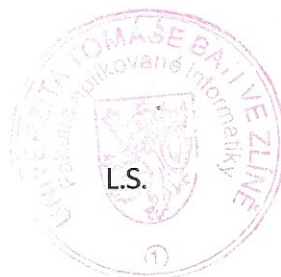
**22. února 2013**

Termín odevzdání diplomové práce:

**22. května 2013**

Ve Zlíně dne 22. února 2013

prof. Ing. Vladimír Vašek, CSc.  
*děkan*



doc. Mgr. Roman Jašek, Ph.D.  
*ředitel ústavu*

## ABSTRAKT

Diplomová práce se zabývá návrhem a implementací webového ekonomického informačního systému. Cílem je navrhnout univerzální komplexní řešení pro podnikatelské subjekty a jejich provoz, zejména usnadnit evidenci vydaných a přijatých dokladů, pohledávek a závazků, řízení toku financí, zajistit automatizaci některých procesů v běžném provozu a poskytnout statistiky hospodaření. Po úvodním seznámení se s teoretickou analýzou technologií a potřeb ekonomických subjektů se práce věnuje návrhu systému a popisuje implementaci řešení. V poslední části se práce zaměřuje na testování systému v reálném provozu a je diskutován možný budoucí rozvoj systému.

Klíčová slova: informační systém, webové technologie, podnikatelský subjekt, daňová evidence, účetnictví, daňové doklady, PHP, MySQL, Apache

## ABSTRACT

This thesis describes the design and implementation of web-based economic information system. The aim is to design a universal comprehensive solution for companies and their operations, especially to facilitate the records of issued and received invoices, accounts receivable and payable, the control of cash flow, ensure automation of some processes in usual operations and to provide economic statistics. After an initial introduction to the theoretical analysis of technologies and the needs of companies, the thesis focuses on the system design and describes the implementation of solution. The last part of the thesis focuses on testing of the system in real operations and a possible future development of the system is discussed.

Keywords: Information System, web technologies, business entity, tax records, bookkeeping, invoices, PHP, My SQL, Apache

Rád bych poděkoval panu Ing. Petru Šilhavému, Ph.D. za jeho ochotu a trpělivost při vedení mé diplomové práce. Dále bych rád poděkoval svým rodičům a prarodičům za jejich podporu v průběhu celého studia.

**Prohlašuji, že**

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

**Prohlašuji,**

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně 29. 4. 2013

.....  
podpis diplomanta

**OBSAH**

<b>ÚVOD.....</b>	<b>10</b>
<b>I TEORETICKÁ ČÁST .....</b>	<b>12</b>
<b>1 ANALÝZA TECHNOLOGIÍ.....</b>	<b>13</b>
1.1 SÍŤOVÉ PROTOKOLY .....	13
1.1.1 Protokoly TCP/IP .....	13
1.1.2 HTTP.....	14
1.1.3 HTTPS, SSL.....	15
1.2 URL.....	16
1.3 HTML.....	16
1.4 XML .....	17
1.5 CSS.....	18
1.6 JAVASCRIPT .....	18
1.7 JQUERY.....	19
1.8 JSON .....	19
1.9 AJAX.....	20
1.10 PHP.....	20
1.11 RELAČNÍ DATABÁZOVÉ SYSTÉMY .....	21
1.11.1 MySQL.....	21
1.11.2 SQL .....	22
1.12 DIBI – PHP KNIHOVNA .....	22
1.13 APACHE.....	23
1.14 CRON.....	24
1.15 MEMCACHED.....	25
1.16 WEB API.....	25
1.17 SMARTY .....	26
1.18 PDF.....	26
<b>2 POTŘEBY A POVINNOSTI EKONOMICKÝCH SUBJEKTŮ .....</b>	<b>27</b>
2.1 VEDENÍ EVIDENCE PODNIKATELSKÉ ČINNOSTI .....	27
2.2 DAŇOVÁ EVIDENCE .....	27
2.3 ÚČETNICTVÍ .....	28
2.4 ÚČETNÍ DOKLADY .....	29
2.4.1 Náležitosti účetních dokladů .....	29
2.4.2 Vyhotovování účetních dokladů.....	30
2.4.3 Druhy účetních dokladů .....	30
2.5 DAŇ Z PŘIDANÉ HODNOTY.....	31

2.5.1	Plátce DPH .....	31
2.5.2	Sazby DPH .....	32
2.5.3	Daňové doklady .....	32
2.5.4	Dodání a pořízení zboží v rámci Evropské unie .....	33
2.5.5	Ověření platnosti DIČ .....	33
2.6	MĚNY .....	34
2.7	MĚNOVÉ KURZY .....	35
2.8	BANKOVNÍ ÚČTY .....	35
2.9	ABO FORMÁT .....	36
<b>3</b>	<b>REŠERŠE EXISTUJÍCÍCH ŘEŠENÍ .....</b>	<b>37</b>
3.1	POHODA ONLINE .....	37
3.2	IDOKLAD .....	37
3.3	FLEXIBEE ONLINE .....	38
3.4	FAKTUROID .....	38
3.5	QFAKTURY .....	38
3.6	IÚČTO .....	38
<b>II</b>	<b>PRAKTICKÁ ČÁST .....</b>	<b>39</b>
<b>4</b>	<b>NÁVRH SYSTÉMU .....</b>	<b>40</b>
4.1	KONCEPTUÁLNÍ POPIS SYSTÉMU .....	40
4.2	ANALÝZA POŽADAVKŮ .....	41
4.2.1	Funkční požadavky .....	41
4.2.2	Nefunkční požadavky .....	48
4.3	DEFINICE AKTÉRŮ SYSTÉMU .....	48
4.3.1	Zákazník .....	49
4.3.2	Administrátor .....	49
4.3.3	Dodavatel .....	50
4.4	MODEL PŘÍPADŮ UŽITÍ .....	50
4.5	NÁVRH UŽIVATELSKÉHO ROZHRANÍ .....	52
4.6	DEFINICE ANALYTICKÝCH TŘÍD .....	54
4.7	NÁVRH ZABEZPEČENÍ SYSTÉMU .....	56
4.7.1	Zabezpečení síťového provozu .....	56
4.7.2	Zabezpečení klientské části .....	57
4.7.3	Zabezpečení datové části .....	57
<b>5</b>	<b>IMPLEMENTACE SYSTÉMU .....</b>	<b>59</b>
5.1	SPECIFIKACE TECHNOLOGIÍ .....	59
5.2	ARCHITEKTURA MVC .....	59
5.2.1	Model .....	60
5.2.2	View .....	63
5.2.3	Controller .....	67



5.3	ZABEZPEČENÍ SYSTÉMU PROTI ÚTOKŮM .....	67
5.3.1	Zabezpečení síťového provozu .....	67
5.3.2	Zabezpečení proti XSS .....	68
5.3.3	Zabezpečení proti SQL injection .....	68
5.4	PARALELNÍ PROVOZ VÍCE SYSTÉMŮ .....	68
<b>6</b>	<b>TESTOVÁNÍ SYSTÉMU V REÁLNÉM PROVOZU .....</b>	<b>72</b>
6.1	ŽIVNOSTNÍK – NEPLÁTCE DPH .....	72
6.1.1	Představení subjektu .....	72
6.1.2	Výsledky testování .....	73
6.2	OBCHODNÍ SPOLEČNOST S RUČENÍM OMEZENÝM – PLÁTCE DPH .....	74
6.2.1	Představení subjektu .....	74
6.2.2	Výsledky testování .....	75
<b>ZÁVĚR .....</b>		<b>78</b>
<b>ZÁVĚR V ANGLIČTINĚ .....</b>		<b>79</b>
<b>SEZNAM POUŽITÉ LITERATURY .....</b>		<b>80</b>
<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK .....</b>		<b>83</b>
<b>SEZNAM OBRÁZKŮ .....</b>		<b>85</b>
<b>SEZNAM TABULEK .....</b>		<b>86</b>
<b>SEZNAM PŘÍLOH .....</b>		<b>87</b>

## ÚVOD

Webový prohlížeč je dnes jedním z nejrozšířenějších programů, který je uživateli intuitivně každodenně využíván na různých platformách. Moderní doba s sebou přináší stále nová zařízení, ve kterých je právě webový prohlížeč obsažen v nejrůznějších podobách. Webové aplikace jsou tedy podporovány nejen v osobních počítačích a notebookech, lze je používat i na mobilních telefonech, tabletech, PDA nebo dokonce v televizorech. Rozšířená dostupnost internetu a webových prohlížečů usnadňuje komunikaci a umožňuje rychlé sdílení dat, internet se tak stal nedílnou součástí každodenního života mnoha lidí po celém světě.

Využitelnost internetu v podnikání je v 21. století beze sporu denní rutinou téměř každého ekonomického subjektu, ať už se jedná o drobného živnostníka nebo velkou obchodní společnost. Většina podnikatelských subjektů v ČR je povinna vést evidenci podnikatelské činnosti (účetnictví nebo daňovou evidenci) zachycující jejich hospodářskou činnost, s čímž je spojena spousta každodenních úkonů potřebných k běžnému provozu firem. Cílem webového ekonomického informačního systému je využití internetu a širokého zastoupení webových prohlížečů na odlišných zařízeních k získání moderního ekonomického softwaru.

Ekonomické aktivity firem vyžadují samozřejmě vysoké zabezpečení. K základním požadavkům na webový ekonomický systém tak beze sporu patří mnoho bezpečnostních opatření ze strany poskytovatele webových služeb. Nutností se stává ochrana před nežádoucím přístupem k uživatelským a firemním datům.

Cílem této práce je seznámení se s technologiemi, které budou následně využity k návrhu a implementaci webového ekonomického informačního systému. Nedílnou součástí práce je analýza potřeb ekonomických subjektů, zejména evidence přijatých a vydaných dokladů, pohledávek, řízení toku peněz a propojení s bankovními účty. Práce je mimo jiné zaměřena na automatizaci některých procesů v běžném provozu ekonomických subjektů.

Návrh systému s ohledem na uvedenou analýzu klade důraz zejména na zabezpečení víceuživatelského webového ekonomického IS s využitím Apache, PHP a MySQL a dalších technologií. Uživatelé systému mají různá oprávnění a je podporován paralelní provoz více systémů na oddělených databázích v rámci jedné aplikace, avšak na různých doménách pro jednotlivé ekonomické subjekty.

Samotná realizace na základě návrhu systému zahrnuje i některá rozšíření dle potřeb konkrétních ekonomických subjektů. Implementace ekonomického informačního systému se zaměřuje i na zabezpečení aplikace proti útokům.

Webový ekonomický informační systém je následně testován v reálném provozu skutečného ekonomického subjektu, díky tomu práce obsahuje reálné ekonomické výsledky a přínosy při užití informačního systému v běžném provozu.

V závěru práce jsou diskutovány výsledky práce a je navržen možný rozvoj systému do budoucna.

## **I. TEORETICKÁ ČÁST**

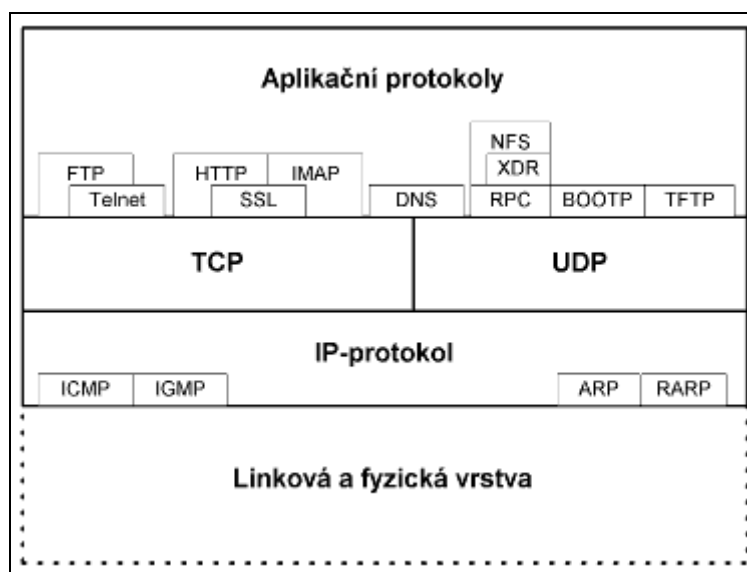
# 1 ANALÝZA TECHNOLOGIÍ

## 1.1 Síťové protokoly

Počítače v počítačových sítích používají pro vzájemnou komunikaci síťové protokoly. Síťový protokol je norma [1], podle které se uskutečňuje přenos dat a elektronická komunikace. V internetu se používají normy nazývané RFC (*Request For Comments*), které se číslují průběžně od 1. Vzhledem ke složitosti problému je síťová komunikace rozdělena do tzv. vrstev [2], které znázorňují hierarchii činností. Výměna informací mezi vrstvami je exaktně definována [1], každá vrstva využívá služeb vrstvy nižší a poskytuje své služby vrstvě vyšší. Síťových protokolů existuje mnoho, v internetu se používají síťové protokoly TCP/IP.

### 1.1.1 Protokoly TCP/IP

Rodina protokolů TCP/IP (*Transmission Control Protocol / Internet Protocol*) se až na výjimky nezabývá fyzickou a linkovou vrstvou [1]. Obrázek (Obr. 1) znázorňuje vazby mezi některými protokoly z rodiny protokolů TCP/IP. Internet Protocol (IP) prakticky odpovídá síťové vrstvě. IP přenáší tzv. IP datagramy mezi vzdálenými počítači. Každý IP datagram ve svém záhlaví nese adresu příjemce [2], což je úplná směrovací informace. Síť tedy může přenášet každý IP datagram samostatně a IP datagramy mohou k adresátovi dorazit v jiném pořadí, než v jakém byly odeslány.



Obr. 1. Některé protokoly z rodiny protokolů TCP/IP [1]

Protokol TCP odpovídá transportní vrstvě. Protokol dopravuje data pomocí TCP segmentů, které jsou adresovány jednotlivým aplikacím [2]. Protokol TCP je tzv. spojovanou službou, tj. příjemce potvrzuje přijímaná data. V případě ztráty dat si příjemce vyžádá zopakování přenosu.

### 1.1.2 HTTP

HTTP (*Hypertext Transfer Protocol*) je protokol aplikační vrstvy, který byl původně určený [3] pouze pro výměnu hypertextových dokumentů ve formátu HTML. HTTP protokol verze 1.0 je definován v RFC 1945 a verze 1.1 protokolu je definována v RFC 2616 [3]. Protokol funguje způsobem dotaz-odpověď. Klient odešle serveru dotaz ve formě datové zprávy, obsahujícího jednoznačné označení požadovaného dokumentu, informace o schopnostech prohlížeče klienta apod. Server po přijetí a zpracování požadavku odpoví pomocí zprávy obsahující hlavičku (*header*) popisující výsledek dotazu, za kterou zpravidla následují data požadovaného dokumentu.

HTTP je bezstavový protokol [3] - protokol neumí uchovávat stav komunikace. Tato vlastnost není vhodná pro implementaci složitějších informačních systémů (např. potřeba uchovávat informaci o identitě uživatele). K tomuto účelu byl protokol HTTP rozšířen o tzv. HTTP cookies, které umožňují serveru uchovávat informace na klientské stanici.

HTTP definuje několik dotazovacích metod [3], nejdůležitější z nich jsou:

- GET – Nejčastěji používaný HTTP požadavek, který slouží k získání dokumentu.
- HEAD - Podobné metodě GET, ale už nepředává data. Poskytne pouze metadata o požadovaném dokumentu.
- POST - Odesílá uživatelská data na server, např. odesláním formuláře na webu. Oproti metodě GET umožňuje zaslání většího množství dat.
- OPTIONS - Dotaz na server na jeho podporované metody.
- CONNECT - Používá se při průchodu skrze proxy pro ustanovení kanálu SSL.

Formát HTTP odpovědi ve formě datové zprávy je rozdělen na hlavičku a obsah. První řádek hlavičky se nazývá „status line“ a skládá se ze tří částí [3]. První část identifikuje verzi HTTP protokolu, druhá část obsahuje „status code“ (číslo značící výsledek

požadované operace) a v třetí části je slovní reprezentace stavového kódu. Stavový kód je trojciferné číslo, kde nejvyšší řád rozděluje stavové kódy do 5 hlavních skupin [3]. Tabulka (Tab. 1) popisuje významy stavových kódu v těchto hlavních skupinách.

Skupina stavových kódů	Účel stavových kódů
1xx	Kódy informačního charakteru.
2xx	Kódy označující úspěšné vykonání požadavku.
3xx	Od klienta je požadovaná další akce, např. přesměrování.
4xx	Během zpracování požadavku došlo k nějaké chybě.
5xx	Došlo k chybě na straně serveru.

*Tab. 1. Hlavní skupiny stavových kódů HTTP odpovědí*

### 1.1.3 HTTPS, SSL

HTTPS (*Hypertext Transfer Protocol Secure*) je nadstavba síťového protokolu HTTP [3]. Samotný přenos dat je u protokolu HTTP zcela nezabezpečený před případným odposloucháváním síťové komunikace, proto je v případě potřeby zabezpečení komunikace nutné využít právě nadstavbu HTTPS, která nejenže zabezpečí spojení mezi klientem a serverem před odposloucháváním, ale umožňuje rovněž ověřit identitu protistrany a zabezpečit klienta proti podvržení dat. HTTPS používá právě protokol HTTP (standardní port na straně serveru je 443) a přenášená data jsou šifrována pomocí SSL nebo TLS.

SSL (*Secure Sockets Layer*) je vrstva vložená mezi transportní a aplikační vrstvu [3], která zabezpečuje komunikaci autentizací obou stran a šifrováním dat. Každá z komunikujících stran má vždy dva šifrovací klíče - veřejný a soukromý klíč. Data jsou zašifrována na základě obou klíčů a veřejný klíč je možné zveřejnit, jelikož zašifrovaná data lze dešifrovat pouze pomocí soukromého klíče.

Na začátku komunikace si klient se serverem vymění jejich veřejné klíče. Následně odesílaná data jsou zašifrována veřejným klíčem protistrany, tím se zabezpečí, že data může dešifrovat pouze důvěryhodný protějšek se svým soukromým klíčem. Do celého procesu vstupuje ještě další element a tím je certifikační autorita, která označuje veřejný klíč svým otiskem, aby byla maximálně zabezpečena důvěryhodnost serveru. Záleží pak už

pouze na protistraně, zda certifikační autoritě důvěřuje nebo nikoliv. Certifikační autoritou totiž může být i sám server (využíváno nejčastěji v intranetu).

## 1.2 URL

URL (*Uniform Resource Locator*) je řetězec znaků, který jednoznačně specifikuje umístění různých typů zdrojů dat v rámci sítě [2], např. v rámci internetu. Jedná se tedy o jedinečnou adresu dokumentu v síti.

Syntaxe URL je následující:

[PROTOKOL] : // [SERVER] : [PORT] / [UMISTENI] ? [DATA]

Některá pole jsou však nepovinná, protože v některých případech nemají význam nebo se očekává výchozí předdefinovaná hodnota. Např. pro protokol HTTP je výchozí port 80. Význam jednotlivých polí je objasněn v následující tabulce (Tab. 2).

Pole ze syntaxe URL	Význam
[PROTOKOL]	Protokol pro připojení k cílovému počítači (HTTP, HTTPS, FTP, atd.).
[SERVER]	Počítač, ke kterému se chceme připojit (doménové jméno nebo IP adresa).
[PORT]	Port připojení (např. pro HTTP implicitně 80).
[UMISTENI]	Cesta ke konkrétnímu souboru.
[DATA]	Např. formulářová data uvedená za otazníkem ve tvaru „název=hodnota“ oddělené znakem „&“.

Tab. 2. Význam polí v syntaxi URL

## 1.3 HTML

HTML (*HyperText Markup Language*) je značkovací jazyk pro hypertext [4]. Předchůdcem HTML je rozsáhlý univerzální značkovací jazyk SGML (*Standard Generalized Markup Language*) [5]. HTML používá přesně definovanou syntaxi příkazů, které říkájí webovému prohlížeči, jak má zobrazit obsah daného dokumentu, tedy jak ho naformátovat a jak pracovat s textem, obrázky a dalšími podporovanými formáty. Samotné příkazy (tzv. *tagy*) prohlížeč nezobrazuje. Mimo formátování umožňuje jazyk i definici tzv.



hypertextových odkazů, které propojují dokument s jinými dokumenty nebo elementy sítě WWW.

