

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství



Bakalářská práce

Programovací jazyky pro tvorbu webu

Jan Hurt

©2015 ČZU v Praze

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Katedra informačního inženýrství

Provozně ekonomická fakulta

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Jan Hurt

Informatika

Název práce

Programovací jazyky pro tvorbu webu

Název anglicky

Programming languages for web development

Cíle práce

Bakalářská práce je zaměřena na oblast tvorby webových prezentací s využitím moderních programovacích jazyků k tomu určených. Hlavním cílem je porovnat existující programovací jazyky a na ukázkové aplikaci demonstrovat vlastnosti jednoho z těchto jazyků.

Metodika

Metodika bakalářské práce je založena na studiu a analýze odborných informačních zdrojů. Na základě syntézy zjištěných teoretických poznatků budou shrnuty klady a zápory jednotlivých jazyků a vybrán jeden jazyk, který bude využit pro tvorbu ukázkové aplikace, která bude demonstrovat jeho vlastnosti a možnosti využití.

Doporučený rozsah práce

35-40 stran

Doporučené zdroje informací

978-80-247-2850-6, 17.07.2009

978-80-247-3060-8, 24.05.2010

E., Holzschlag Molly – HTML a CSS – jdi do toho, ISBN:80-247-1454-X, 17.03.2006

Hal Edwin Fulton-Ruby kompendium znalostí pro začátečníky i profesionály, ISBN:

Kosek Jiří-PHP a XML, ISBN 8024711168

Leiss Oliver, Schmidt Jasmin-PHP v praxi,ISBN:

Mark Lutz David Ascher-Naučte se Python – pohotová příručka, ISBN: 80-247-0367-X, 2003

Slavoj Písek: HTML- Tvorba jednoduchých internetových stránek, ISBN: 978-80-2476-094-0, Grada 2006

Steven M. Schafer-HTML, XHTML a CSS Steven M. Schafer,ISBN:

Ullman, Larry-PHP a MySQL: názorný průvodce tvorbou dynamických www stránek, ISBN: 8025100634
8074130185, 9788074130182, 2009

Předběžný termín obhajoby

2015/06 (červen)

Vedoucí práce

Ing. Jiří Brožek, Ph.D.

Elektronicky schváleno dne 10. 11. 2014

Ing. Martin Pelikán, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 10. 11. 2014

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 11. 03. 2015

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Programovací jazyky pro tvorbu webu" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 16.3.2015

Poděkování

Rád bych touto cestou poděkoval vedoucímu této bakalářské práce Ing. Jiřímu Brožkovi, Ph.D. za přínosné rady, konzultaci a vedení bakalářské práce, pomoc s teoretickou i praktickou částí této práce.

Programovací jazyky pro tvorbu webu

Programming languages for web development

Souhrn

Cílem bakalářské práce je představení problematiky tvorby webových prezentací s využitím specifických moderních programovacích jazyků. Zaměřuje se na porovnání různých vybraných existujících jazyků, historii jejich vzniku, výhody a nevýhody a ukázkou části kódu.

Teoretická část práce se zaměřuje na téma webových stránek a problematiku od začátků internetu, přes službu World Wide Web až po základní stavební kámen webu, jazyk HTML. Dále stručně představuje jednotlivé skriptovací technologie pro tvorbu webových aplikací od jejich zrodu po současný stav. Mimo jiné bude teoretická část práce obsahovat ukázkou tvorby, tj. psaní, kódu. V následující kapitole je hlavním pilířem této práce výběr jedné z technologií, v tomto případě skriptovacího programovacího jazyka PHP (*Hypertext Preprocessor*). V neposlední řadě práce detailně popisuje fungování tohoto jazyka.

V praktické části je pro demonstraci vytvořena ukázkou programovacího jazyka PHP, v našem případě se jedná o dynamickou webovou stránku. Práce obsahuje také představení, popis a využití sekundárního softwaru, bez kterého by demonstrace nemohla být vytvořena.

Summary

The aim of this thesis is to introduce the issue of creating websites using specific modern programming languages. It focuses on the comparison of various selected existing languages, the history of their origin, advantages and disadvantages, syntax, sample code.

The theoretical part of work focuses on the topic of the website and the issues from the beginnings of the Internet via the World Wide Web to the basic building block of the Web, Hypertext Markup Language. It also briefly presents the different scripting

technologies for building Web applications from their birth to the current state. Among other things, the theoretical part of the paper includes sample code. The following chapter is the main pillar of this work and selects one of the technologies, in this case, a scripting programming language PHP (*Hypertext Preprocessor*). Finally, it describes in detail this functioning of the language.

In the practical part, a sample of PHP programming language is created for demonstration, in this example, a dynamic website. The work also includes performances, description and use of secondary software, without which the demonstrations could not be created.

Klíčová slova: Programovací jazyk, www, PHP, ASP.NET, Ruby, Python, HTML

Keywords: Programming language, www, PHP, ASP.NET, Ruby, Python, HTML

Obsah

1.	Úvod.....	7
2.	Cíl práce a metodika	8
2.1.	Cíl práce.....	8
2.2.	Metodika.....	8
3.	Přehled řešené problematiky	9
3.1.	WWW.....	9
3.1.1.	Uniform Ressource Locator	9
3.1.2.	HyperText Markup Language	10
3.1.3.	HyperText Transfer Protocol.....	10
3.1.4.	WWW server a WWW klient.....	11
3.1.4.1.	WWW server	11
3.1.4.2.	WWW klient	11
3.1.5.	Grafika pro WWW	12
3.1.5.1.	Statická grafika pro World Wide Web.....	12
3.1.5.2.	Dynamická grafika pro World Wide Web.....	13
3.1.6.	Dílčí shrnutí.....	13
3.2.	HTML.....	13
3.2.1.	Historie	15
3.2.1.1.	Verze	15
3.2.2.	Editor pro grafiku HTML.....	16
3.3.	Programovací skriptovací jazyk	16
3.3.1.	Charakteristika programovacího skriptovacího jazyka	17
3.3.1.1.	Index TIOBE.....	17
3.3.2.	ASP.NET	17
3.3.2.1.	Historie.....	18
3.3.2.2.	Výhody.....	18
3.3.2.3.	Nevýhody.....	19
3.3.3.	Ruby	19
3.3.3.1.	Historie.....	19
3.3.3.2.	Výhody.....	20