Vývoj HTML byl samozřejmě ovlivněn vývojem webových prohlížečů, které zpětně ovlivňovaly definici jazyka. Vývoj jazyka HTML byl původně ukončen verzí 4.01 [4]. Tento standard je dnešními prohlížeči prakticky plně podporován, ale existují některá nestandardní rozšíření, která vedou k nekompatibilitě mezi jednotlivými prohlížeči. Dle W3C měl další vývoj psaní webových dokumentů patřit jazyku XHTML - následníkovi HTML, využívajícímu univerzální jazyk XML [5]. 7. března 2007 W3C založilo novou pracovní skupinu HTML, jejímž cílem bylo uvolnit specifikaci nové verze HTML. Tato verze nese označení HTML5.

Jak již bylo zmíněno, dokument HTML je tvořen příkazy (tagy), které definují vzhled a strukturu dokumentu, a textem, který představuje samotný obsah dokumentu. Dokument HTML je ohraničen párovým příkazem „<html>“ a obsahuje hlavičku „<head>“ a tělo „<body>“. V hlavičce se uvádí např. název HTML dokumentu a lze zde uvést další doplňující informace [4], které prohlížeč používá při zobrazení. Tělo obsahuje již samotný obsah dokumentu, společně s dalšími příkazy, které upřesňují formát zobrazení a sémantiku obsahu. Dalšími příkazy lze do dokumentu vkládat např. obrázky, objekty, tabulky, seznamy, formuláře či odkazy na jiné dokumenty.

## 1.4 XML

XML (*Extensible Markup Language*) neboli rozšiřitelný značkovací jazyk je podobně jako jazyk HTML odvozen od jazyka SGML [5]. Na rozdíl od HTML umožňuje XML vytvářet vlastní elementy, atributy a strukturu dokumentu, lze jej tedy použít k popsání jakéhokoli typu informace. Dokumenty XML mohou být zpracovány a zobrazovány v různých typech aplikací, nejenom ve webovém prohlížeči. Hlavním smyslem jazyka [6] je poskytovat univerzální formát pro ukládání, výměnu a zobrazování informací na webu.

Jazyk XML poskytuje pouze informace o samotných datech, o jejich struktuře a uspořádání. Samotné zobrazení dat nechává na softwaru, který je analyzuje a zpracovává. Definice typu dokumentu (*DTD*) jazyka XML se skládá pouze z prázdné kostry syntaxe. Při vytváření dokumentu XML jsou tvořeny vlastní elementy s libovolnými názvy, čemuž

odpovídá pojem „rozšiřitelný“ (extensible) v názvu XML. Díky tomu je možné jazyk použít k popsání skutečně libovolného typu dokumentu. [6]

Následuje jednoduchá ukázka XML dokumentu, který popisuje příklad uložení informací o studentech školy (v ukázce jsou zobrazeni pouze 2 studenti, v reálném dokumentu by jich mohlo být samozřejmě více):

```
1 <?xml version="1.0"?>
2
3 <SEZNAM_STUDENTU>
4     <STUDENT>
5         <JMENO>Petr Novák</JMENO>
6         <TRIDA>8.A</TRIDA>
7     </STUDENT>
8     <STUDENT>
9         <JMENO>Pavel Milostný</JMENO>
10        <TRIDA>6.B</TRIDA>
11    </STUDENT>
12 </SEZNAM_STUDENTU>
```

## 1.5 CSS

Kaskádové styly se zkratkou CSS (*Cascading Style Sheets*) je jazyk, pomocí kterého lze popsat způsob zobrazení dokumentů napsaných v jazycích HTML, XHTML nebo XML. Hlavním smyslem je oddělení vzhledu dokumentu od jeho vlastní struktury a obsahu [5].

Do HTML dokumentů lze kaskádové styly vkládat přímo párovou značkou „<style>“ v hlavičce dokumentu, lze je vkládat jako parametr HTML značek s názvem „style“, kde hodnotou parametru jsou právě samotné styly oddělené středníkem, nejčastěji jsou však styly uloženy v externím souboru a do samotného dokumentu je externí soubor se styly vkládán značkou „<link>“.

## 1.6 JavaScript

JavaScript je interpretovaný programovací jazyk, který je nejčastěji vykonáván ve webovém prohlížeči [7]. HTML stránky tak mohou obsahovat tuto „klientskou“ verzi JavaScriptu, která umožňuje webovým stránkám obsahovat spustitelný obsah - dynamické

programy, které např. komunikují s uživatelem, řídí prohlížeč nebo dynamicky upravují obsah HTML.

Rané verze JavaScriptu měly problémy se zabezpečením [7], jelikož některé starší verze prohlížečů obsahovaly nebezpečné interprety jazyka. V současné době jsou však u většiny novějších prohlížečů všechny známé bezpečnostní chyby opraveny.

Podobně jako CSS lze do HTML dokumentů JavaScript vkládat více možnostmi - párovou značkou „<script>” přímo v HTML dokumentu, lze ho vkládat jako parametry HTML značek s různými názvy označující typ akce, kdy se má kód vykonat (např. „onclick“ při kliknutí na prvek nebo „onmouseover“ při najetí myši na prvek), nejčastěji je ale kód uložen v externím souboru a do samotného dokumentu je vkládán značkou „<script>“ s parametrem „src“, jehož hodnota obsahuje adresu externího souboru.

## 1.7 JQuery

JQuery je javascriptová knihovna [8] (*framework*), která je zaměřena na interakci mezi JavaScriptem a HTML. Jedná se o svobodný a otevřený software pod licencemi GPL (*General Public License*) a MIT (*Massachusetts Institute of Technology*). Použitím tohoto frameworku je mimo jiné zabezpečena funkčnost zdrojového kódu v různých interpretech JavaScriptu ve webových prohlížečích většiny výrobců, jelikož řeší jejich omezení a odlišnosti.

## 1.8 JSON

JSON (*JavaScript Object Notation*) je odlehčený datový formát, který je zcela nezávislý na platformě [9]. Primárně je tento datový formát určen pro snadný přenos dat. Formát je jednoduše čitelný člověkem, navíc je snadno strojově analyzovatelný i generovatelný. Jak již název napovídá, je založen na podmnožině programovacího jazyka JavaScript. JSON je textový formát, který je založen na dvou strukturách: kolekce párů název/hodnota a tříděný seznam hodnot. Jedná se o naprosto univerzální datové struktury, které v nějaké formě podporují v podstatě všechny dnešní moderní programovací jazyky [9]. Vstupem dat pro JSON ale může být libovolná datová struktura (řetězec, objekt, číslo, pole), výstupem je právě textový řetězec využívající uložení pomocí zmíněných datových struktur.

V následující ukázce je opět demonstrováno uložení informací o studentech školy, tentokrát ve formátu JSON:

```
1 {"seznam_studentu": {  
2     "student": [  
3         {  
4             "jmeno": "Petr Novák",  
5             "trida": "8.A"  
6         },  
7         {  
8             "jmeno": "Pavel Milostný",  
9             "trida": "6.B"  
10        }  
11    ]  
12 }}
```

## 1.9 AJAX

AJAX (*Asynchronous JavaScript and XML*) je označení pro techniky vývoje interaktivních webových aplikací [9], které mění obsah svých stránek získáváním dat ze serveru bez nutnosti znovunačtení kompletní webové stránky. Jak už název napovídá, AJAX je založen na JavaScriptu, ve skutečnosti se ale nejedná pouze o jednu určitou technologii, nýbrž je to pojem označující několik technologií dohromady. Jako komunikační prostředek je používán XHR [9] (*XMLHttpRequest*), což je rozhraní (*interface*) umožňující webovým aplikacím komunikaci mezi klientem a serverem prostřednictvím síťového protokolu HTTP, resp. HTTPS. Důležitým omezením webových prohlížečů je omezení použití AJAX pro získání dat z různých domén, čili je nutné požadované XHR dotazy odesílat a přijímat v rámci stejné domény a v rámci stejného portu, jako má aktuální webová stránka.

## 1.10 PHP

Hypertextový preprocesor PHP je velice rozšířený skriptovací programovací jazyk, který je zejména využíván k vytváření dynamických webových stránek a aplikací [10]. PHP lze aplikovat všude tam, kde je nainstalován interpret jazyka PHP, např. jako rozšíření webového serveru. Skripty jsou tak vykonávány na straně serveru a ke klientovi se přenášejí až výstup z jejich zpracování. PHP podporuje celou řadu knihoven. Lze jej tak použít např. pro práci se soubory, pro komunikaci s dalšími servery, ke zpracování grafiky, komprimaci

dat a taky pro přístup k většině databázových systémů. PHP tedy může tvořit rozhraní mezi webovým serverem a databázovým systémem, např. MySQL, Oracle, PostgreSQL atd.

PHP je interpretovaný jazyk, tedy až do okamžiku vykonání je zdrojový kód uložen v nekompilovaném tvaru [10]. Interpret jazyka PHP předkládá zdrojový kód do strojového kódu až v okamžiku spuštění skriptu, což je samozřejmě pomalejší než v případě kompilovaného kódu ve spustitelném tvaru. Na druhou stranu interpretovaný jazyk umožňuje snazší úpravy, jelikož není nutné po každé úpravě program znovu překládat, aby byla změna vykonána. Další výhodou je snadná přenositelnost mezi různými operačními systémy. Syntaxe jazyka vychází z jazyka C a je mu tedy velmi podobná. Pro vykonání zdrojového kódu PHP musí být webový server nakonfigurován tak, aby požadavky na dokumenty s určenou příponou (např. „.php“) byly nejprve předány preprocesoru PHP.

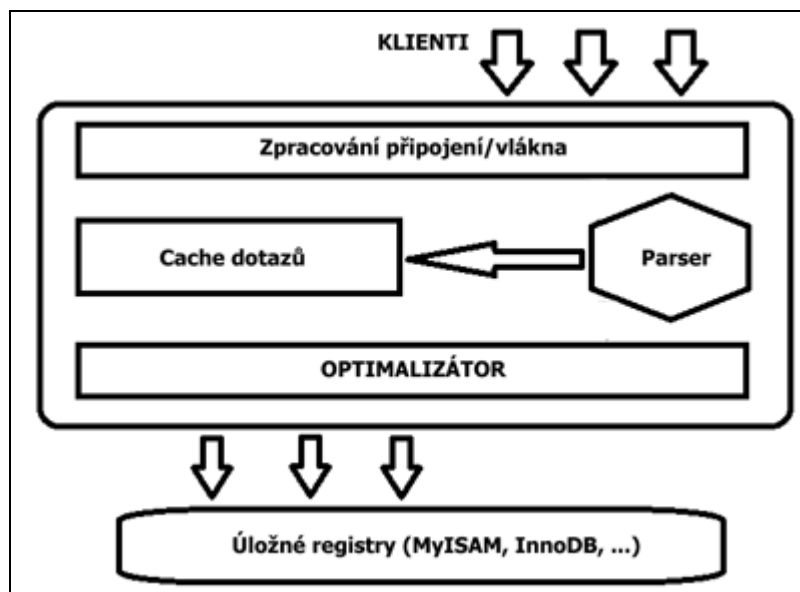
## 1.11 Relační databázové systémy

Většina dnes používaných databázových systému je založena na relačním modelu dat [11], jehož název vychází z relační algebry. V tomto modelu je základní pojmem relace, kterou si lze představit jako tabulku, která se skládá ze sloupců a řádků. Soubor těchto tabulek pak tvoří tzv. relační schéma. Každý atribut v tabulce má svůj datový typ (např. celé číslo, řetězec znaků, datum, logická hodnota, atd.).

V moderních aplikacích hrají relační databázové systémy zásadní úlohu [11], protože poskytují velmi obecný a výkonný nástroj pro uložení a správu dat s možností se více soustředit na funkcionalitu samotné aplikace. Jedním z hojně využívaných databázových systémů je např. MySQL.

### 1.11.1 MySQL

MySQL je platformě nezávislý databázový systém [12], může tak být provozován na různých operačních systémech. Komunikace s databázovým systémem probíhá pomocí jazyka SQL (*Structured Query Language*), který je doplněn o některé rozšíření specifické právě pro MySQL. Od počátku bylo MySQL optimalizováno [12] především na rychlost a k jeho rozšíření vedla převážně skutečnost, že je od verze 3.23.19 systém šířen zdarma pod licencí GPL. Obrázek (Obr. 2) znázorňuje základní architekturu MySQL databázového serveru. Autentizace je založena na uživatelském jménu a heslu.



Obr. 2. Architektura MySQL databázového serveru [12]

### 1.11.2 SQL

Jazyk SQL (*Structured Query Language*) vznikl přibližně v 70. letech 20. století [13] a první standard označovaný jako SQL86 byl přijat v roce 1986. V roce 1992 byla přijata opravená verze označovaná SQL92 nebo SQL2. Od roku 1992 se pracovalo na verzi SQL3, která přinesla podporu objektů do relačních databází. V roce 1999 byla schválena verze SQL:1999, která byla v podstatě podmnožinou SQL3. V současné době se pracuje na verzi SQL4, která by měla SQL3 nahradit.

SQL patří [13] do kategorie tzv. neprocedurálních neboli deklarativních jazyků, takže pomocí jazyka pouze udáváme, jaká data chceme získat, nezajímáme se o způsob provedení. Nepíšeme pomocí něj samostatné programy, ale používáme jej při komunikaci se serverem.

### 1.12 Dibi – PHP knihovna

Dibi je knihovna pro PHP, která umožňuje snadný přístup k databázím mnoha typů, např. MySQL, PostgreSQL, SQLite, MS SQL, Oracle, Access atd. Zjednodušuje zápis SQL příkazů, usnadňuje přístup k dotazovacím metodám a obsahuje funkce pro několik základních rutinních úkonů. Díky tomuto přístupu snižuje počet výskytů chyb, protože používá přehledný zápis SQL příkazů a zabraňuje tzv. SQL injection, což je technika napadení databázového systému [12]. V případě správné aplikace knihovny je zdrojový

kód přenositelný mezi různými databázovými systémy, protože automaticky podporuje rozdílné konvence a různé formáty odlišných databázových systémů. Navíc sjednocuje některé základní funkce, např. připojení k databázi, získání výsledku dotazu apod.

Díky svému provozu vyžaduje PHP ve verzi 5.2 nebo novější. Knihovna je uvolněna pod licencí BSD (*Berkeley Software Distribution*) a GPL (*GNU General Public License*).

### 1.13 Apache

Apache je v současné době jedním z nejrozšířenějších [14] a nejstabilnějších webových serverů. Apache je téměř nezávislý na platformě operačního systému a je navíc distribuován zdarma, díky čemuž získal oblibu u mnoha vývojářů webových aplikací po celém světě. Avšak nejen díky snadné konfiguraci a nevelkým nárokům na operační systém se může webový server Apache pyšnit tak velkou skupinou uživatelů, nýbrž je to i díky jeho obrovské modularitě a přizpůsobitelnosti dle potřeb uživatelů. Webový server Apache se objevuje nejen na platformě Unix/Linux, ale i na počítačích s operačním systémem Microsoft Windows, NetWare, BeOS, Mac aj. Tabulka (Tab. 3) dokumentuje široké zastoupení webového serveru Apache.

Výrobce	Stav v říjnu 2012 [%]
Apache	58,00
Microsoft	16,28
nginx	11,80
Google	3,38

Tab. 3. Zastoupení hlavních webových serverů [14]

Architektura Apache [15] je tvořena několika vrstvami. Tou základní je jádro, které zodpovídá za komunikaci s operačním systémem a komunikaci se sítovým prostředím. Vlastní funkčnost Apache je dána moduly, které mohou být jeho pevnou součástí nebo mohou být dynamicky přidávány nebo odebírány přímo za běhu. Moduly obsahují řadu direktiv, které pak nastavují konkrétní vlastnosti serveru.

Jádro webového serveru Apache zajišťuje [15] nejen základní funkčnost serveru, ale vytváří aplikační rozhraní pro moduly. Jádro tedy naslouchá na komunikačních portech

protokolu TCP/IP, přenáší požadavky protokolu HTTP(S) a musí rovněž zajistit bezpečné propojení ostatních modulů. Z hlediska bezpečnosti je důležitý modul „mod\_ssl“, který umožňuje šifrování např. použitím protokolu SSL. K provozu dynamického obsahu je zapotřebí např. některého z modulů pro interpretaci programovacího jazyka PHP.

### 1.14 Cron

Cron je systémový nástroj [16] (démon), který spouští v daný čas automaticky nějaký proces. Funguje jako softwarový démon s funkcí plánovače úloh, který usnadňuje periodické spouštění požadovaných akcí, čímž umožňuje některé procesy plně automatizovat. Je možné definovat minuty, hodiny, den v měsíci, měsíc a den v týdnu. Některé verze Cronu umožňují i složitější syntaxe, např. násobky dané hodnoty apod.

V souboru „crontab“ jsou uvedeny jednotlivé definice plánování úloh. Na každém řádku je uvedena právě jedna definice plánování. Každý řádek obsahuje několik elementů oddělených tabulátory nebo mezerami. První část řádku definuje časy spouštění dané úlohy. Příkaz, který se má vykonat, je na poslední pozici řádku. Popis jednotlivých pozic na řádku je uveden v následující tabulce (Tab. 4).

Pozice na řádku	Význam
1.	Minuta (0-59)
2.	Hodina (0-23)
3.	Den v měsíci (1-31)
4.	Měsíc (1-12)
5.	Den v týdnu (0-6; kde 0=neděle a 6=sobota)
6.	Příkaz, který se má vykonat.

*Tab. 4. Význam pozic v syntaxi řádku definice Cron*

Pozici 1-5 lze nahradit hvězdičkou, což značí periodické opakování hodnoty na dané pozici.



### 1.15 Memcached

Memcached je jednoduchý démon [16] (služba), který má předem alokované nastavené množství operační paměti, do které mohou různé aplikace ukládat jakákoli data. Přístup k těmto uloženým datům je pak velmi rychlý a samotná data mohou být snadno sdílena mezi různými procesy. V případě, že paměť nedostačuje, odmazávají se nejstarší záznamy. Každému záznamu lze navíc nastavit čas expirace. Memcached je novější projekt než např. GNU/Linux, Apache, MySQL nebo PHP. První verzi tohoto projektu vytvořil [16] Brad Fitzpatrick (americký programátor) v květnu 2003 pro jeho webový projekt LiveJournal. Hlavním cílem projektu [16] bylo odlehčení databázovým systémům s velkým zatížením, zejména snížení počtu přístupů k datům v databázích. Od počátku je systém distribuován zdarma a je multiplatformní, lze jej tedy provozovat na různých operačních systémech.

Využití systému Memcached je limitováno pouze velikostí nastavené alokované paměti a navíc jeho základními vlastnostmi, např. uložením dat pouze v operační paměti, takže při ztrátě napájení dojde ke ztrátě uložených dat. Samotný démon si při spuštění alokuje potřebnou paměť a pak již naslouchá na nastaveném portu (výchozí port je 11211) a IP adrese (např. 127.0.0.1 = pro užití na stejném počítači). Komunikace se systémem Memcached probíhá pomocí velmi jednoduchého protokolu pomocí několika základních příkazů s poměrně jednoduchou syntaxí. Pro použití např. v kombinaci s PHP je však nejčastěji využívána stejnojmenná knihovna, která zajistí samotnou komunikaci pomocí tohoto protokolu a postará se tak o výměnu dat mezi aplikací a systémem Memcached.

### 1.16 Web API

API (*Application Programming Interface*) označuje [17] rozhraní pro programování aplikací. Jde vlastně o několik funkcí, tříd či procedur nějaké knihovny nebo programu, které může programátor využívat v dalších aplikacích. API je tedy obecné rozhraní, pomocí kterého lze řešit konkrétní problémy, avšak nejčastěji s využitím různých programovacích jazyků, protože bývá jazykově nezávislé. Ve významu „Web API“, tedy použití při vývoji webových stránek, je API nejčastěji definováno [17] pomocí HTTP protokolu, kde jsou jako odpovědi na dotazy zasílány obvykle XML nebo JSON formáty. V kontextu tohoto užití se pak jedná o webovou službu, která na základě dotazu klienta provede určitou činnost a vrací klientovi výstup v daném datovém formátu.

### 1.17 Smarty

Smarty [18] je komplexní a funkcionálně hojně vybavený šablonovací systém pro PHP. Primárně jsou šablonovací systémy určeny k oddělení prezentace dat od jejich samotného zpracování. Vývojář se tak může soustředit na funkčnost aplikace (aplikační logiku), samotné zobrazení dat je odděleno do jiných zdrojových kódů a prezentaci dat (prezenční logiku) tak může zpracovávat jiný vývojář. V ideálním případě tak dochází k lepšímu rozčlenění a zjednodušení vývoje, což snižuje náklady spojené s údržbou softwaru. Smarty generuje webový obsah umístěním speciálních příkazů v dokumentu. Tyto příkazy jsou v průběhu zpracování nahrazeny jiným kódem. Syntaxe Smarty odděluje samotný dokument od příkazů pro Smarty pomocí speciálních znaků, konkrétně využívá složené závorky „{“ a „}“. Proměnné jsou podobně jako v PHP označeny počátečním znakem dolaru „\$“. Smarty dále umí definovat nejrůznější funkce pro snazší práci v prezenční vrstvě aplikace, ale obsahuje i spoustu již předdefinovaných funkcí. Samozřejmostí jsou nejrůznější cykly a smyčky, stejně jako podmíněné části kódu apod.

### 1.18 PDF

PDF (*Portable Document Format*) je otevřený souborový formát pro uložení dokumentů nezávisle na hardwaru i softwaru, na kterém byly vytvořeny [19], tzn. na cílové platformě je dokument zobrazen naprosto totožně. Souborový formát byl vyvinut firmou Adobe [19], která jej představila již v roce 1993. Specifikace tohoto formátu byla poskytnuta bezplatně, ovšem zdarma nebyl původně k dispozici software pro prohlížení a vytváření tohoto formátu. Po vydání druhé verze Adobe začalo distribuovat program pro prohlížení dokumentů PDF s názvem Acrobat Reader zdarma, což znamenalo obrovské rozšíření tohoto formátu a stal se jedním z nejpoužívanějších formátů pro ukládání a výměnu dokumentů současné doby. Formát PDF vychází z jazyka PostScript [19], avšak některé elementy jsou ve formátu PDF implementovány zcela odlišně a některé nejsou dostupné vůbec. Naopak byla do PDF přidána schopnost vložit do PDF dokumentu omezenou sadu znaků, což umožňuje jejich pozdější tisk a zobrazení na zařízeních, které tuto omezenou sadu znaků neobsahují. Na rozdíl od formátu PostScript používá formát PDF kompresi, výsledné dokumenty jsou tedy datově mnohem menší. Soubory typu PDF mohou obsahovat nejen texty, ale i vektorovou či rastrovou grafiku, tabulky, apod.

## 2 POTŘEBY A POVINNOSTI EKONOMICKÝCH SUBJEKTŮ

Většina ekonomických (podnikatelských) subjektů v ČR je povinna [20] vést evidenci podnikatelské činnosti (účetnictví nebo daňovou evidenci) zachycující jejich hospodářskou činnost, s čímž je spojena spousta každodenních úkonů potřebných k běžnému provozu.

### 2.1 Vedení evidence podnikatelské činnosti

Od 1. 1. 2004 existují v České republice dvě účetní soustavy [20], podnikatelskou činnost tedy lze evidovat dvojím způsobem:

- **Daňová evidence** – musí ji vést ty fyzické osoby, které dosahují příjmů z podnikání a z jiné samostatné výdělečné činnosti (např. živnostníci, umělci, daňoví poradci) a nejsou přímo účetní jednotkou. Je upravena zákonem o daních z příjmů [20].
- **Účetnictví** – musí ho vést právnické osoby (např. obchodní společnosti) a fyzické osoby, které se staly účetní jednotkou. Je upraveno zákonem o účetnictví [21].

Fyzická osoba se stává účetní jednotkou [20] hlavně v těchto případech:

- obrat fyzické osoby za předcházející kalendářní rok přesáhl částku 25 milionů Kč,
- fyzická osoba se dobrovolně rozhodla vést účetnictví,
- fyzická osoba je zapsána v obchodním rejstříku.

Úkolem daňové evidence je zejména zjištění správné výše základu daně z příjmů, oproti tomu účetnictví poskytuje mnohem komplexnější pohled na celkové hospodaření firmy.

### 2.2 Daňová evidence

Daňová evidence je obdobou do konce roku 2003 používaného jednoduchého účetnictví. Daňovou evidenci neupravují účetní předpisy ze zákona o účetnictví, ale zákon o daních z příjmů [20].

Daňová evidence musí poskytovat informace:

- a) o příjmech a výdajích,
- b) o majetku a závazcích ekonomického subjektu.

Hlavním požadavkem pro rozdělení příjmů a výdajů je zajistit podklady pro správný výpočet základu daně z příjmů, tj. schopnost rozdělit příjmy a výdaje na zahrnované a nezahrnované do základu daně. Na rozdíl od účetnictví je [20] daňová evidence upravena pouze zákonem o daních z příjmů.

## 2.3 Účetnictví

Účetnictví je písemné zaznamenávání informací o hospodářských jevech podniku [20], a to v peněžních jednotkách. Účetnictví svými písemnými nebo elektronickými záznamy eviduje činnost účetní jednotky. V účetnictví se eviduje např. nákup materiálu, zboží a majetku, zaznamenává se prodej výrobků, zboží, služeb, vznik pohledávek a závazků ekonomického subjektu vůči obchodním partnerům (odběratelům, dodavatelům), zaměstnancům, finančním orgánům, peněžním ústavům (bankám) apod.

Účetnictví musí poskytovat [21] informace:

- a) o nákladech, výnosech a s tím souvisejícím zjištění hospodářského výsledku,
- b) o stavu a pohybu majetku a závazků.

Účetnictví zajišťuje tyto funkce:

1. Poskytuje informace o hospodaření firmy.
2. Při vedení případných sporů slouží jako důkazní prostředek (např. účetní doklady nebo záznamy).
3. Poskytuje informace pro rozhodování a řízení podniku.
4. Uchovává informace pro daňové účely.
5. Údaje v účetnictví usnadňují kontrolu stavu majetku a hospodaření s majetkem.

Základním předpisem pro vedení účetnictví je zákon o účetnictví [22], kterým se řídí vedení účetních záznamů závazně pro všechny účetní jednotky. Povinností podle zákona o účetnictví je vést účetnictví úplně, průkazným způsobem a správně tak, aby věrně projektovalo skutečnosti, které jsou jeho předmětem. Průkazným způsobem je účetnictví vedeno, jestliže jsou všechny účetní aktiva doloženy účetními doklady. Za správnost účetnictví zodpovídá sám podnikatel. Účetní jednotky sice mohou pověřit vedením účetnictví i jinou osobu, tím se ale účetní jednotka [20] nezabavuje odpovědnosti za vedení

účetnictví. Účetnictví se vede v české měně, tedy i v případech, kdy jsou závazky nebo majetky vyjádřeny v cizí měně, musí účetní jednotka vyjádřit jejich hodnotu v české měně.

## 2.4 Účetní doklady

Účetní doklad je jasně průkazný účetní záznam [20], který zachycuje a rovněž ověřuje účetní nebo hospodářskou operaci. Účetní doklady musí obsahovat předepsané náležitosti a fakta, které jasně definuje zákon. Podstatou účetního dokladu je jednoznačná identifikace účetních případů.

### 2.4.1 Náležitosti účetních dokladů

Aby byly účetní doklady jednoznačným průkazným prostředkem o provedených operacích firmy, musí mít podle zákona [21] několik náležitostí:

- a) **Označení účetního dokladu** – smysl označení účetních dokladů je jejich jednoznačná, jasná a rychlá identifikace. Účetní jednotky mají povinnost označovat účetní doklady takovým způsobem, aby bylo naprosto jasné, že jsou kompletní. Zákon nestanovuje pravidla pro označování účetních dokladů, ale je vhodné používat jednu nebo více číselných řad.
- b) **Označení účastníků účetního případu** – příjmení, jméno a místo podnikání fyzické osoby podle živnostenského listu nebo obchodní jméno a sídlo právnické osoby dle výpisu z obchodního rejstříku.
- c) **Označení obsahu účetního případu** – např. za nákup zboží, za poskytnutí advokátních služeb atd.
- d) **Okamžik vyhotovení účetního dokladu** – jedná se o datum (zpravidla s přesností na dny), kdy je účetní doklad fyzicky vytvořen. Může být stejný jako okamžik uskutečnění účetního případu.
- e) **Vyjádření množství a peněžní částka nebo údaje o ceně za měrnou jednotku** – např. 12 000 Kč nebo 120 ks knih po 240 Kč.
- f) **Okamžik uskutečnění účetního případu** – uvádí se pouze v případě, když není shodný s okamžikem vyhotovení účetního dokladu.

- g) **Podpisový záznam osoby odpovědné za účetní případ** – tato osoba stvrzuje obsah účetního případu. Jedná se tedy o podpis osoby, která garantuje věcnou správnost účetního případu.
- h) **Podpisový záznam osoby odpovědné za zaúčtování účetního případu** – tato osoba svým podpisem garantuje ověření dokladu z formálního hlediska, např. kontrolou úplnosti náležitostí účetních dokladů.

Výše uvedené náležitosti jsou povinné a zpravidla není možné vyžadovat žádné jiné náležitosti. Z praktického hlediska se však doporučuje uvádět na účetních dokladech další informace:

- otištěné razítko firmy přímo na účetním dokladu,
- předkontace neboli účtovací předpis, tj. uvedení účtů, na kterých se případ zaúčtuje,
- uvedení dalších kontaktních údajů účastníků případu (např. telefon, email atd.).

#### 2.4.2 Vyhотовování účetních dokladů

Vyhотовování účetních dokladů má probíhat bez odkladů, v ideálním případě okamžitě po zjištění skutečností, které se těmito doklady ověřují. Zákon o účetnictví [22] nestanovuje formu a typ dokladů, stanovuje pouze jejich povinné náležitosti. Povinností je zápis údajů způsobem zaručující trvalost zápisu, aby byly údaje čitelné po celou dobu archivace. Lhůtu úschovy účetních dokladů stanovuje zákon. Opravy účetních dokladů nesmí vést k jejich nesprávnosti, neúplnosti či neprůkaznosti a lze je realizovat [22] těmito způsoby:

- a) editací původního (opravovaného) dokladu (tzv. bezdokladový způsob),
- b) vystavením nového opravného účetního dokladu k původnímu (chybnému) dokladu (tzv. dokladový způsob).

#### 2.4.3 Druhy účetních dokladů

Účetní doklady lze podle druhu dokladu rozdělit [20] na několik typů, z nichž má každý jiný účel:

- a) **Faktury vystavené** – účetní doklady, které účtují provedené dodávky zboží, výrobků nebo služeb, na základě kterých vzniká podnikatelskému subjektu pohledávka (nárok na zaplacení) vůči odběrateli.

- b) **Faktury přijaté** – účetní doklady, které účtují obdržené dodávky zboží, výrobků nebo služeb od dodavatelů, na základě kterých vzniká podnikatelskému subjektu závazek (povinnost zaplatit) vůči dodavateli.
- c) **Příjmové pokladní doklady** – používají se pouze u přijatých hotovostních plateb a jsou vystavovány ve dvou provedeních (originál pro zákazníka a kopii pro podnikatelský subjekt dodávající předmět dokladu).
- d) **Výdajové pokladní doklady** – obdobné jako příjmový pokladní doklad, ale na jejich základě se účtují výdaje v hotovosti.
- e) **Bankovní výpisy** – obsahují informace o stavu a pohybu peněz na bankovních účtech podnikatelských subjektů.
- f) **Interní účetní doklady** – účtují hospodářské operace uvnitř firmy, např. příjemky, výdejky nebo výplatní listiny.

## 2.5 Daň z přidané hodnoty

Daň z přidané hodnoty (*DPH*) je nepřímá daň [22]. Podnikatelský subjekt se v tomto případě stává prostředníkem mezi finančním úřadem (státem) a poplatníkem daně (spotřebitelem, zákazníkem). Daň je obsažena v ceně zboží, materiálu nebo služeb, není placena přímo státu. Podnikatel tedy tuto daň připočítává k cenám služeb nebo výrobků a spotřebitel daň hradí přímo s cenou. Na druhou stranu podnikatelské subjekty (plátcí DPH), mají nárok na odpočet DPH v případech, kdy výše daně z nakoupeného majetku nebo služeb převyší hodnotu daně z prodaných služeb či zboží (DPH zaplacené na vstupu není součástí nákladů podnikatele).

### 2.5.1 Plátce DPH

Pokud podnikatel není plátce DPH, při nákupech majetku či služeb nemá nárok na odpočet daně na vstupu a naopak nemá povinnost odvádět DPH z tržeb finančnímu úřadu.

Právnícká nebo fyzická osoba, která uskutečňuje zdanitelná plnění v rámci svého podnikání, se může stát plátcem DPH. Plátcí DPH se k dani musí zaregistrovat u finančního úřadu, registrace [20] může být:

- a) **povinná** – nejčastěji kvůli překročení obratu (obrat přesáhne za 12 po sobě následujících měsíců částku 1 milion Kč),
- b) **dobrovolná** – podnikatelský subjekt sice nepřekračuje limit povinné registrace, ale stává se plátcem DPH dobrovolně na základě vlastní žádosti.

### 2.5.2 Sazby DPH

Sazba daně má v současnosti [22] v České republice dvě základní úrovně:

- a) **základní sazba** – většina zboží i služeb podléhá právě této sazbě,
- b) **snížená sazba** – pouze několik artiklů a služeb, které jsou definovány zákonem o DPH (např. potraviny, knihy apod.).

Pro rok 2013 je základní sazba DPH 21% a snížená sazba DPH činí 15%. V roce 2012 byla ovšem základní sazba DPH 20% a snížená sazba pouze 14%. Jak již z vývoje sazeb DPH vyplývá, výše této daně je v ČR často proměnlivá. Není vyloučeno, že budou obě tyto sazby v budoucnu sjednoceny.

### 2.5.3 Daňové doklady

Plátce DPH je povinen [20] vystavovat daňové doklady. Každý plátce daně z přidané hodnoty musí vést záznamy o základu daně, určit přesně den vzniku daňové povinnosti a určit správnou výši daně. Odběratel (plátce DPH) může uplatnit nárok na odpočet daně pouze na základě daňových dokladů, naopak vznik daňové povinnosti signalizuje důvod pro vystavení daňového dokladu ze strany dodavatele (plátce DPH). Daňový doklad může mít mnoho forem, nejčastěji se však setkáváme s dvěma typy daňových dokladů:

- a) **Běžný daňový doklad** – zpravidla se jedná o fakturu obsahující povinné náležitosti běžného daňového dokladu. Oproti běžným účetním dokladům musí navíc obsahovat [20] přesné označení prodávajícího i kupujícího včetně DIČ (*daňové identifikační číslo*), datum uskutečnění zdanitelného plnění, rozsah a předmět zdanitelného plnění a údaje rozhodné pro stanovení výše DPH (výše daně, základ daně a sazbu daně). Zákon o DPH stanovuje [23], že údaje o základu a výši daně musí být na daňových dokladech uváděny v české měně.



- b) **Zjednodušený daňový doklad** – oproti běžnému daňovému dokladu nemusí obsahovat některé náležitosti. Zejména nemusí obsahovat údaje o kupujícím a datum vystavení dokladu, stačí pouze datum uskutečnění zdanitelného plnění (data jsou totožná). Na zjednodušeném daňovém dokladu nemusí být uveden základ daně (cena bez daně) a výše daně, musí na něm být uvedena pouze sazba DPH a celková uhrazená částka. Tento doklad je možné vystavit pouze do částky 10 tisíc Kč včetně DPH [23] a doklad se vyhotovuje ihned při prodeji. Doklad lze vystavit pouze v případě úhrady za hotové, šekem nebo prostřednictvím platební karty.

#### 2.5.4 Dodání a pořízení zboží v rámci Evropské unie

Vstup České republiky do Evropské unie (EU) znamenal pro český právní systém mnoho změn. V oblasti DPH byl zaveden nový systém uplatňování DPH, který začal upravovat nový zákon o DPH. Jednou z největších změn bylo ustanovení nového postojů k dodání a pořízení zboží zákazníkům registrovaným k DPH v jiné členské zemi EU. Od roku 2004 se již nejedná [23] z hlediska celních a daňových předpisů o vývoz nebo dovoz zboží, nově se tato skutečnost považuje za dodání nebo pořízení zboží z jiného členského státu EU, čímž je zboží osvobozeno od plnění DPH (povinnost přiznat DPH přechází na protistranu).

Při dodání zboží do jiného členského státu musí mít dodavatel (český plátc DPH) k dispozici daňové identifikační číslo odběratele z jiného členského státu EU a je povinen [23] před dodáním zboží bez DPH ověřit jeho formát a platnost. V případě, že není dodavatel schopen prokázat splnění všech povinných náležitostí, nemůže uplatnit osvobození od české DPH. Vystavené daňové doklady musí navíc obsahovat sdělení, že se jedná o plnění osvobozené od daně dle § 64 zákona o DPH [23].

#### 2.5.5 Ověření platnosti DIČ

Pro ověření platnosti DIČ (zejména u odběratelů registrovaných k DPH v jiném členském státu EU) existuje několik možností, např. vyžádání si osvědčení o registraci přímo od odběratele. K dispozici jsou ale i jiné a pohodlnější možnosti ověření registrace DIČ k DPH v jiném členském státu EU:

- telefonicky, emailem, faxem nebo písemně na příslušném útvaru Ministerstva financí ČR,

- telefonicky, emailem, faxem, písemně nebo osobní návštěvou na místě příslušném finančním úřadě,
- na internetových stránkách Evropské komise pomocí systému VIES (*VAT Information Exchange System*).

Poslední varianta ověření platnosti DIČ je nejsnazší, nejrychlejší a nejlépe dostupná pomocí internetu. Ukázka webového výstupu z VIES po ověření platného DIČ je na uvedeném obrázku (Obr. 3).

**Ověřování DIČ pro účely DPH prostřednictvím systému VIES**

**Ano, platné DIČ**

<b>Členský stát</b>	CZ
<b>DIČ</b>	CZ 70883521
<b>Datum přijetí žádosti</b>	2013/04/09 16:04:03
<b>Jméno</b>	Univerzita Tomáše Bati ve Zlíně
<b>Adresa</b>	nám. T. G. Masaryka 5555 ZLÍN 760 01 ZLÍN 1
<b>unikátní číslo konzultace</b>	WAPIAAAAT3vHD7Mr

Obr. 3. Ukázka výstupu ze systému VIES po ověření platného DIČ

## 2.6 Měny

Měna je definována [24] jako národní forma peněz. Penězi jsou chápány české koruny, euro, americké dolary, britské libry apod., ale měnou v ČR je pouze česká koruna. Nadnárodní formou peněz je myšleno např. euro, kterým se platí ve více státech. Každá měna má své ekonomické a technické jevy (znaky), které slouží k charakterizaci a odlišení jednotlivých měn. Technické jevy jsou spojeny s formálními skutečnostmi.

Technické znaky měny [24] jsou zejména:

- a) **název měny** – např. česká koruna, euro, americký dolar atd.,

- b) **kód a symbol měny** – každá měna má svůj mezinárodní kód o délce 3 znaků (např. pro českou korunu se jedná o „CZK“ nebo pro americký dolar „USD“) a často svůj symbol, který je použit při vyjadřování hodnoty (např. „Kč“ pro českou korunu nebo „\$“ pro americký dolar),
- c) **dělení a kumulace základní měnové jednotky** – u „silnějších“ měn je důležitější dělení než kumulace a naopak,
- d) **hotovostní druhy vč. nominální struktury** – základní měnová jednotka se dělí na menší jednotky a používají se násobky základní nominální hodnoty (mince a bankovky),
- e) **výlučnost měny na daném území** – pouze určitá měna musí být akceptována na daném území,
- f) **způsob ustanovení měnového kurzu** – různé možnosti stanovení hodnoty měny ve vztahu k dalším měnám, popř. vyjadřuje vztah měny ke zlatu.

## 2.7 Měnové kurzy

Měnový kurz je vyjádření hodnoty jedné měny v násobcích základní nominální hodnoty jiné měny. Stanovení měnového kurzu probíhá v různých formách, nejčastěji podle druhu měny. U nesměnitelných měn se kurzy nevytváří na devizových trzích [24], kurzy jsou tzv. fixní a stanovuje je centrální orgán státu, ve kterém je měna používána. U volně směnitelných měn se používají tzv. pružné kurzy, což znamená, že se jejich kurz vůči ostatním měnám neustále mění. Ve smyslu zákona o účetnictví a dalších právních norem pro neobchodní i obchodní účely [22] stanovuje a pravidelně zveřejňuje měnové kurzy Česká národní banka (ČNB) na základě monitorování vývoje měn na mezibankovním devizovém trhu.

## 2.8 Bankovní účty

Na základě smluv o vedení účtů poskytují banky svým klientům bankovní účty pro bezhotovostní úhrady a příjem financí, které jsou v dnešní době v rámci podnikání jednou z nejrozšířenějších platebních metod pro úhradu pohledávek a závazků. Bankovní účet zaznamenává veškeré finanční transakce [25] a eviduje tak jeho finanční situaci.

Rozlišuje se několik typů bankovních účtů. Základním rozdělením jsou účty podle jejich měny, ve které jsou vedeny. Účty tedy mohou být vedeny v tuzemské měně (Kč) a v cizích měnách (např. EUR, USD apod.), což jsou tzv. účty devizové.

Pomocí bankovních účtů mohou klienti provádět několik typů operací a transakcí, nejčastěji se jedná o příkaz k úhradě, příkaz k inkasu, vklad a výběr hotovosti. Veškeré bankovní operace a transakce jsou po té evidovány na bankovních výpisech, které banky poskytují ke všem bankovním účtům v papírové nebo elektronické podobě.

## 2.9 ABO formát

Formát ABO se v České republice běžně používá [25] pro výměnu finančních zpráv, tj. např. export informací o pohledávkách do tohoto formátu a následný import do banky, kde se na jeho základě provede úhrada pohledávek. Jedná se o datový soubor s jasně a pevně definovanou strukturou [25]. Hlavním účelem tohoto formátu je částečná automatizace některých procesů a rovněž snížení chybovosti při přepisu údajů k platbám apod. Technickou dokumentaci k ABO formátu zveřejňují bankovní instituce a jsou volně dostupné na webu.

Uspořádání v datovém souboru ve formátu ABO je rozděleno na několik bloků. Datový soubor se skládá ze hlavičky účetního souboru, která uvádí, zda se jedná o příkazy k úhradě nebo příkazy k inkasu a obsahuje některé další základní informace, jako např. kód banky účtu, kterého se datový soubor týká. Soubor dále obsahuje hlavičky jednotlivých skupin příkazů, které se již mohou v souboru opakovat. Tyto hlavičky skupin udávají číslo účtu příkazce (pokud není uvedeno, musí být uvedeno u jednotlivých položek), celkovou sumu převodů či inkas ve skupině a datum splatnosti. Dále už v každé skupině následují jednotlivé položky (úhrady, inkasa). Každá položka může obsahovat číslo účtu příkazce (pokud není uvedeno v hlavičce skupiny), číslo účtu příjemce nebo inkasovaného, částku, variabilní symbol, konstantní symbol, směrový kód banky a zprávu pro příjemce. Může obsahovat i specifický symbol, není ale povinný.

### 3 REŠERŠE EXISTUJÍCÍCH ŘEŠENÍ

Cílem webového ekonomického informačního systému je využití internetu a širokého zastoupení webových prohlížečů na odlišných zařízeních k získání moderního ekonomického softwaru, který umožňuje přístup do systému nejenom samotným administrátorům systému, ale i zákazníkům a dodavatelům ekonomických subjektů. Systém musí rovněž poskytovat možnost automatické fakturace periodických služeb.

Ekonomických informačních systémů je na českém trhu celá řada, většina z nich ale není navržena jako webový informační systém, jedná se převážně o standardní aplikace s architekturou klient-server pro operační systém Microsoft Windows, popř. pouze lokální verze aplikací určených pouze pro přístup z jednoho počítače. Existuje i několik webových řešení, žádné z nich ale neumožňuje přístup zákazníků a dodavatelů do systému za účelem poskytování podstatných informací. Následující část představuje významné existující řešení a jejich nevýhody, z kterých plyne nesplnění základních požadavků na webový ekonomický informační systém.

#### 3.1 POHODA online

Velmi rozšířený ekonomický informační systém POHODA od společnosti STORMWARE s.r.o. [26] je oborově neutrální ekonomický software pro malé, střední a větší firmy z řad fyzických i právnických osob. Systém POHODA nabízí i variantu „online“, která umožňuje přístup k ekonomickému systému za využití internetu. Tato varianta produktu ovšem vyžaduje nainstalovanou speciální klientskou aplikaci pro operační systém Microsoft Windows, která realizuje vzdálený přístup k systému, nejedná se tedy o standardní webovou aplikaci pro běžně dostupný webový prohlížeč. Systém neumožňuje automatickou fakturaci periodických služeb. POHODA online nepodporuje přístup zákazníků a dodavatelů, jedná se pouze o ekonomický systém pro administrátory.

#### 3.2 iDoklad

Softwarová společnost CÍGLER SOFTWARE, a.s. se zabývá převážně vývojem a implementací ekonomických informačních systémů Money. Nejnovější službou společnosti je řešení nazvané iDoklad [27], které je založené na webovém přístupu za využití běžného webového prohlížeče. V tomto případě se však nejedná o komplexní ekonomický systém,

služba slouží pouze k vystavování a evidenci vydaných dokladů a kontaktů. IDoklad nepodporuje přístup zákazníků a dodavatelů a neumožňuje automatickou fakturaci služeb.

### **3.3 FlexiBee online**

Systém FlexiBee od společnosti FlexiBee Systems s.r.o. je komplexní ekonomický informační systém, který podporuje širokou řadu evidencí. Systém nabízí [28] možnost vzdáleného přístupu přes internet. Stejně jako v případě systému POHODA online je zde možnost instalace speciální klientské aplikace, lze ale využít i jiných operačních systémů (Linux, Mac OS X, Android, iOS). V případě FlexiBee online lze k systému a všem jeho funkcím přistupovat i přes webové rozhraní, systém ale neumožňuje automatickou fakturaci periodických služeb a nepodporuje přístup zákazníků a dodavatelů.

### **3.4 Fakturoid**

Fakturoid je [29] online fakturační systém pro živnostníky a malé firmy od spoluautorů Lukáše Konarovského a Jana Korbela. Jedná se o webový informační systém, který mimo jiné umožňuje automatickou fakturaci periodických služeb. Fakturoid bohužel nepodporuje přístup zákazníků a dodavatelů a neumožňuje evidenci přijatých dokladů.

### **3.5 Qfaktury**

Fakturační software Qfaktury [30] je webový informační systém určený převážně živnostníkům a malým firmám. Autorem systému je Ing. Jan Daniel (společnost Q2 Interactive). Informační systém umožňuje i automatickou fakturaci periodických služeb, ale jako v předchozím případě nepodporuje evidenci přijatých dokladů, ani přístup zákazníků a dodavatelů.

### **3.6 IÚčto**

Webový systém iÚčto od společnosti Direct Accounting s.r.o. umožňuje [31] plnohodnotné vedení účetnictví a daňové evidence online za využití webového prohlížeče. Oproti webovým systémům Fakturoid a Qfaktury je tedy v systému i možnost evidence přijatých dokladů, bohužel systém naopak nepodporuje automatickou fakturaci periodických služeb, ani přístup zákazníků a dodavatelů.

## **II. PRAKTICKÁ ČÁST**

## 4 NÁVRH SYSTÉMU

Obsahem této kapitoly je návrh webového ekonomického informačního systému.

### 4.1 Konceptuální popis systému

Cílem projektu je webový ekonomický informační systém pro malé a střední společnosti vedoucí účetnictví a současně pro drobné podnikatele či živnostníky vedoucí daňovou evidenci. Ekonomický systém bude oborově neutrální a bude tak vhodný pro podnikatelské subjekty zabývající se jakýmkoli obchodem či poskytováním různých služeb. Systém bude provozován jako webová služba přístupná přes webový prohlížeč, uživatelé budou mít přístup do systému po přihlášení. Systém bude poskytovat intuitivní webové uživatelské rozhraní. Úkolem systému nebude nahradit pokročilé účetní programy, které na trhu již jsou, nýbrž usnadnit podnikatelským subjektům běžný provoz jejich firem. Vzhledem ke komplexnosti celé problematiky je již v návrhu kalkulováno s postupným vývojem systému a postupným rozšiřováním jeho možností.

Systém musí podporovat ekonomické subjekty plátce DPH a zároveň subjekty, kteří nejsou plátcí DPH. IS musí umožnit podnikatelským subjektům přechod z režimu neplátce DPH na plátce a obráceně. Plátcí DPH budou díky systému moci na základě DIČ ověřovat, zda je protistrana rovněž plátce DPH, což je vhodné např. při dodání zboží do jiného členského státu EU.

K hlavním funkcím bude patřit evidence přijatých a vydaných faktur, evidence pohledávek, závazků a finančních transakcí. Veškeré operace mohou být jak v tuzemské, tak v devizové měně. Aktualizace kurzů měn může probíhat automaticky dle ČNB, popř. může subjekt nastavovat účetní kurzy manuálně. Systém bude umožňovat generování vydaných faktur v PDF formátu. Příchozí bankovní transakce budou do systému ukládány pouze automaticky na základě elektronických oznámení z bank, popř. na základě jiných principů dle možností konkrétních bankovních institucí. K úhradě závazků bude systém umožňovat generování datových souborů ve formátu ABO, podnikatelům tak bude umožněno hradit své závazky pohodlně přímým importem do banky. Systém musí zpřístupňovat základní statistiky finančních toků včetně historických záznamů. Systém bude dále evidovat veškeré kontakty odběratelů a dodavatelů, čímž bude automaticky vznikat adresář kontaktů.



K dalším funkcím systému budou patřit základní exporty a importy některých dat, např. export vydaných faktur pro účetní software Money S3.

Systém musí reprezentovat ekonomická data subjektů na základě uživatelských oprávnění. Periodické služby zákazníkům budou fakturovány automaticky, zákazníci mají proto do systému přístup, aby mohli např. nastavit délku fakturačního období nebo zkontrolovat stav jejich služeb. Zákazníkům je umožněno prohlížet jejich vlastní doklady, popř. je ukládat ze systému přímo ve formátu PDF. Do systému mohou mít přístup i dodavatelé, aby mohli např. zobrazit své evidované doklady či zkontrolovat stav jejich úhrad, což je v současné fázi vývoje pouze jedno z možných rozšíření. V návrhu systému musí být s dodavateli kalkulováno, jejich rozšířené funkce v systému však mohou být definovány až v budoucnu. V současném návrhu je k dodavatelům přístupováno stejně jako k zákazníkům, protože sám dodavatel může být rovněž zákazníkem podnikatelského subjektu. Jako další rozšíření může být implementována skladová evidence, která bude případně úzce spojena se samotnou fakturací. Další rozšíření mohou být do systému implementována v budoucnu dle potřeb konkrétních ekonomických subjektů.

## 4.2 Analýza požadavků

Na základě konceptuálního popisu systému lze definovat seznam funkčních a nefunkčních požadavků na webový ekonomický informační systém.

### 4.2.1 Funkční požadavky

Funkční požadavky definují požadované funkce informačního systému. Požadavky jsou pro přehlednější členění rozděleny do podkategorií.

#### **Vydané doklady:**

- Výpis vydaných dokladů – tabulkové zobrazení obsahující číslo dokladu, název odběratele, celkovou částku včetně i bez DPH (u neplátců DPH pouze konečnou cenu), datum vystavení, formu úhrady a informaci o tom, zda už je doklad uhrazen. Musí být umožněno přímé zobrazení a stažení dokladu, popř. další akce spojené se správou vydaných dokladů. Výpis dokladů lze filtrovat podle měsíců, ve kterém byly doklady vystaveny, a podle stavu jejich úhrady.

- Vystavení nového vydaného dokladu – formulář umožňující vytvoření nové faktury nebo zálohové faktury (proforma faktury). Kontakt odběratele lze zvolit z adresáře nebo je možné zadat nový kontaktní údaj. Na vydaný doklad lze přidat jakékoli množství položek, u každé lze nastavit sazbu DPH, počet množstevních jednotek a cenu. Dále je nutné u vydaných dokladů umožnit nastavení veškerých povinných náležitostí, které byly podrobně analyzovány v kapitole 2.4.1. Před samotným vystavením dokladu lze nejprve zobrazit náhled v PDF formátu. Pokladní příjmové doklady jsou vystavovány automaticky při volbě hotovostní formy úhrady.
- Zobrazení vydaného dokladu (webová verze, generování PDF) – zobrazení vydaného dokladu obsahující veškeré zadané údaje. V případě webové verze jsou navíc zobrazeny události spojené s vydaným dokladem a je umožněna správa daného dokladu, např. editace, vystavení dobropisu.
- Editace vydaného dokladu – podobně jako vystavení nového vydaného dokladu, ale jsou editovány údaje z již vytvořeného dokladu.
- Označení vydaného dokladu jako uhrazený – ve výpisu dokladů nebo v zobrazení dokladu lze označit doklad ručně jako uhrazený. Označení úhrad vydaných dokladů probíhá částečně automatizovaně v případě propojení s bankovními účty na základě příchozích plateb a jejich variabilních symbolů.
- Storno vydaného dokladu – je umožněno pouze neplátcům DPH, u plátců DPH je vystavován dobropis. Slouží zejména ke zrušení dokladu se zachováním návaznosti číselné řady vydaných dokladů.
- Vystavení opravného daňového dokladu (dobropisu) – je možné vystavit přímo jako nový vydaný doklad nebo lze vytvořit dobropis z již vystaveného vydané dokladu, což značně urychlí celý proces, jelikož budou veškeré údaje předem vyplněny.
- Uzamčení vydaného dokladu – znemožní jakékoliv další úpravy vydaného dokladu, popř. lze tímto označit doklad jako zkontrolovaný apod.
- Export vydaných dokladů ve formátu XML – systém musí umožnit export vydaných dokladů ve formátu XML pro účetní software Money S3. Samotný export je umožněn hromadným označením jedné a více položek ve výpisu vydaných dokladů.

**Přijaté doklady:**

- Výpis přijatých dokladů – tabulkové zobrazení obsahující ID dokladu, číslo dokladu, název dodavatele, celkovou částku včetně i bez DPH (u dokladů přijatých od neplátců DPH pouze konečnou cenu), datum vystavení, účel dokladu, datum splatnosti a datum úhrady, popř. je zobrazen počet dní před splatností. Musí být umožněno přímé zobrazení dokladu, popř. další akce spojené se správou přijatých dokladů. Výpis dokladů lze filtrovat podle měsíců, ve kterém byly doklady vystaveny, a podle stavu jejich úhrady.
- Vytvoření nového přijatého dokladu – formulář umožňující přidání nového přijatého dokladu do systému. Kontakt dodavatele lze zvolit z adresáře nebo je možné zadat nový kontaktní údaj. U přijatých dokladů lze ověřit zadáním DIČ, zda je dodavatel plátcem DPH. Dále lze u přijatých dokladů evidovat bankovní účet pro následnou úhradu závazku.
- Přílohy souborů k přijatému dokladu – ke každému přijatému dokladu lze uložit jeden nebo více souborů (např. naskenovaný doklad v PDF formátu), což umožní archivaci přijatých dokladů.
- Zobrazení přijatého dokladu - zobrazení webové verze přijatého dokladu obsahující veškeré zadané údaje o dokladu. Dále je možné uložit nebo zobrazit příložené soubory.
- Editace přijatého dokladu – podobně jako vytvoření nového přijatého dokladu, ale jsou editovány údaje z již vytvořeného dokladu.
- Smazání přijatého dokladu – přijaté doklady lze smazat po potvrzení v dialogovém okně. Spolu s přijatým dokladem jsou smazány i příložené soubory.
- Označení přijatého dokladu jako uhrazený – ve výpisu dokladů nebo v zobrazení dokladu lze označit doklad ručně jako uhrazený,
- Generování souborů ve formátu ABO – slouží ke snadné úhradě závazků plynoucích z přijatých dokladů. Soubory jsou exportovány hromadným označením alespoň jednoho neuhrazeného přijatého dokladu a následně importovány v internetovém bankovníctví. Po vygenerování je možné hromadně označit doklady jako uhrazené.

**Služby (periodická fakturace):**

- Výpis služeb – tabulkové zobrazení obsahující název služby, označení domény (předmětu) služby, datum zprovoznění služby, datum ukončení služby (resp. datum, do kdy je služba uhrazena), roční poplatek za službu, délku fakturačního období a aktuální vydaný doklad ke službě.
- Vytvoření nové služby – formulář umožňující přidání nové služby do systému. Je možné zvolit veškeré údaje, které jsou zobrazeny ve výpisu služeb.
- Přiřazení služby k uživateli – službu je možné přiřadit ke konkrétnímu zákazníkovi, který může po přihlášení do systému evidovat své služby.
- Změna délky fakturačního období – zákazník i administrátor mohou měnit délku fakturačního období, popř. je umožněno pozastavení další fakturace služeb.
- Automatická fakturace služeb - na základě služeb v systému dochází k jejich automatické fakturaci. Automaticky vystavené doklady za služby jsou v případě služeb přiřazených ke konkrétním zákazníkům zasílané emailem.
- Prodlužování služeb po úhradě dokladu za službu - vydané doklady, které jsou vystaveny na základě fakturace služeb, jsou po jejich úhradě zodpovědné za automatické prodlužování těchto služeb o nastavené fakturační období.

**Uživatelé:**

- Přihlášení do systému – uživatelé se musí do systému přihlásit zadáním uživatelského jména a jejich hesla.
- Odhlášení ze systému – přihlášený uživatel se může kdykoliv ze systému odhlásit.
- Změna hesla – přihlášený uživatel může změnit své heslo zadáním svého původního hesla a nového hesla, které je pro kontrolu vyžadováno dvakrát.
- Obnova zapomenutého hesla – nepřihlášený uživatel může po zadání svého uživatelského jména požádat o vygenerování odkazu pro přiřazení nového hesla. Vygenerovaný odkaz je odeslán na emailovou adresu uživatele. Po otevření vygenerovaného odkazu je na emailovou adresu odesláno nové heslo.

- Výpis uživatelů systému – tabulkové zobrazení obsahující uživatelské jméno, aktuální kontaktní údaje a uživatelskou roli. Výpis uživatelů systému lze filtrovat podle uživatelských rolí.
- Vytvoření nového uživatele – formulář pro vytvoření nového uživatele, přidělení uživatelské role a zadání kontaktních údajů. Uživateli je možné přiřadit bankovní účet a heslo, popř. je možné nechat heslo náhodně vygenerovat a zaslat uživateli emailem.
- Editace uživatele – podobně jako při vytváření nového uživatele lze pomocí stejného formuláře uživatelské údaje editovat.
- Změna fakturačních a kontaktních údajů – každému uživateli lze přiřadit aktuální kontaktní údaje, popř. lze kontaktní údaje zvolit z adresáře.
- Různá uživatelská oprávnění – uživatelé jsou rozděleni do několika skupin podle jejich uživatelských rolí v systému. Je nutné odlišit zákazníky, administrátory systému a dodavatele.

**Bankovní transakce:**

- Výpis příchozích bankovních transakcí – tabulkové zobrazení obsahující datum přijetí transakce, číslo účtu odesílatele, číslo účtu příjemce, částku v měně účtu, částku v Kč, variabilní symbol a ostatní dostupné údaje o transakci. Výpis lze filtrovat podle stavu přiřazení transakce k dokladu.
- Automatické ukládání příchozích transakcí – ukládání příchozích transakcí do systému probíhá na základě propojení s bankovní institucí pomocí dostupných technologií.
- Párování transakcí s vydanými doklady – transakci je možné přiřadit manuálně k vydanému dokladu nebo je párování zajištěno automaticky v případě shody částky a variabilního symbolu.
- Manuální párování transakcí s přijatými doklady – transakci je možné přiřadit manuálně k přijatému dokladu.
- Označení transakce jako interní – příchozí bankovní transakci je možné označit jako interní.

**Adresář:**

- Výpis kontaktních údajů – tabulkové zobrazení obsahující název firmy nebo jméno a příjmení, adresu, IČ, DIČ, telefon, email a webovou adresu.
- Přidání nového kontaktního údaje – formulář pro přidání nových kontaktních údajů do adresáře. Kontaktní údaje lze přiřadit konkrétnímu uživateli systému. V případě potřeby lze kontaktní údaje dohledat podle IČ pomocí systému ARES.
- Automatické sběr kontaktních údajů – při zadávání kontaktních údajů při vytváření dokladů a registraci uživatelů jsou kontaktní údaje automaticky ukládány do adresáře.

**Statistiky:**

- Finanční tok (cash flow) – na základě dat o úhradách přijatých a vydaných dokladů je zobrazována denní, týdenní, měsíční a roční statistika finančního toku. Statistická data jsou zobrazována ve formě tabulky a v grafické podobě.
- Předpokládaný finanční tok – na základě dat o pohledávkách a závazcích je zobrazována statistika předpokládaného finančního toku. Statistická data jsou zobrazována ve formě tabulky.
- Finanční statistiky uživatelů – na základě uživatelům přiřazených dokladů jsou zobrazovány statistické údaje o uživateli. Statistická data jsou zobrazována ve stejném zobrazení jako statistické údaje samotného ekonomického subjektu.
- U plátců DPH orientační tok DPH – orientační výpočet daně z přidané hodnoty na základě dokladů v systému.

**Nastavení systému:**

- Nastavení označení systému a systémového emailu – administrátor systému může nastavit libovolné označení systému (např. název firmy) a emailovou adresu, která bude použita jako údaj o odesílateli u systémových zpráv.
- Editace číselných řad dokladů včetně jejich formátu – administrátor může nastavit libovolný formát číselné řady se zachováním číselného formátu pro automatickou

inkrementaci. Lze nastavit číselnou řadu pro vydané faktury, pro vydané zálohové faktury a pro příjmové pokladní doklady.

- Editace patičky vydaných dokladů – lze nastavit libovolnou patičku pro vystavené doklady.
- Generování API klíče – pro přístup k údajům ze systému pomocí webového API lze vygenerovat API klíč k autorizaci.
- Editace platebních metod na dokladech – lze nastavit libovolné platební metody, které jsou po té použity jako formy úhrady u dokladů. Ve výchozím nastavení musí být dostupná forma úhrady bankovním převodem a hotově.
- Správa bankovních účtů – administrátor může nastavit libovolné množství bankovních účtů. Při výběru bankovní instituce je informován o možnostech automatického datového propojení s bankovní institucí.
- Editace kurzů měn – kurzy jednotlivých měn lze nastavit manuálně pro každou měnu nebo je možné aktualizovat kurzy automaticky dle ČNB. Kurz může být aktualizován každý den nebo může platit vždy kurz platný k poslednímu dni daného měsíce v celém následujícím měsíci.
- Uložení vlastního loga subjektu – administrátor systému může nastavit systémové logo zobrazené v uživatelském rozhraní a může uložit logo pro vydané doklady.
- Uložení razítka pro vydané doklady – administrátor může uložit do systému naskenované razítko, které je později automaticky přidáváno na vydané doklady.
- Plátce/neplátce DPH – systém musí umožnit libovolný přechod z plátce DPH na neplátce a naopak.

#### **Ostatní:**

- Ověření plátce DPH podle DIČ – systém musí umožnit ověření plátce DPH podle DIČ za využití systému VIES.
- Výpis aktuálních účetních kurzů měn v systému – administrátor může zobrazit aktuálně platné účetní kurzy nastavené v systému. Tyto kurzy jsou automaticky ukládány u dokladů v cizích měnách.

- Automatické zaznamenávání událostí v systému – systém musí zaznamenávat tzv. události, které souvisí s funkčními požadavky systému a obsahují datum a čas vzniku události, uživatele, který je spojen s danou událostí, a popis samotné události.

#### 4.2.2 Nefunkční požadavky

Nefunkční požadavky nesouvisí přímo s funkcemi informačního systému z pohledu uživatele, nicméně se jedná o velmi důležitou součást analýzy požadavků s cílem vyvinout kvalitní a stabilní informační systém. Nefunkční požadavky na webový ekonomický informační systém jsou:

- webový typ aplikace,
- webový server Apache, programovací jazyk PHP, databázový systém MySQL,
- paralelní provoz více systémů v rámci jedné aplikace,
- intuitivní jednoduché ovládání,
- bezpečnost (autentizace, autorizace, šifrování),
- rozšiřitelnost (budoucí rozšíření funkcionality, objektový model),
- česká lokalizace systému.

#### 4.3 Definice aktérů systému

Z konceptuálního popisu systému vyplývá několik uživatelských rolí:

- Nepřihlášený uživatel
- Přihlášený uživatel
  - Zákazník
  - Administrátor
  - Dodavatel

Téměř veškeré funkce systému budou přístupné pouze přihlášeným uživatelům. Nepřihlášený uživatel může přistoupit k systému pouze za účelem přihlášení do systému a



je mu umožněno zaslání nového hesla na jeho registrační email v případě, že své původní heslo zapomněl.

U přihlášených uživatelů je problematika mnohem složitější a je nutné uživatele rozdělit do několika skupin podle jejich oprávnění v systému. Společnou funkcí pro všechny přihlášené uživatele je možnost odhlášení ze systému a změna hesla pro přístup do systému.

#### **4.3.1 Zákazník**

Zákazník může v systému editovat své fakturační údaje pro účely vystavování faktur, např. za poskytnuté služby podnikatelským subjektem. Jelikož bude systém umožňovat periodickou fakturaci služeb, může zákazník zobrazit veškeré své služby a zkontrolovat jejich stav. Může tedy snadno zjistit, na jaké období má danou službu uhrazenou, kolik ho daná služba bude stát financí, rychle dohledat např. poslední fakturu za konkrétní službu nebo dohledat neuhrazené služby. Zákazník může rovněž editovat délku fakturačního období služeb, takže mu můžou být služby fakturovány s různou periodou, např. měsíčně, čtvrtletně, ročně apod.

Zákazník může samozřejmě zobrazit veškeré faktury, které byly podnikatelským subjektem vystaveny a přiřazeny k jeho uživatelskému účtu (na fakturách je zákazník veden jako odběratel). U těchto faktur může kontrolovat stav úhrady a zobrazit základní informace o platbě, popř. si může faktury uložit pro své účely ve formátu PDF nebo je exportovat ve formátu XML pro účetní software Money S3.

#### **4.3.2 Administrátor**

Klíčovou rolí celého systému bude administrátor. Jedná se o uživatele, který má oprávnění k veškerým úkonům spojených s ekonomickým informačním systémem. V podstatě se bude jednat o uživatele pověřeného podnikatelským subjektem k administrátorským činnostem. U menších firem a živnostníků se zpravidla bude jednat pouze o jediného uživatele, který bude administrátorské úkony vykonávat. Návrh systému ale uvažuje i větší podnikatelské subjekty, které mají větší množství zaměstnanců, a činnost spojenou s administrací může ve firmě provádět více lidí včetně samotného majitele.

Administrátor má přístup k evidenci vydaných i přijatých dokladů, může doklady vytvářet, editovat je a provádět další operace s doklady (např. export do Money S3, uzamčení, storno apod.). Tato uživatelská role má přístup ke kompletním statistikám, může aktualizovat kurzy měn, nastavovat a přiřazovat fakturační údaje ostatním uživatelům, ověřovat platnost DIČ ostatních plátců DPH nebo může realizovat migraci systému z neplátce DPH na plátce a naopak. Administrátor může vytvářet nové kontakty v adresáři, může generovat datové soubory v ABO formátu pro úhrady závazků, může označovat doklady jako uhrazené, je mu umožněno procházet bankovní transakce a má přístup k editaci služeb zákazníků. Pouze administrátor může v systému registrovat nové uživatele a přidělovat jim uživatelské role. Administrátor dále provádí kompletní nastavení systému, např. vytváření bankovních účtů, zakládání nových typů plateb, editaci formátu číselných řad dokladů a nastavení vzhledu dokladů přidáním loga nebo razítka. Při budoucím rozšíření systému budou veškeré nově vzniklé funkce systému umožněny s největší pravděpodobností právě administrátorům.

#### 4.3.3 Dodavatel

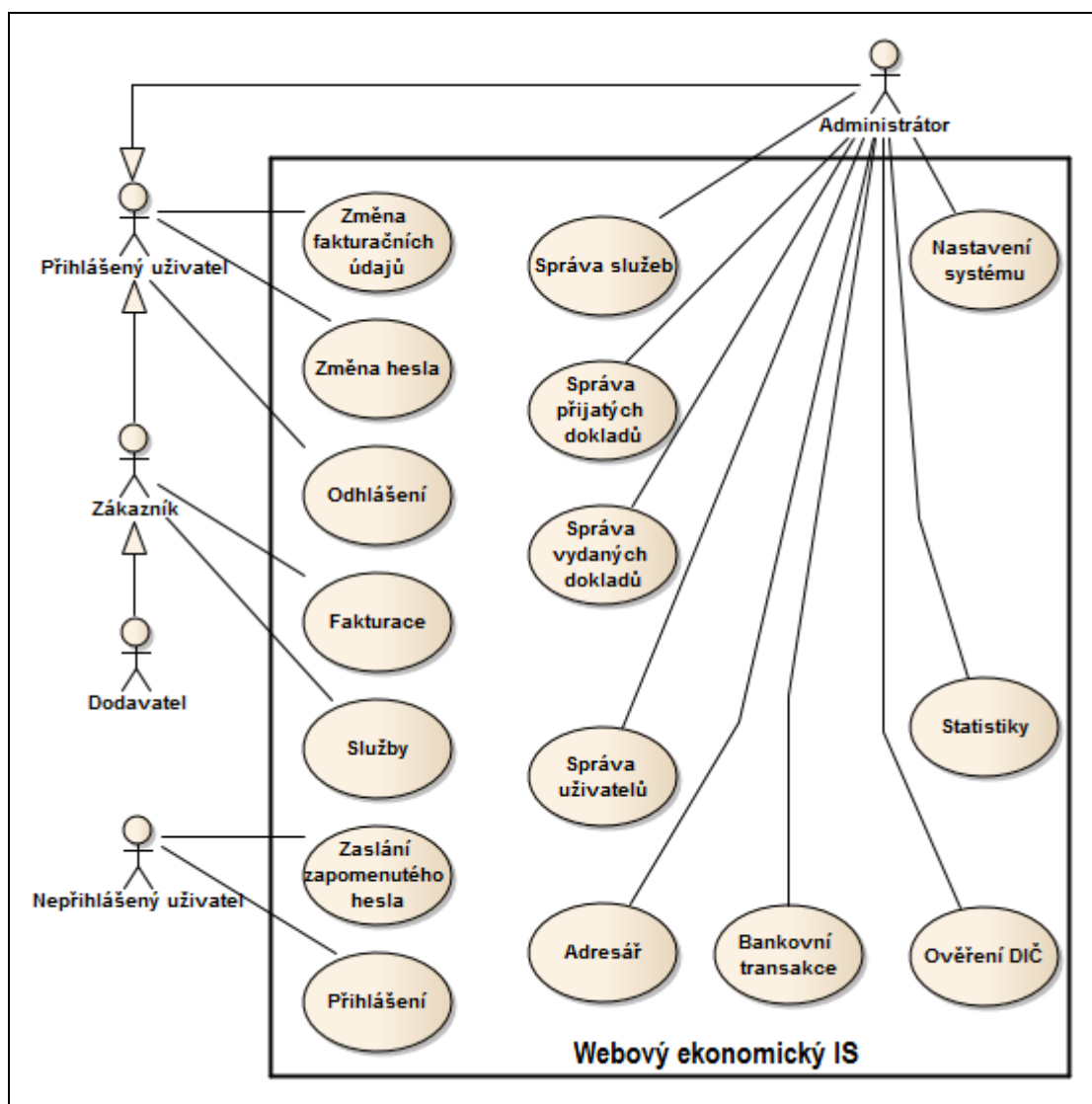
Do systému budou mít přístup i dodavatelé. V návrhu systému je s dodavateli kalkulováno, jejich funkce v systému však bude definována až v budoucnu, v současné fázi návrhu systému je k dodavatelům přístupováno stejně jako k zákazníkům, protože sami dodavatelé mohou být zároveň zákazníky podnikatelského subjektu. V budoucnu by mohli dodavatelé např. zobrazit své evidované doklady (přijaté doklady přiřazené administrátorem k dodavateli) či zkontrolovat stav jejich úhrad, což je v současném návrhu systému pouze jedno z možných rozšíření.

#### 4.4 Model případů užití

V předchozí kapitole byli definováni aktéři systému a jejich hlavní aktivity v systému. Zjednodušený model případů užití (*Use case diagram*) na obrázku (Obr. 4) určuje funkční strukturu webového ekonomického informačního systému z pohledu definovaných aktérů a jejich uživatelských oprávnění a model je tak primárně určen k definici chování systému vůči aktérům, aniž by odhaloval vnitřní strukturu samotného systému.

Z modelu je patrné, že dodavatelé mají v současném návrhu systému stejnou funkci jako zákazníci, v návrhu systému jsou ale záměrně odděleni kvůli pravděpodobnému budoucímu rozšíření informačního systému o nové funkce.

Jak již bylo řečeno, celý model je zobecněný a zjednodušený. Model zobecňuje hlavní skupiny případů užití, aby objasnil uživatelská oprávnění uživatelských skupin (aktérů). Hlavní skupiny případů užití by v podrobném modelu byly za použití generalizace dále členěny na jednotlivé podrobnější případy užití, které byly definovány v předchozí kapitole v podkategoriích funkčních požadavků na webový ekonomický informační systém. Pro znázornění nejdůležitějších uživatelských oprávnění je však tento zjednodušený model zcela dostačující.



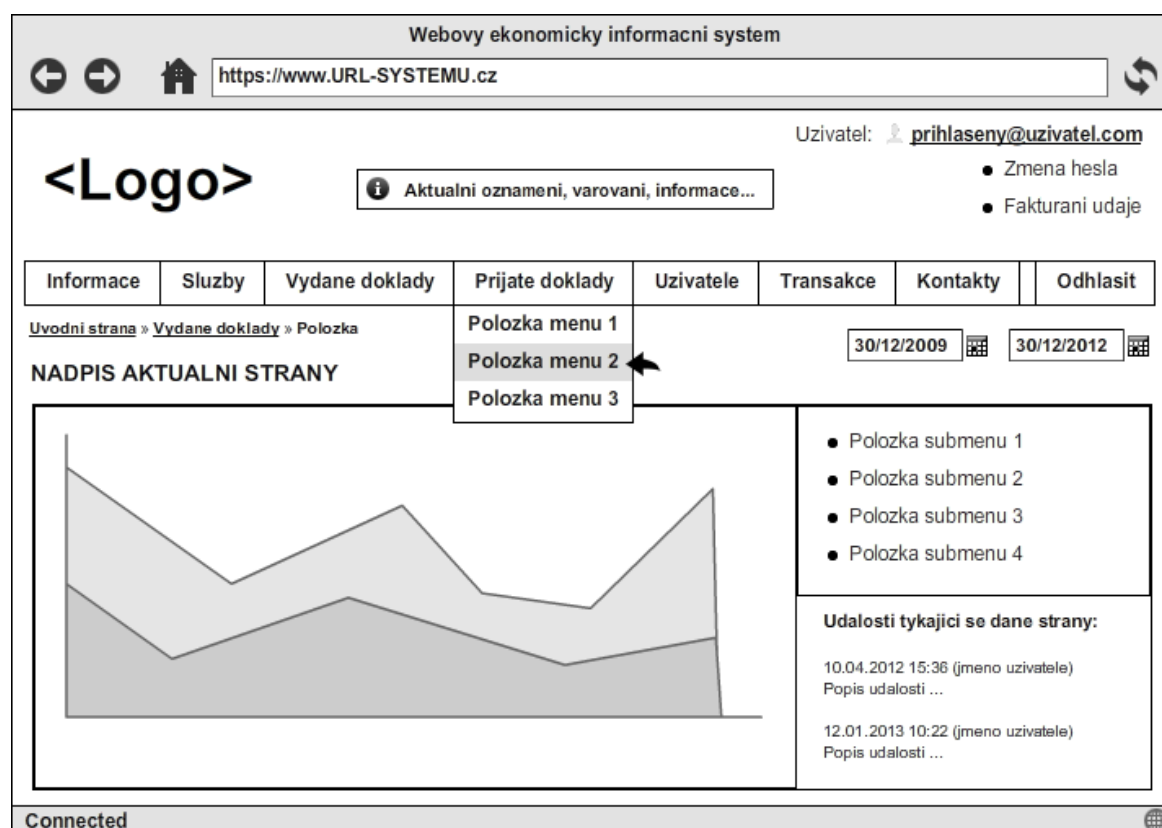
Obr. 4. Zjednodušený model případů užití

## 4.5 Návrh uživatelského rozhraní

Uživatelské rozhraní, resp. GUI (*Graphical User Interface*) definuje způsob ovládání a formu přístupu k systému pomocí interaktivních ovládacích prvků. V případě webového ekonomického informačního systému se bude jednat o webové rozhraní.

Uživatel bude používat standardní vstupní webové prvky, konkrétně elementy jako jsou tlačítka, formuláře, menu, ikony apod. Výstupem systému bude zpravidla webová stránka v HTML/CSS, která bude zobrazena přímo ve webovém prohlížeči, popř. prohlížeč zobrazí výzvu k uložení datového souboru. Pro pohodlnou práci budou použity doplňující webové technologie jako je JavaScript, jQuery, AJAX apod.

Jak již vyplývá z předchozí kapitoly, kde byli definováni aktéři systému, webový ekonomický informační systém bude různým způsobem přistupovat k přihlášeným a nepřihlášeným uživatelům. Aby bylo na první pohled zřejmé, zda je uživatel přihlášen či nikoliv, bude to vhodné odlišit již pomocí uživatelského rozhraní. Návrh webového uživatelského rozhraní pro přihlášené uživatele je znázorněn na obrázku níže (Obr. 5).



Obr. 5. Návrh uživatelského rozhraní – přihlášený uživatel

Administrátoři systému budou mít možnost upravit barevné schéma webového grafického uživatelského rozhraní z důvodu sjednocení barevného designu s firemním designem podnikatelských subjektů. K dispozici bude modré, zelené a oranžové barevné provedení, v budoucnu může být počet barevných schémat rozšířen. Administrátor tak zvolí nejlépe vyhovující barevné provedení uživatelského rozhraní, aby co nejvíce odpovídalo korporátní identitě a zákazníci nebyli zmateni z nejednotného vizuálního stylu.

K dalším administrátorským funkcím systému bude patřit možnost nastavení vlastního loga, které bude zobrazeno v levém horním rohu uživatelského rozhraní přihlášených uživatelů a v případě GUI nepřihlášených uživatelů bude zobrazeno nad přihlašovacím formulářem. Návrh uživatelského rozhraní pro nepřihlášené klienty je na následujícím obrázku (Obr. 6). Z návrhu je jasné patrné, že funkce systému budou pro nepřihlášené uživatele značně omezeny a grafické rozhraní tedy bude velmi zjednodušeno.

Webový ekonomický informační systém

https://www.URL-SYSTEMU.cz

<Logo>

Přihlasení uživatele do systému

**i** Aktualní oznámení, varování, informace...

Uživatelské jméno:

Heslo:

[Zapomenuté heslo](#)

Connected

Obr. 6. Návrh uživatelského rozhraní – nepřihlášený uživatel

Aby byla práce s informačním systémem opravdu intuitivní, bude uživatelské rozhraní využívat nejrůznějších ovládacích prvků pro usnadnění práce.

K výběru položek do vydaných dokladů bude vhodné použít tzv. „našeptávač“, jehož funkcionality bude spočívat v tom, že zatímco bude uživatel psát text položky, systém mu

bude poskytovat nápovědu ve formě výběru z několika již dříve použitých položek, které budou odpovídat již napsané části textu. V případě, že bude uživatel v psaní textu pokračovat, nabízený výběr historických položek se bude neustále zužovat. K zajištění této funkcionality bude sloužit webová technologie AJAX za podpory knihovny jQuery, která bude data načítat na základě napsaného textu položky z databázového systému a zobrazovat ve formě výběrového pole.

Jelikož bude nutné ve formulářích webového ekonomického informačního systému velmi často pracovat s datem, bylo by odpovídající volit datum pohodlným způsobem, snazším než pouhým psaním textu, což by bylo v řadě případů velmi zdlouhavé. K těmto účelům se nabízí využití intuitivního dynamického kalendáře, který se zobrazí po výběru pole pro zadání data. Ukázka takového ovládacího prvku je znázorněna na obrázku (Obr. 7).



Obr. 7. Ovládací prvek – výběr data

## 4.6 Definice analytických tříd

Třída (*Class*) je definice pro množinu podobných objektů. Třída zajišťuje chování objektů a specifikace atributů, jedná se tedy o určitou abstrakci nad objekty. Objekt je potom instance příslušné třídy. Třidu lze tedy chápat jako šablonu pro vytváření objektů.

Analytické třídy mapují pojmy řešené problematiky na abstraktní entity modelu (třídy). Analytické třídy tedy představují abstraktní pohled na systém, slouží pro identifikaci entit v řešené doméně a vztahů mezi nimi, jsou definovány na vysoké úrovni abstrakce. Tyto

třídy potom mohou být dále rozpracovány do návrhových tříd. Každá analytická třída musí zachytit určitý element z domény a musí mít určitou soudržnost.

Analytické třídy webového ekonomického informačního systému jsou:

- **Bankovní účet** – uchovává informace o bankovních účtech, např. předčísli účtu, číslo účtu, kód banky, název banky, měnu účtu apod.
- **Bankovní transakce** – nese informace o bankovní transakci, tedy zejména částku, symboly, odesílatele transakce apod.
- **Vydaný doklad** – např. daňový doklad nebo pokladní příjmový doklad, jedná se o nejpodstatnější část systému, jedná se tedy o klíčovou analytickou třídu.
- **Přijatý doklad** – doklady přijaté od jiných dodavatelů, součástí může být i soubor (naskenovaný doklad).
- **Sazba DPH** – uchovává informace o sazbě DPH podle zákona, tedy hlavně procentuální výši daně, název sazby apod.
- **Platební metoda** – existuje více platebních metod, např. hotovostní platba, bankovní převod, platba platební kartou apod.
- **Uživatel** – jedná se o uživatele systému, který k němu přistupuje.
- **Uživatelská role** – definuje aktéry systému, kteří mají v systému určitá předem definovaná uživatelská oprávnění a mají v systému různou funkci a možnosti.
- **Kontakt** – z kontaktů vzniká adresář, uchovává jméno nebo název firmy, adresu, telefon, email, web, IČ, DIČ apod.
- **Služba** – slouží zejména k periodické fakturaci a k přehledné evidenci veškerých služeb subjektů.
- **Měna** – uchovává informace o názvu, symbolu a kódu měny, nese s sebou aktuální účetní kurz měny v systému.
- **Událost** – zachycuje veškeré dění v systému, jde spíše o třídu informativního charakteru, uchovává nejčastěji čas a popis události.

## 4.7 Návrh zabezpečení systému

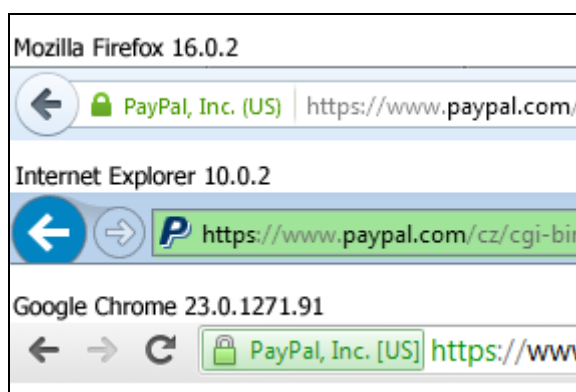
Ekonomické aktivity firem vyžadují samozřejmě vysoké zabezpečení. K základním požadavkům na webový ekonomický systém tak beze sporu patří mnoho bezpečnostních opatření ze strany poskytovatele webových služeb. Nutností se stává ochrana před nežádoucím přístupem k uživatelským a firemním datům.

### 4.7.1 Zabezpečení síťového provozu

Základem zabezpečení webového informačního systému musí být jednoznačně samotný síťový provoz. Systém bude provozován na webovém serveru Apache, který nám v základním sestavení poskytne HTTP přenosy veškerých dat, ovšem v nešifrované podobě.

Na webovém serveru tedy bude nutné zajistit nadstavbu HTTPS a šifrování pomocí SSL. Základem bude využití Apache modulu „mod\_ssl“ a digitální certifikát (podepsaný veřejný klíč) ověřený důvěryhodnou certifikační autoritou. Každá certifikační autorita vydává vlastní druhy certifikátů, které se liší v účelu použití, úrovni důvěryhodnosti atd. Digitální certifikáty vydávané komerčními firmami (např. VeriSign, Thawte, ipsCA apod.), jsou placené, bude s tím tedy nutné kalkulovat pro budoucí provoz systému.

Veškerá komunikace s informačním systémem tedy musí být šifrovaná. Uživatelé se na klientských počítačích musí vždy ujistit, že jsou k serveru bezpečně připojeni přes HTTPS protokol a došlo k řádnému ověření certifikační autoritou. Následující obrázek (Obr. 8) znázorňuje adresní řádky nejznámějších webových prohlížečů při ověřeném šifrovaném spojení přes protokol HTTPS/SSL.



Obr. 8. Šifrované spojení v prohlížečích



#### 4.7.2 Zabezpečení klientské části

Webový systém bude data reprezentovat převážně v HTML s využitím JavaScriptu a pomocí již dříve představených technologií (jQuery, AJAX). Klientská stanice tedy bude muset být vybavena webovým prohlížečem.

Jak již bylo zmíněno, uživatelé (klienti) jsou o šifrovaném síťovém provozu informováni prohlížečem. Je proto důležité, aby si každý z uživatelů tuto skutečnost sám ověřil. Základním předpokladem je rovněž udržování aktuální verze webového prohlížeče, mít aplikovány aktuální záplaty operačních systémů a mít klientskou stanici zajištěnou proti případným počítačovým virům, které by mohly data zneužít až po dešifrování přímo na klientské stanici.

Uživatele systému bude nutné před přístupem k datům autentizovat. Autentizace slouží k jednoznačné definici uživatele, který přistupuje k informačnímu systému. Cílem procesu autentizace je zajistit, že informační systém přesně ví, s jakým uživatelem komunikuje.

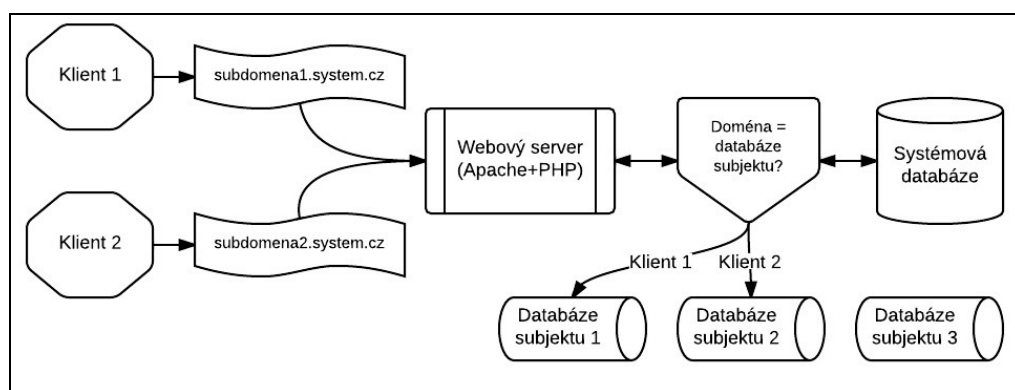
Nejrozšířenější a nejvyužívanější autentizační technikou je autentizace pomocí hesel, při které uživatel zadává své přihlašovací jméno (nejčastěji emailová adresa) a heslo. Kvalitu hesla zvoleného uživatelem určuje počet znaků abecedy a délka hesla. Uživatel tedy předkládá systému heslo, které je řetězcem znaků společně se svou identifikací. Doporučovaným způsobem pro zvyšování bezpečnosti hesla je obměňování základní množiny znaků a jejich kombinací. Jak již bylo řečeno, ještě před odesláním přihlašovacích údajů je nutné zkontrolovat důvěryhodnost šifrovaného spojení, v opačném případě by mohlo dojít k odposlouchávání těchto přihlašovacích údajů. Heslo je na serveru uloženo v hashované podobě, aby nemohlo být zneužito.

Bezprostředně po úspěšném procesu autentizace následuje proces autorizace. Autorizace je proces zakazování nebo přidělování přístupu k datům pro uživatele či skupiny uživatelů (uživatelské role) a definování toho, jaké operace s daty mají povoleno provádět. U webového ekonomického IS bude proces autorizace a logiku přístupu k datům zajišťovat především programovací jazyk PHP ve spolupráci s databázovým systémem MySQL.

#### 4.7.3 Zabezpečení datové části

Webový ekonomický IS bude podporovat paralelní provoz více systémů na oddělených MySQL databázích několika ekonomických subjektů. Samotná aplikace bude navržena tak,

aby byla realizována neduplicitními zdrojovými kódy, resp. veškeré paralelní systémy budou provozovány v rámci jedné aplikace s napojením na správnou databázi autorizovaného ekonomického subjektu (na různých doménách). Tento přístup bude mimo jiné zaručovat budoucí automatické aktualizace všech systémů v jednom kroku, jelikož provozovatel bude aktualizovat pouze jednu verzi zdrojových kódů aplikace. Obrázek (Obr. 9) je zaměřen na bližší objasnění celé situace přístupu k odděleným databázím a autentizaci jednotlivých systémů k jejich databázím podle domén. Dále bude nutné zabezpečit data fyzicky, čímž je myšleno bezpečné fyzické uložení serverů v zabezpečeném datacentru pod dohledem.



Obr. 9. Schéma přístupu ke korektní databázi ekonomického subjektu

## 5 IMPLEMENTACE SYSTÉMU

Před zahájením popisu implementace informačního systému je důležité podotknout, že stávající webový ekonomický informační systém se jako takový stále vyvíjí a s největší pravděpodobností je již v této době implementace systému rozšířenější, než je v této práci popsáno. Na vývoji systému a jeho postupném rozšiřování spolupracují podnikatelské subjekty, které systém již v dnešní době používají v reálném provozu a informační systém je postupně rozšiřován dle jejich potřeb a zejména dle jejich dynamicky se rozšiřujícího působení na trhu. Systém tedy splňuje nefunkční požadavek na rozšiřitelnost, která byla v definici nefunkčních požadavků velmi podstatná. Samotná legislativa v ČR týkající se účetnictví je velmi často aktualizována a téměř každý rok prochází řadou změn, do budoucna je tedy zřejmé, že bude systém nutné neustále udržovat a aktualizovat dle úprav v zákonech.

### 5.1 Specifikace technologií

- Webový server: **Apache HTTP server** (min. 2.0)
- Programovací jazyk: **PHP** (min. 5.2)
- Databázový systém: **MySQL** (min. 5.1)

### 5.2 Architektura MVC

Architektura MVC (*Model-View-Controller*) je forma přístupu k realizaci systému nebo aplikace, kdy je samotná implementace rozdělena na 3 oddělené logické části takovým způsobem, aby šlo tyto části editovat samostatně a při tom byl dopad změn na ostatní části minimální. Jak již název napovídá, jedná se o tyto části:

- Model – reprezentuje třídy systému a uložení dat,
- View – zobrazuje uživatelské rozhraní,
- Controller – zajišťuje tok událostí v aplikaci a obecně aplikační logiku.

Webový ekonomický IS je implementován právě na základě architektury MVC, popis implementace jednotlivých logických částí je uveden v následujících podkapitolách.

### 5.2.1 Model

Logická část „Model“ je v systému implantována technikou ORM (*objektově relační mapování*), což je programovací technika, která v implementaci systému zajišťuje automatickou konverzi dat mezi objektovými třídami (entitami) a databázovým systémem za použití programovacího jazyka PHP.

PHP třídy jsou obecně uloženy v adresáři, ze kterého jsou do PHP skriptů automaticky načítány (není nutné je „includovat“ do skriptů, tzv. „autoloading“). Jedná se o adresář:

```
/include/classes/
```

Objektové třídy jsou uloženy přímo v tomto adresáři v oddělených souborech, které jsou vždy pojmenovány dle následující syntaxe:

```
[NAZEV_OBJEKTU].class.php
```

Díky tomuto názvosloví lze snadno dohledat zdrojový kód k třídě a naopak je z názvu souboru ihned jasné, jak se jmenuje samotná třída.

Třídy zajišťující konverzi dat mezi objektovými třídami a MySQL za využití PHP frameworku Dibi (tzv. „mappery“) jsou univerzálně pojmenovány podle objektových tříd s následující syntaxí:

```
DB_[NAZEV_OBJEKTU]Mapper
```

Soubory s „mappery“ mají podobné názvosloví jako objektové třídy (mají tedy koncovku „class.php“) a jsou uloženy v adresáři:

```
/include/classes/DB/
```

Jako příklad může sloužit objektová třída „BankovniUcet“:

- Objektová třída: BankovniUcet
- Zdrojový kód třídy: /include/class/BankovniUcet.class.php
- Třída „mapperu“: DB\_BankovniUcetMapper
- Zdrojový kód třídy: /include/class/DB/DB\_BankovniUcetMapper.class.php

V tabulce (Tab. 5) jsou uvedeny nejdůležitější implementované PHP objektové třídy a jejich popis.

Objektová třída	Popis
BankovniUcet	Reprezentuje bankovní účet na vydaném dokladu nebo ve spojení s uživatelem.
CiselnaRada	Číselná řada na vydaných dokladech.
Event	Zahrnuje veškeré události v systému.
Kontakt	Uchovává informace o kontaktních údajích.
Login	Zajišťuje proces autorizace a autentizace.
Mena	Reprezentuje finanční měnu.
ObrazekFaktura	Obrázky na vydaných dokladech, např. razítko.
Paticka	Libovolně definovatelná patička na fakturách.
Polozka	Reprezentuje položku vydaného dokladu.
PolozkaFaktura	Spojeno s položkou vydaného dokladu, určuje množstevní jednotku, cenu, sazbu DPH apod.
PrijatyDoklad	Uchovává informace o přijatých dokladech.
SazbaDph	Označuje sazbu DPH včetně její aktuální výše.
Sluzba	Periodicky fakturovaná služba.
SluzbaUser	Služba ve spojení s konkrétním uživatelem.
Soubor	Reprezentuje jakýkoliv datový soubor.
System	Reprezentuje systém daného subjektu.
Transakce	Automaticky přijatá platba na bankovním účtu.
TypPlatby	Označuje formu úhrady přijatých i vydaných dokladů.
User	Reprezentuje uživatele systému.
VydanyDoklad	Reprezentuje vydané doklady.

Tab. 5. Nejdůležitější implementované objektové třídy

V tabulce je uveden přehled nejvýznamnějších databázových tabulek, ze kterých je v systému tvořena struktura MySQL databáze ekonomického subjektu.

Tabulka	Popis
bankovni_ucet	Udrží informace o bankovních účtech.
ciselna_rada	Data o číselných řadách vydaných dokladů.
event	Zahrnuje data o událostech v systému.
faktura	Reprezentuje vydané doklady.
faktura_bankovni_ucet	Spojuje bankovní účty s vydanými doklady.
kontakt	Uchovává informace o kontaktních údajích.
mena	Reprezentuje finanční měnu.
nastaveni	Nastavení systému daného ekonomického subjektu.
obrazek	Obrázky na vydaných dokladech, např. razítko.
obrazek_faktura	Propojení obrázku s odpovídajícím dokladem.
paticka	Libovolně definovatelná patička na fakturách.
polozka	Reprezentuje položku vydaného dokladu.
polozka_faktura	Spojení položky vydaného dokladu s dokladem.
prijaty_doklad	Reprezentuje přijaté doklady.
prijaty_doklad_soubor	Soubory přiložené k přijatým dokladům.
sazba_dph	Označuje sazbu DPH včetně její aktuální výše.
sluzba	Periodicky fakturovaná služba.
sluzba_user	Služba ve spojení s konkrétním uživatelem.
soubor	Reprezentuje jakýkoliv datový soubor.
transakce	Automaticky přijatá platba na bankovním účtu.
typ_platby	Označuje formu úhrady přijatých i vydaných dokladů.
user	Reprezentuje uživatele systému.

Tab. 6. Nejvýznamnější MySQL databázové tabulky

### 5.2.2 View

Logická část „View“ reprezentuje zejména zobrazení uživatelského rozhraní a všeobecně přístup k veškerým výstupům systému. V případě ekonomického IS se jedná o webovou aplikaci, která byla v implementaci systému realizována převážně šablonovacím systémem Smarty, což je komplexní šablonovací systém pro programovací jazyk PHP. V implementaci systému slouží k oddělení uživatelského rozhraní od samotného zpracování dat a od zpracování aplikační logiky.

Šablony pro Smarty mají v implementaci vždy koncovku „.tpl“ a jsou vždy uloženy v adresáři:

```
/home/templates/
```

V realizaci systému je implementována třída WebSmarty, která dědí vlastnosti třídy Smarty (originální třída šablonovacího systému). Tímto způsobem lze snadno upravovat potřebné metody a nastavení Smarty, což lze dobře využít např. při přepisu absolutních URL adres v šablonách. Za tímto účelem je ve třídě WebSmarty metodou „register\_outputfilter“ zaregistrovaná funkce „system\_hostname\_filter“, které je ještě před odesláním dat předán veškerý výstup ze Smarty. Tato funkce po té přepíše absolutní URL adresy podle doménového jména aktuálního systému a upravená data jsou odeslána na klientskou stanici.

Logický celek „View“ byl mimo Smarty dále implementován pomocí již dříve představených webových technologií (HTML, CSS, JavaScript, jQuery, jQuery UI, AJAX, PDF apod.).

Implementace grafického uživatelského rozhraní byla podle návrhu z kapitoly 4.5 vizuálně rozdělena podle uživatelských rolí v systému. V případě nepřihlášeného uživatele je nutné nejprve uživatele vyzvat k přihlášení do systému, což je dle návrhu GUI vhodné odlišit. Na obrázku (Obr. 10) lze vidět ukázkou implementace GUI v případě nepřihlášeného uživatele.

Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

## Přihlášení uživatele

✖ Neplatné uživatelské jméno nebo heslo!

Uživatelské jméno:

Heslo:

[Zapomenuté heslo](#)

Obr. 10. Implementace GUI – nepřihlášený uživatel

Na následujících dvou obrázcích je ukázka implementace části grafického uživatelského rozhraní v případě přihlášeného uživatele, v prvním případě se jedná o GUI pro administrátora systému (Obr. 11), na druhém obrázku (Obr. 12) je ukázka uživatelského rozhraní pro zákazníka. Z obrázků je na první pohled patrné zjednodušení GUI určeného pro zákazníka.

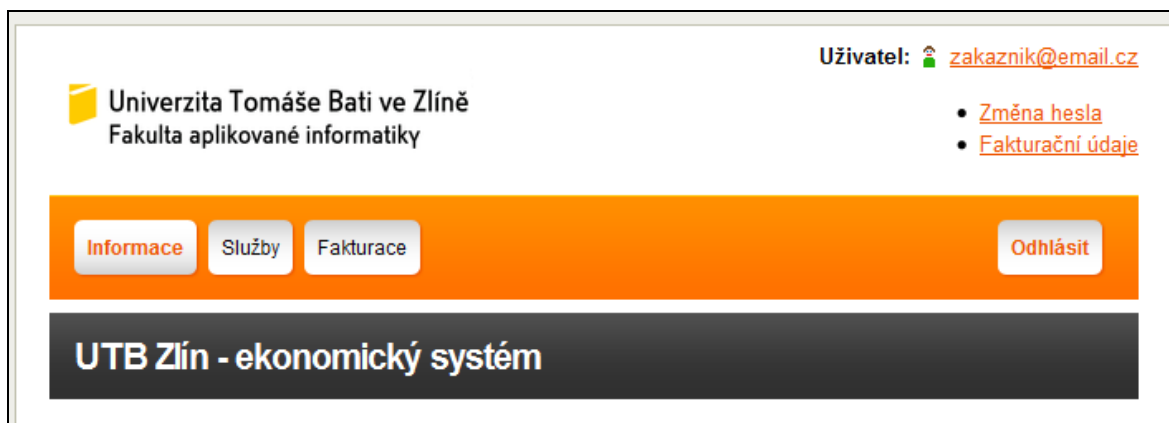
Uživatel: [test@email.cz](#)

- [Změna hesla](#)
- [Fakturační údaje](#)
- [Nastavení systému](#)

**UTB Zlín - ekonomický systém**

Obr. 11. Implementace GUI – přihlášený uživatel (administrátor)







Obr. 12. Implementace GUI – přihlášený uživatel (zákazník)

Vystavování elektronických vydaných faktur ve formátu PDF bylo implementováno za využití FPDF, což je PHP třída, která umožňuje vytvářet PDF soubory pouze pomocí PHP, bez použití knihovny PDFLib. FPDF je zdarma a třídu je možné používat bez jakéhokoliv omezení.

V implementaci ekonomického informačního systému je třída FakturaPdf, která dědí vlastnosti třídy FPDF (originální třída). Stejně jako u třídy Smarty lze tímto způsobem přepisovat potřebné metody a nastavení.

Aplikační logiku generování vydaných faktur v PDF formátu zajišťuje skript „faktura.php“, který mimo jiné zabezpečuje zobrazení a stažení faktur pouze oprávněným a přihlášeným uživatelům. Administrátor může zobrazit veškeré faktury v systému, zákazník má ale přístup pouze k dokladům, na kterých je uveden jako odběratel. Administrátor má navíc možnost vygenerovat vydanou fakturu v PDF formátu ve verzi s razítkem (pokud je razítko v systému nastaveno) nebo bez razítka. Stejně tak může administrátor ovlivnit zobrazení loga subjektu přímo na faktuře.

Ukázka elektronické faktury v PDF formátu s razítkem a logem lze vidět na následujícím obrázku (Obr. 13).

<b>FAKTURA - DAŇOVÝ DOKLAD</b>																													
<b>Dodavatel:</b> Univerzita Tomáše Bati ve Zlíně nám. T. G. Masaryka 5555 760 01 Zlín Česká republika  IČ: 70883521 DIČ: CZ70883521  Telefon: +420 576 035 221 E-mail: dekanat@fal.utb.cz Web: www.utb.cz		<b>Číslo faktury:</b> 20130005  <b>Variabilní symbol:</b> 1234  <b>Odběratel:</b> IČ: 12345678 DIČ: CZ12345678  <b>Ekonomický subjekt s.r.o.</b> Ulice 89/6 999 99 Město Česká republika																											
<b>Forma úhrady:</b> Bankovním převodem 284596357/0600 - GE Money Bank, a.s. 9876945832/8330 - Fio banka, a.s. SK			<b>Datum vystavení:</b> 05.05.2013 <b>Datum splatnosti:</b> 12.05.2013 <b>Datum usk. zdaniteln. plnění:</b> 05.05.2013																										
<b>Položky</b>	<b>Jedn.oena bez DPH</b>	<b>Počet MJ</b>	<b>Celk.oena bez DPH</b>	<b>DPH %</b>	<b>Celk.oena vč. DPH</b>																								
Školení zaměstnanců (01.04.2013 - 30.04.2013)			980,00 EUR	21 %	1.185,80 EUR																								
Náklady na dopravu			120,00 EUR	21 %	145,20 EUR																								
Studijní materiál	10,00 EUR	10 ks	100,00 EUR	21 %	121,00 EUR																								
<b>Celkem bez DPH</b>					1.200,00 EUR																								
<b>Celkem vč. DPH</b>					1.452,00 EUR																								
<b>Uhrazeno zálohově</b>					500,00 EUR																								
<b>CELKEM K ÚHRADĚ</b>					952,00 EUR																								
<div style="display: flex; justify-content: space-between;"> <div> <p><b>Rekapitulace DPH v EUR</b></p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Sazba DPH %</th> <th>Základ daně</th> <th>DPH</th> </tr> </thead> <tbody> <tr> <td>0 %</td> <td>0,00 EUR</td> <td>0,00 EUR</td> </tr> <tr> <td>15 %</td> <td>0,00 EUR</td> <td>0,00 EUR</td> </tr> <tr> <td>21 %</td> <td>1.200,00 EUR</td> <td>252,00 EUR</td> </tr> </tbody> </table> <p>Účetní kurz 1 EUR = 25,687 CZK</p> <p><b>Rekapitulace DPH v CZK</b></p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Sazba DPH %</th> <th>Základ daně</th> <th>DPH</th> </tr> </thead> <tbody> <tr> <td>0 %</td> <td>0,00 Kč</td> <td>0,00 Kč</td> </tr> <tr> <td>15 %</td> <td>0,00 Kč</td> <td>0,00 Kč</td> </tr> <tr> <td>21 %</td> <td>30.824,40 Kč</td> <td>6.473,12 Kč</td> </tr> </tbody> </table> </div> <div style="text-align: center;">  </div> </div> <p style="font-size: small; margin-top: 20px;">Firma zaps. do obch. rejstříku u rejstřík. soudu ve Zlíně, oddíl C, vč. 6600X.</p>						Sazba DPH %	Základ daně	DPH	0 %	0,00 EUR	0,00 EUR	15 %	0,00 EUR	0,00 EUR	21 %	1.200,00 EUR	252,00 EUR	Sazba DPH %	Základ daně	DPH	0 %	0,00 Kč	0,00 Kč	15 %	0,00 Kč	0,00 Kč	21 %	30.824,40 Kč	6.473,12 Kč
Sazba DPH %	Základ daně	DPH																											
0 %	0,00 EUR	0,00 EUR																											
15 %	0,00 EUR	0,00 EUR																											
21 %	1.200,00 EUR	252,00 EUR																											
Sazba DPH %	Základ daně	DPH																											
0 %	0,00 Kč	0,00 Kč																											
15 %	0,00 Kč	0,00 Kč																											
21 %	30.824,40 Kč	6.473,12 Kč																											
Vytiskl fakturační systém TripleWeb.cz. <div style="float: right;">Strana 1/1</div>																													

Obr. 13. Ukázka implementace vydaných dokladů v PDF formátu

### 5.2.3 Controller

Logická část „Controller“ zajišťuje tok událostí v aplikaci a obecně aplikační logiku celého systému. V implementaci webového ekonomického IS se jedná o skripty pro programovací jazyk PHP. Skripty jsou uloženy v adresáři a jeho podadresářích s následující cestou, jedná se zároveň o tzv. „ServerRoot“:

```
/www/
```

Převážná část PHP skriptů nejprve provede autorizaci uživatele, po té vytvoří instanci třídy WebSmarty a začne provádět aplikační logiku, v průběhu které předávají Smarty potřebné parametry k zobrazení HTML výstupu. Pokud není uživatel úspěšně autorizován, je o tom informován a aplikační logika se nevykoná. Tím je znemožněno uskutečnit neautorizované operace, např. upravit nastavení systému podnikatelského subjektu zákazníkem. V případě, že není uživatel přihlášen, je přesměrován na přihlašovací skript, který zajišťuje zobrazení odlišného uživatelského rozhraní s přihlašovacím formulářem. Po vykonání aplikační logiky je ve skriptech volána metoda „display“, která zobrazí finální HTML výstup.

## 5.3 Zabezpečení systému proti útokům

### 5.3.1 Zabezpečení síťového provozu

Aby byl síťový provoz zabezpečen šifrováním díky protokolu HTTPS, je nutné správně nakonfigurovat webový server Apache. V první řadě je nutné povolit nahrávání SSL modulu (ve verzi Apache 2.0 a novější se jmenuje „mod\_ssl“), a to úpravou konfiguračního souboru „httpd.conf“. V souboru se musí objevit následující řádek:

```
LoadModule ssl_module [CESTA_K_MODULU]/mod_ssl.so
```

Následně je nutné pomocí webového serveru Apache založit tzv. „virtuální server“, což se provede přidáním následujících direktiv např. nakonec stejného souboru:

```
<VirtualHost *: [CISLO_PORTU_HTTPS]>
DocumentRoot [CESTA_K_APLIKACI]/www/
ServerName [DOMENA_SYSTEMU]
SSLEngine on
SSLCertificateFile /[CESTA_K_SSL]/server.pem
SSLCertificateKeyFile /[CESTA_K_SSL]/server.key
.....
</VirtualHost>
```

Pokud jsou správně nastaveny DNS záznamy (zejména A záznam s IP adresou webového serveru), aplikace se stává dostupná přes HTTPS protokol na URL adrese:

```
https://[DOMENA_SYSTEMU]:[CISLO_PORTU_HTTPS]/
```

### 5.3.2 Zabezpečení proti XSS

XSS (*Cross Site Scripting*) jsou založeny na vložení nebezpečného skriptu do obsahu stránky. Nejdůležitějším zabezpečením proti XSS je nepřipustit do veřejných částí webového systému žádné skriptovací konstrukce.

V implementaci systému jsou klíčové vstupní data obsahující znaky „<“ a „>“ překódované na HTML entity „&lt;“ a „&gt;“, čímž je efektivně zamezeno spouštění nebezpečných částí kódu později vkládaných do stránek. V případě zobrazení nezakódovaných dat přímo v HTML jsou data zobrazeny pomocí PHP funkce „htmlspecialchars“, která zajistí bezpečné zobrazení ve formě HTML entit.

### 5.3.3 Zabezpečení proti SQL injection

SQL injection je způsob napadení databázového systému vnořením příkazů přes neošetřená vstupní data a následné vykonání upraveného SQL dotazu. Nejčastěji je útok prováděn přes nesprávně zabezpečený formulář, úpravou URL adresy nebo zasláním upravené cookie.

V implementaci systému je pro přístup k databázi využita PHP knihovna Dibi, která byla blíže představena v teoretické části v kapitole 1.12. Tato knihovna mimo jiné zabezpečuje veškerá vstupní data proti SQL injection, takže je systém proti tomuto typu útoku ochráněn.

## 5.4 Paralelní provoz více systémů

Od počátečního návrhu je systém připravován pro podporu paralelního provozu více systémů v rámci jedné aplikace na oddělených databázích několika subjektů, díky čemuž je aplikace realizována neduplicitními zdrojovými kódy. Tento přístup mimo jiné zaručuje automatické aktualizace všech systémů v jednom kroku, jelikož provozovatel aktualizuje pouze jednu verzi zdrojových kódů aplikace.

K takovému provozu je zapotřebí systémová MySQL databáze, která uchovává informace o jednotlivých systémech, a jednotlivé MySQL databáze samotných podnikatelských

subjektů mající shodnou strukturu. O výběru správné databáze subjektu rozhoduje programovací jazyk PHP, který na základě doménového jména nejprve ověří v systémové databázi existenci daného systému, po korektním vyhodnocení je vráceno identifikační číslo systému a teprve po té je přistupováno do správné databáze daného podnikatelského subjektu, ve které probíhá následná autentizace a autorizace uživatelů. Tím je zamezeno neoprávněnému přístupu k databázi jiného subjektu. Schéma přístupu ke korektní databázi subjektu bylo znázorněno na obrázku (Obr. 9) v podkapitole 4.7.3.

Jednotlivé databáze musí být pojmenovány právě na základě identifikačních čísel v koncovkách názvů databází (tzv. suffixy). Počáteční část názvů databází (tzv. prefixy) mohou být různé, lze je nastavit v tomto konfiguračním souboru:

```
/include/app_top.php
```

Syntaxe pojmenování databází je tedy následující:

```
[PREFIX] [ID_SYSTEMU]
```

Název systémové databáze má pak následující syntaxi:

```
[PREFIX]admin
```

V implementaci systému je ve výchozím stavu nastaven prefix „faktury\_“, při názorném použití systému s identifikačním číslem 4 jsou tedy názvy databází tyto:

```
faktury_4  
faktury_admin
```

Systémová databáze má velmi jednoduchou strukturu obsahující pouze jednu databázovou tabulku uchovávající základní informace o systémech jednotlivých ekonomických subjektů. Tato MySQL databázová tabulka má název „system“, v tabulce (Tab. 7) je uvedena její struktura.

Název	Hodnota	Klíč	Popis
system_id	int(11)	PRIMARY	ID systému ekonomického subjektu.
domena	varchar(200)	UNIQUE	Doménové jméno pro přístup k danému systému.

Tab. 7. Struktura systémové databáze - tabulka „system“

MySQL uživatel, který se k MySQL databázi připojuje, musí mít samozřejmě oprávnění pro přístup ke všem databázím, aby byla tato implementace realizovatelná. Nastavení MySQL uživatele včetně hesla se nastavuje rovněž v souboru „app\_top.php“.

K zajištění této funkcionality je rovněž důležité správné nastavení webového serveru Apache. Nutné je pro každý systém subjektu založit tzv. virtuální server, jak již bylo uvedeno v předchozí kapitole 5.3. Z příkladu v této kapitole je nutné obměnit hodnotu direktivy „ServerName“, která nastavuje doménové jméno systému. Předpokládá se i užití doménového jména 3. řádu, kdy nejčastěji podnikatelské subjekty využijí své stávající domény i k provozu webového ekonomického informačního systému.

Pokud tedy podnikatelský subjekt vlastní a používá např. doménové jméno „ukazkovafirma.cz“, může k provozu systému využít subdoménu „fakturace.ukazkovafirma.cz“, stačí pouze správně nastavit DNS záznam pro tuto subdoménu, např.:

```
fakturace 3600 A [IP_ADRESA_SERVERU]
```

Díky tomuto A záznamu je subdoména směřována na správný server s aplikací, kde je již na základě správného nastavení virtuálního serveru vyhledáno doménové jméno v systémové databázi a je rozhodnuto o připojení k odpovídající databázi subjektu.

Pro snadný přístup uživatelů k webovému systému přes protokol HTTPS je vhodné přesměrovat HTTP požadavky přes standardní port 80 na korektní zabezpečenou URL adresu. Uživatelé systému pak stačí znát pouze danou subdoménu a webový server Apache již zajistí správné přesměrování na správnou adresu včetně čísla portu. V konfiguračním souboru Apache je nutné přidat následující direktivy:

```
<VirtualHost *:80>
DocumentRoot [CESTA_K_APLIKACI]/www/
ServerName [DOMENA_SYSTEMU]
RewriteEngine on
ReWriteCond %{SERVER_PORT} !^[CISLO_PORTU_HTTPS]$
RewriteRule ^/(.*) https://%{HTTP_HOST}:[CISLO_PORTU_HTTPS]/$1 [NC,R,L]
</VirtualHost>
```

Pro každý systém je tak možné použít odlišný port pro HTTPS protokol a není při tom nutné obtěžovat uživatele s komplikovanou URL adresou. Pokud je tedy webový systém

subjektu používán na subdoméně „fakturace.ukazkovafirma.cz“ např. přes port 9980, uživateli stačí zadat ve webovém prohlížeči URL adresu:

`http://fakturace.ukazkovafirma.cz/`

Webový server na základě korektního nastavení zajistí přesměrování na adresu systému, tedy na URL:

`https://fakturace.ukazkovafirma.cz:9980/`

Tímto způsobem je garantováno bezproblémové užití zabezpečeného síťového provozu pro všechny paralelní systémy.

## 6 TESTOVÁNÍ SYSTÉMU V REÁLNÉM PROVOZU

Realizovaný systém byl testován v reálném provozu dvou podnikatelských subjektů. První podnikatelský subjekt je živnostník – neplátce DPH (vedoucí daňovou evidenci) a druhý podnikatelský subjekt je obchodní společnost s ručením omezeným – plátce DPH (vedoucí účetnictví). Podnikatelská činnost obou subjektů je zcela odlišná, proto se podařilo systém otestovat v různých formách s využitím odlišných funkcí systému.

### 6.1 Živnostník – neplátce DPH

#### 6.1.1 Představení subjektu

Živnostník (neplátce DPH), na jehož podnikatelské činnosti byl systém testován, se zabývá zejména poskytováním služeb, které jsou fakturovány periodicky, např. webhosting, pronájem software apod. K periodickým službám fakturuje jednorázově bez pravidelnosti další občasné služby.

Živnostník používá k podnikání pouze jeden bankovní účet, převážná část úhrad jeho pohledávek probíhá právě bezhotovostně prostřednictvím tohoto bankovního účtu. Hotovostní platby přijímá subjekt v minimálním množství. Živnostník nevede účetnictví, jeho činnost vyžaduje pouze daňovou evidenci.

Před nasazením systému musel živnostník veškeré své aktivity vykonávat svépomocí, což např. u periodické fakturace zabralo velkou část pracovního času. Kontrola úhrad jeho pohledávek probíhala standardním přihlašováním do internetového bankovníctví a vyžadovala jeho denní aktivitu. Samotným vystavováním dokladů strávil týdně i několik hodin a nevyhnul se nejrozumnějším chybám, které způsobil lidský faktor.

Živnostník svůj obrat zjišťoval pouze jednou ročně spolu s daňovým přiznáním, protože v opačném případě by musel vždy přepočítávat celkové částky veškerých vydaných dokladů, popř. vést evidenci v nějakém tabulkovém procesoru, což by opět vyžadovalo další čas. Živnostník tedy neměl příliš přehled o průběžném hospodaření a jeho výsledek se dozvěděl vždy jednou ročně.



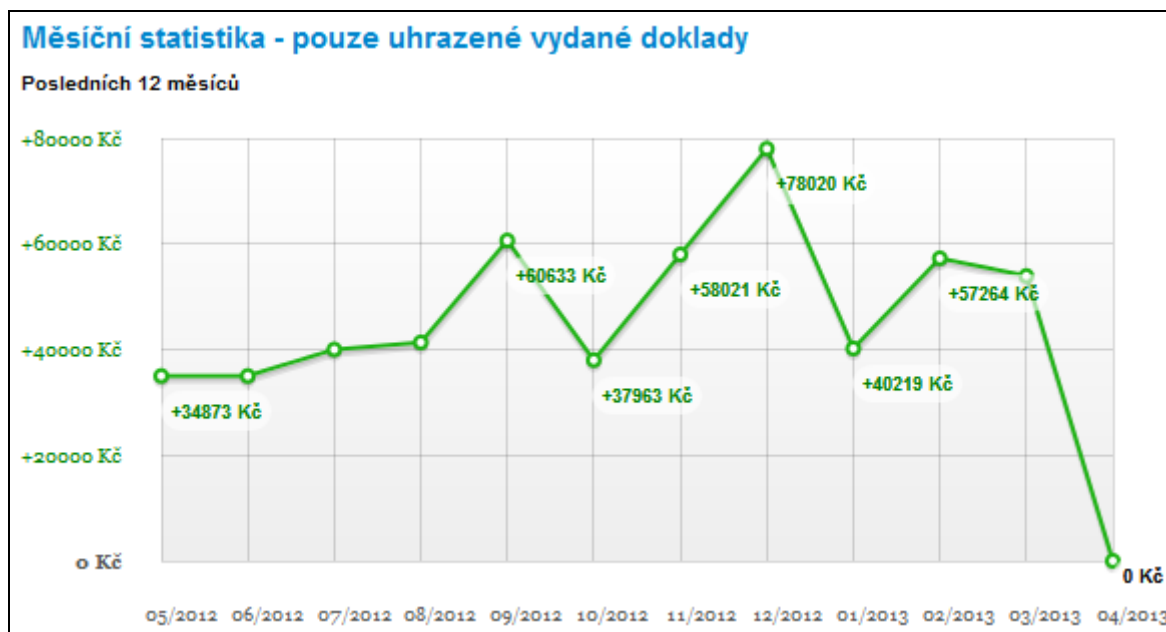
### 6.1.2 Výsledky testování

Po nasazení systému nemusí živnostník téměř vůbec kontrolovat periodickou fakturaci poskytovaných služeb, zákazníkům jsou faktury za služby zasílány automaticky, v případě přijetí platby za fakturu jsou zákazníci automaticky informováni o úhradě a prodloužení platnosti služby, popř. si mohou stav jejich služeb sami ověřit přes webové prostředí, což před nasazením systému nebylo vůbec možné a bylo nutné kontaktovat podnikatele telefonicky nebo emailem. Jeho podnikatelská činnost tak nyní působí na zákazníky mnohem profesionálněji a zákazníci jsou mnohem spokojenější, jelikož mají o službách větší přehled.

Po přechodu na webový ekonomický informační systém nedochází k žádným chybám způsobených lidským faktorem. Veškeré bankovní transakce živnostník eviduje přímo v systému, vydané faktury jsou automaticky označovány jako uhrazené a služby jsou tím automaticky prodlužovány o další fakturační období, odpadlo tak každodenní přihlašování do internetového bankovníctví a manuální prodlužování platnosti služeb zákazníků.

Bez informačního systému strávil živnostník minimálně 2 hodiny denně ekonomickou činností potřebnou k jeho podnikatelským aktivitám, nyní mu veškerá činnost spojená s ekonomikou jeho podnikání zabere maximálně 1 hodinu měsíčně. Ve výsledku tedy živnostník ušetřil přes 40 pracovních hodin za měsíc.

Živnostník získal rovněž kompletní přehled o finančním toku a jeho průběžném obratu, který dosud zjišťoval pouze jednou ročně. Díky systému tak může snadno sledovat jeho hospodaření. Nejpodstatnější je pro něj v současné chvíli grafická měsíční statistika uhrazených vydaných dokladů, která monitoruje jeho podnikatelskou činnost. Ukázka statistiky tohoto živnostníka je na následujícím obrázku (Obr. 14).



Obr. 14. Ukázka grafické statistiky uhrazených vydaných dokladů

## 6.2 Obchodní společnost s ručením omezeným – plátce DPH

### 6.2.1 Představení subjektu

U obchodní společnosti s ručením omezeným (plátce DPH), na které byl informační systém podrobně testován, je situace oproti živnostníkovi podstatně komplikovanější. Dynamicky se vyvíjející společnost se zabývá primárně prodejem zboží a jako hlavní prodejní kanál využívá převážně internetový obchod, po té až telefon a email. Díky kamenné prodejně prodává firma pouze cca 5% zboží, k distribuci prodejního artiklu tedy využívá převážně přepravní společnosti a poštu. Firma zaměstnává několik zaměstnanců a v pracovním procesu firmy jsou zařazeni i samotní majitelé společnosti.

Společnost distribuuje zboží zejména v České republice a na Slovensku, z čehož plyne jednoznačná potřeba více měn, minimálně české koruny a eura. Dodavatelé společnosti nesídlí pouze v ČR, nýbrž je firmě dodáváno zboží téměř z celého světa, od Evropy přes Spojené státy americké až po Čínu. Pro firmu je tak velmi podstatná možnost využití i dalších měn mimo eura, zejména amerického dolaru a britské libry.

Pro účely účetnictví firma používá devizové kurzy z ČNB platné vždy k poslednímu dni v měsíci pro celý následující měsíc. Společnost využívá k podnikání více bankovních účtů

u několika bankovních institucí, konkrétně dva bankovní účty vedené v české měně a jeden bankovní účet vedený v eurech.

Firma nefakturuje žádné služby periodicky s nějakou pravidelností, hlavní činností je prodej zboží koncovým zákazníkům a velkoobchodní prodej zboží dalším společnostem. Platebními metodami za jejich pohledávky jsou zejména bankovní převod a tzv. dobírka, pomocí které jsou ale nakonec finance od přepravních společností opět zasílány bankovním převodem. Hotovostní platby jsou aplikovány pouze v kamenné prodejně, takže jich je minimální množství.


















Společnost vystavuje průměrně 500 dokladů měsíčně, v předvánočním období se jedná až o 1000 dokladů měsíčně. Přijatých dokladů je v průměru okolo 100 měsíčně, z nich vznikající závazky jsou hrazeny převážně bankovním převodem.

#### **6.2.2 Výsledky testování**

Po přechodu společnosti na webový ekonomický informační systém byly téměř veškeré ekonomické procesy automatizovány. Devizové kurzy měn jsou automaticky aktualizovány z ČNB, veškeré příchozí bankovní transakce jsou automaticky ukládány do systému a vydané doklady jsou na základě těchto transakcí označovány jako uhrazené.

Společnost v systému eviduje i přijaté doklady v elektronické podobě, úhrady pohledávek realizuje díky ABO formátu ve formě dávkových příkazů. Systém je zároveň využíván jako centrální úložiště veškerých dokladů.

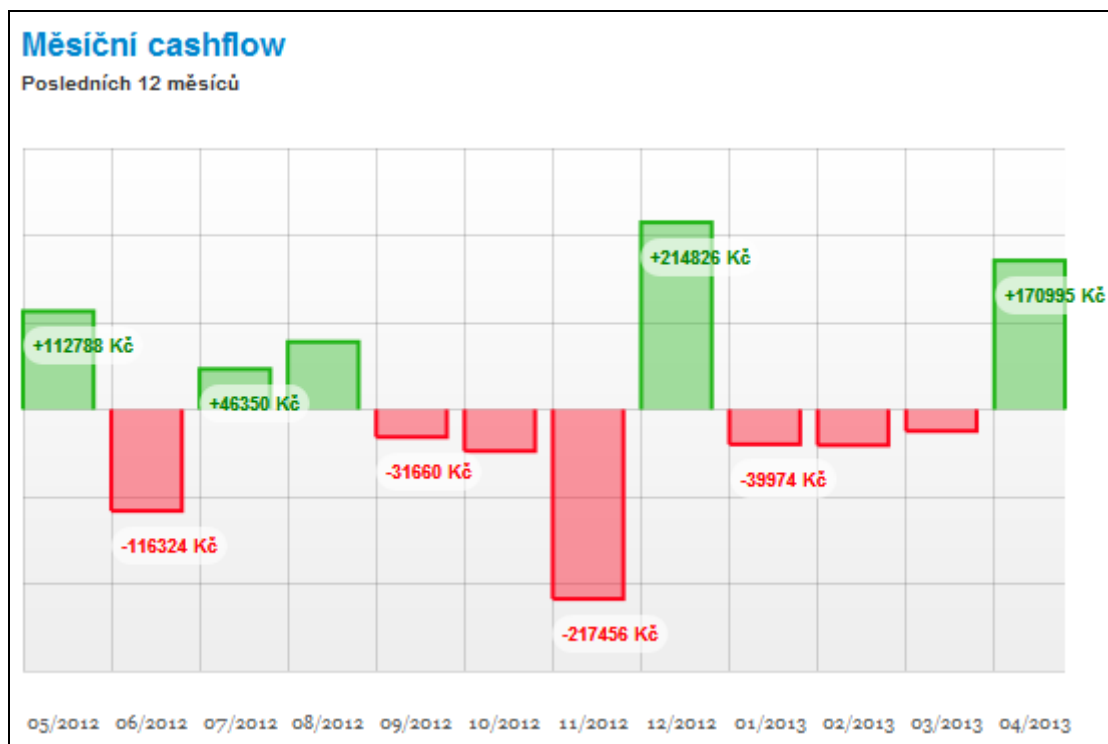
Na obrázku (Obr. 15) je ukázka výpisu příchozích bankovních transakcí a jejich automatického přiřazení k odpovídajícím vydaným dokladům. Z ukázkového výpisu je rovněž patrné přehledné sloučení příchozích transakcí z více bankovních účtů v různých měnách do společného výpisu.

◆ Datum	◆ Číslo účtu odesílatele	◆ Číslo účtu	◆ Částka	◆ VS	◆ Příjem
26.03.2013	157546896/0300	2600175217/2010	+ 883,00 Kč	20130929	 <a href="#">Faktura 20130929</a>
26.03.2013	1148390193/0800	2600175217/2010	+ 267,00 Kč	5245	 <a href="#">Faktura 20131102</a>
26.03.2013	25542463/7500	2400210313/8330	+ 40,51 EUR	20130940	 <a href="#">Faktura 20130940</a>
26.03.2013	25542463/7500	2400210313/8330	+ 116,00 EUR	20130943	 <a href="#">Faktura 20130943</a>
26.03.2013	25542463/7500	2400210313/8330	+ 85,08 EUR	20130944	 <a href="#">Faktura 20130944</a>
26.03.2013	25542463/7500	2400210313/8330	+ 64,71 EUR	20130953	 <a href="#">Faktura 20130953</a>
26.03.2013	366521193/0800	43-6868730267/0100	+ 1.170,00 Kč	5302	 <a href="#">Faktura 20131117</a>
26.03.2013	207520193/0600	43-6868730267/0100	+ 2.784,00 Kč	5253	 <a href="#">Faktura 20131118</a>
25.03.2013	78-2460080207/0100	43-6868730267/0100	+ 998,00 Kč	5273	 <a href="#">Faktura 20131074</a>
25.03.2013	3862654001/5500	2600175217/2010	+ 467,00 Kč	5269	 <a href="#">Faktura 20131089</a>
25.03.2013	157546896/0300	2600175217/2010	+ 1.078,00 Kč	20130901	 <a href="#">Faktura 20130901</a>
25.03.2013	157546896/0300	2600175217/2010	+ 3.048,00 Kč	20130951	 <a href="#">Faktura 20130951</a>
25.03.2013	157546896/0300	2600175217/2010	+ 775,00 Kč	20130962	 <a href="#">Faktura 20130962</a>
25.03.2013	157546896/0300	2600175217/2010	+ 3.540,00 Kč	20130963	 <a href="#">Faktura 20130963</a>
25.03.2013	157546896/0300	2600175217/2010	+ 517,00 Kč	20130975	 <a href="#">Faktura 20130975</a>
25.03.2013	157546896/0300	2600175217/2010	+ 1.855,00 Kč	20130979	 <a href="#">Faktura 20130979</a>
25.03.2013	157546896/0300	2600175217/2010	+ 883,00 Kč	20130982	 <a href="#">Faktura 20130982</a>

Obr. 15. Ukázka výpisu příchozích bankovních transakcí

Činnost spojená s běžným ekonomickým provozem společnosti zaměstnávala před nasazením systému dva zaměstnance. Po přechodu na nový systém a zaškolení je k této činnosti zapotřebí pouze jednoho zaměstnance. Firma se tak může mnohem více věnovat samotnému předmětu podnikání a jeho rozvoji.

Vedení společnosti navíc získalo podrobné statistiky finančního toku. Ukázka grafické statistiky finančního toku je ke zhlédnutí na následujícím obrázku (Obr. 16).



Obr. 16. Ukázka grafické statistiky finančního toku na základě uhrazených dokladů

## ZÁVĚR

Cílem této práce bylo seznámení se s technologiemi, které byly následně využity k návrhu a implementaci webového ekonomického informačního systému. Byly analyzovány potřeby ekonomických subjektů, zejména evidence přijatých a vydaných dokladů, pohledávek, řízení toku peněz a propojení s bankovními účty. Návrh systému byl mimo jiné zaměřen na automatizaci některých procesů v běžném provozu ekonomických subjektů a kladl důraz zejména na zabezpečení víceuživatelského webového systému.

Výsledkem práce je realizace moderního webového ekonomického informačního systému, který umožňuje přístup do systému nejenom samotným administrátorům, ale i zákazníkům a dodavatelům ekonomických subjektů. Informační systém splňuje veškeré definované funkční i nefunkční požadavky. Systém automatizuje fakturaci periodických služeb a umožňuje automatickou evidenci příchozích bankovních transakcí, na základě čehož označuje doklady jako uhrazené, popř. automaticky prodlužuje platnost fakturovaných služeb. Realizovaný systém poskytuje podrobné finanční statistiky. Na základě návrhu systému byla do implementace zahrnuta i rozšíření dle potřeb konkrétních ekonomických subjektů. Aplikace byla zabezpečena proti útokům.

Webový ekonomický informační systém byl úspěšně otestován v reálném provozu dvou ekonomických subjektů a po ukončení testování zůstal ekonomický systém na vlastní žádost subjektů nedílnou součástí jejich provozu, což jednoznačně potvrzuje pozitivní přínosy systému. Implementovaný ekonomický informační systém bude v budoucnu rozšířen o skladovou evidenci.

## ZÁVĚR V ANGLIČTINĚ

The aim of this thesis was an introduction into the technologies that were later used to design and implement a web-based economic information system. It has been analyzed the needs of companies, especially records of issued and received invoices, accounts receivable, the control of cash flow and connection with bank accounts. System design was among other things focused on the automation of some processes in the usual operations of companies and emphasizing security of multi-user web-based system.

The result of work is the realization of a modern web-based economic information system that provides access to system for administrators as well as customers and suppliers of companies. Information system complies with all defined functional and non-functional requirements. The system automates the periodic billing services and provides automatic recording of incoming banking transactions. The realized system provides detailed financial statistics. Application has been secured against attacks.

Web-based economic information system has been successfully tested in real operations of two companies and after that the economic systems have remained at their own requests as an integral part of their operations which unambiguously confirms the positive benefits of the system. Economic information system may be extended of the stock evidence in the future.

## SEZNAM POUŽITÉ LITERATURY

- [1] DOSTÁLEK, Libor. *Velký průvodce protokoly TCP/IP a systémem DNS*. 2. aktualiz. vyd. Praha: Computer Press, 2000, 426 s. ISBN 80-722-6323-4.
- [2] PUŽMANOVÁ, Rita. *TCP/IP v kostce*. 2. upr. a rozš. vyd. České Budějovice: Kopp, 2009, 619 s. ISBN 978-80-7232-388-3.
- [3] SHIFLETT, Chris. *HTTP developer's handbook*. Indianapolis, Ind.: Sams, 2003, 282 s. ISBN 06-723-2454-7.
- [4] SCHAFER, Steven M. *HTML, XHTML a CSS*. Praha: Grada, 2009, 647 s. ISBN 978-80-247-2850-6.
- [5] PROCHÁZKA, David. *CSS a XHTML: tvorba dokonalých WWW stránek krok za krokem*. 2., aktualiz. vyd. Praha: Grada, 2011, 175 s. Průvodce (Grada). ISBN 978-80-247-3897-0.
- [6] YOUNG, Michael J. *XML krok za krokem*. Praha: Mobil Media a.s., 2002, 471 s. ISBN 80-865-9328-2.
- [7] FLANAGAN, David. *Javascript - Kompletní průvodce*. Praha: Computer Press, 1998. ISBN 80-7226-093-6.
- [8] MARGORÍN, Marián. *JQuery bez předchozích znalostí*. Brno: Computer Press, 2011, 253 s. ISBN 978-80-251-3379-8.
- [9] LAL, Rajesh a Lakshmi CHAVA. *Developing Web Widget with HTML, CSS, JSON and Ajax: a complete guide to Web Widget*. Charleston, SC: CreateSpace, 2010. ISBN 14-505-0228-8.
- [10] BRÁZA, Jiří. *PHP 5: začínáme programovat*. Praha: Grada Publishing, 2005, 244 s. ISBN 80-247-1146-X.
- [11] VOSTROVSKÝ, Václav. *Relační databázové systémy*. Praha: ČZU PEF Praha ve vydavatelství Credit, 2001, 95 s. ISBN 80-213-0753-6.
- [12] BELL, Charles A. *Expert MySQL*. New York: Apress, 2007, 577 s. ISBN 978-159-0597-415.
- [13] GROFF, James R. a Paul N. WEINBERG. *SQL: kompletní průvodce*. Brno: CP Books, 2005, 936 s. ISBN 80-251-0369-2.



- [14] November 2012 Web Server Survey. *Netcraft* [online]. 2012 [cit. 2013-02-05]. Dostupné z: <http://news.netcraft.com/archives/2012/11/01/november-2012-web-server-survey.html>
- [15] POŠMURA, Vladimír. *Apache: Příručka správce WWW serveru*. Praha: Computer Press, 2002, 311 s. ISBN 80-722-6696-9.
- [16] GALBRAITH, Patrick. *Developing Web Applications with Apache, MySQL, memcached, and Perl*. Hoboken: John Wiley, 2009. ISBN 978-047-0538-326.
- [17] REINHEIMER, Paul. *Professional Web APIs with PHP*. Indianapolis, Ind.: John Wiley, 2006, 356 s. ISBN 978-076-4589-546.
- [18] HAYDER, Hasin a Lucian GHEORGHE. *Smarty PHP template programming and applications*. Birmingham, U.K: Packt Publishing, 2006. ISBN 18-471-9028-6.
- [19] PADOVA, Ted. *Adobe Acrobat 8 PDF Bible*. Hoboken: John Wiley, 2008. ISBN 04-703-7765-8.
- [20] ŠTOHL, Pavel. *Učebnice Účetnictví 2012: pro střední školy a pro veřejnost*. 13., upr. vyd. Znojmo: Pavel Štohl, 2012. ISBN 978-80-87237-31-1.
- [21] RYLOVÁ, Zuzana. *Daňové zákony 2012*. Brno: Computer Press, 2012, 254 s. ISBN 978-802-5137-949.
- [22] KOVALÍKOVÁ, Hana. *Zákon o účetnictví*. Olomouc: ANAG, 2010. ISBN 978-80-7263-730-0.
- [23] LEDVINKOVÁ, Jana. *Daň z přidané hodnoty*. Praha: VOX, 2013. ISBN 978-80-87480-14-4.
- [24] ČERNOHORSKÝ, Jan a Petr TEPLÝ. *Základy financí*. Praha: Grada, 2011, 304 s. ISBN 978-80-247-3669-3.
- [25] POLOUČEK, Stanislav. *Bankovnictví*. Praha: C. H. Beck, 2006, 716 s. ISBN 80-717-9462-7.
- [26] STORMWARE S.R.O. *POHODA Ekonomický systém* [online]. 2012 [cit. 2013-03-23]. Dostupné z: <http://www.stormware.cz/>

- [27] CÍGLER SOFTWARE A.S. *IDoklad: služby pro online faktury zdarma* [online]. 2012 [cit. 2013-03-23]. Dostupné z: <https://www.idoklad.cz/>
- [28] FLEXIBEE SYSTEMS S.R.O. *FlexiBee: internetové ekonomické systémy* [online]. 2013 [cit. 2013-04-21]. Dostupné z: <http://www.flexibee.eu/>
- [29] KONAROVSKÝ, Lukáš a Jan KORBEL. *Fakturoid* [online]. 2013 [cit. 2013-04-21]. Dostupné z: <http://www.fakturoid.cz/>
- [30] DANIEL, Jan. *Qfaktury: Online vytváření a správa faktur* [online]. 2011 [cit. 2013-04-21]. Dostupné z: <http://www.qfaktury.cz/>
- [31] DIRECT ACCOUNTING S.R.O. *IÚčto* [online]. 2012 [cit. 2013-04-21]. Dostupné z: <http://www.iucto.cz/>

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

AJAX	Asynchronous JavaScript and XML.
API	Application Programming Interface.
CA	Certifikační autorita.
CSS	Cascading Style Sheets.
ČNB	Česká národní banka.
DIČ	Daňové identifikační číslo.
DPH	Daň z přidané hodnoty.
DTD	Definice typu dokumentu.
EU	Evropská unie.
GPL	General Public License.
GUI	Graphical User Interface.
HTML	HyperText Markup Language.
HTTP	Hypertext Transfer Protocol.
HTTPS	Hypertext Transfer Protocol Secure.
IP	Internet Protocol.
IS	Informační systém.
JSON	JavaScript Object Notation.
MIT	Massachusetts Institute of Technology.
MVC	Model-View-Controller.
ORM	Objektově relační mapování.
PDF	Portable Document Format.
RFC	Request For Comments.
SGML	Standard Generalized Markup Language.
SQL	Structured Query Language.

SSL	Secure Sockets Layer.
TCP/IP	Transmission Control Protocol / Internet Protocol.
URL	Uniform Resource Locator.
VIES	VAT Information Exchange System.
WWW	World Wide Web.
XHR	XMLHttpRequest.
XML	Extensible Markup Language.

**SEZNAM OBRÁZKŮ**

Obr. 1. Některé protokoly z rodiny protokolů TCP/IP [1] .....	13
Obr. 2. Architektura MySQL databázového serveru [12] .....	22
Obr. 3. Ukázka výstupu ze systému VIES po ověření platného DIČ .....	34
Obr. 4. Zjednodušený model případů užití .....	51
Obr. 5. Návrh uživatelského rozhraní – přihlášený uživatel .....	52
Obr. 6. Návrh uživatelského rozhraní – nepřihlášený uživatel .....	53
Obr. 7. Ovládací prvek – výběr data .....	54
Obr. 8. Šifrované spojení v prohlížečích .....	56
Obr. 9. Schéma přístupu ke korektní databázi ekonomického subjektu .....	58
Obr. 10. Implementace GUI – nepřihlášený uživatel .....	64
Obr. 11. Implementace GUI – přihlášený uživatel (administrátor) .....	64
Obr. 12. Implementace GUI – přihlášený uživatel (zákazník) .....	65
Obr. 13. Ukázka implementace vydaných dokladů v PDF formátu .....	66
Obr. 14. Ukázka grafické statistiky uhrazených vydaných dokladů .....	74
Obr. 15. Ukázka výpisu příchozích bankovních transakcí .....	76
Obr. 16. Ukázka grafické statistiky finančního toku na základě uhrazených dokladů .....	77

**SEZNAM TABULEK**

Tab. 1. Hlavní skupiny stavových kódů HTTP odpovědí.....	15
Tab. 2. Význam polí v syntaxi URL .....	16
Tab. 3. Zastoupení hlavních webových serverů [14] .....	23
Tab. 4. Význam pozic v syntaxi řádku definice Cron.....	24
Tab. 5. Nejdůležitější implementované objektové třídy .....	61
Tab. 6. Nejvýznamnější MySQL databázové tabulky.....	62
Tab. 7. Struktura systémové databáze - tabulka „system“ .....	69

## SEZNAM PŘÍLOH

P I      Obsah CD

## PŘÍLOHA P I: OBSAH CD

Adresářová struktura přiloženého CD je popsána v následující tabulce:

Cesta	Popis
/DP/pdf/	Tato diplomová práce ve formátu PDF.
/DP/source/	Tato diplomová práce ve formátu Microsoft Word.
/IS/source/	Zdrojové kódy informačního systému.
/IS/sql/master.sql	SQL skript pro vytvoření hlavní systémové databáze.
/IS/sql/system.sql	SQL skript pro vytvoření databáze ekonomického subjektu.