3.3.3.3.	Nevýhody.....	20
3.3.4.	Perl.....	20
3.3.4.1.	Historie.....	21
3.3.4.2.	Výhody.....	21
3.3.4.3.	Nevýhody.....	21
3.3.5.	Python.....	22
3.3.5.1.	Historie.....	22
3.3.5.2.	Výhody.....	23
3.3.5.3.	Nevýhody.....	23
3.3.6.	Java.....	23
3.3.6.1.	Historie.....	23
3.3.6.2.	Výhody.....	24
3.3.6.3.	Nevýhody.....	24
3.3.6.4.	Verze.....	24
4.	PHP.....	26
4.1.	Historie.....	27
4.2.	Výhody.....	28
4.3.	Nevýhody.....	29
5.	Vlastní webová dokumentace.....	30
5.1.	Charakteristika.....	30
5.2.	Struktura webu.....	30
5.2.1.	Layout.....	31
5.2.2.	Obsah vytvořeného příkladu.....	32
5.2.3.	HTML struktura.....	32
5.3.	Hlavní prvek PHP.....	36
5.4.	Sekundární software.....	39
5.4.1.	MySQL.....	39
5.4.2.	phpMySQL.....	39
5.4.3.	Apache.....	41
5.4.4.	Editor.....	41
5.5.	SEO.....	43
6.	Zhodnocení výsledku.....	44

7. Závěr	46
8. Zdroje.....	47
8.1. Tištěné publikované dokumenty.....	47
8.2. Elektronické zdroje.....	48

Seznam obrázků

Obrázek 1: Vazby mezi WWW klientem a WWW serverem	12
Obrázek 2: Ukázka architektury jazyka HTML	14
Obrázek 3: Ukázka hlavních položek webu	31
Obrázek 4: Grafický layout webových stránek.....	32
Obrázek 5: Grafické znázornění dynamického prvku	37
Obrázek 6: Znázornění prostředí phpMyAdmin.....	41
Obrázek 7: Ukázka editoru Sublime Text 3.....	42
Obrázek 8: Celkový vzhled webových dynamických stránek	42
Obrázek 9: SEO optimalizace.....	43

1. Úvod

Člověk je tvor velmi vynalézavý a postupem času zdokonaluje nejen sám sebe, ale i věci, které jemu a jeho okolí usnadňují život. Tento fakt nám potvrdil vývoj lidstva a techniky. Slovo technika je z řeckého „techné“, což znamená řemeslo. Již v antické době docházelo v různých řemeslech nejen k modernizaci, ale i vývoji různých forem provázanosti lidstva. Lidstvo se posunulo kupředu a vyvíjely se různé mechanismy, které se stále zdokonalovaly jak svou složitostí, tak použitou technologií.

Jak se lidská společnost vyvíjela, začala se rozšiřovat do různých koutů světa a vzdálenost mezi jednotlivými lidmi se zvětšovala. Nejdříve stačila pošta či dostavníky, postupem času telefon a nakonec byla vzdálenost tak neúnosná a požadavky natolik vysoké, že se lidé vybaveni moderními výpočetními technologiemi snažili propojit svět sítěmi, které fungují na bázi uživatelských uzlů. Tak vznikla první počítačová celosvětová síť ARPANET v roce 1969. Během následujících let se přidávaly další uzly, až se z původního jednoduchého projektu stala komplikovaná globální „síť sítí“, která se dnes nazývá Internet. Tím vznikl další prostor pro rozvoj lidské vynalézavosti a s tím spojené služby jako WWW či Email. Masivní rozšíření těchto technologií a služeb vyneslo navrchol webové aplikace, které spojily uživatele po celém světě víc, než si kdo na počátku dokázal představit. Největší podíl na současné dominanci webových aplikací má značkovací jazyk HTML. Dnes se k těmto technologiím řadí pojmy ze světa programovacích skriptovacích jazyků, jako PHP, ASP.NET, Ruby, Python a mnohé další.

Důvodem volby tématu této bakalářské práce bylo široké pole působnosti a rozsah problematiky, který je do jisté míry nevyčerpatelný. Každým dnem přichází na svět nové technologie, nápady a různá vylepšení předešlých verzí. Ty posouvají nejen tento směr, ale celkové lidské poznání o kus dále, přičemž dochází ke změně významných paradigmat životních postojů. Tématika informatiky a informačních technologií je nejen zajímavá, ale s ohledem na dnešní dobu také velmi aktuální.

2. Cíl práce a metodika

2.1. Cíl práce

Bakalářská práce je zaměřena na oblast tvorby webových prezentací s využitím moderních programovacích jazyků k tomu určených. Hlavním cílem je porovnat existující programovací jazyky a na ukázkové aplikaci demonstrovat vlastnosti jednoho z těchto jazyků.

2.2. Metodika

První část práce je zaměřena na vysvětlení pojmů a základních charakteristik. Studuje a analyzuje informační zdroje. Na základě syntézy zjištěných teoretických poznatků budou shrnuty klady a zápory jednotlivých jazyků, jejich popis a porovnání. Informace potřebné pro první část práce byly získány s pomocí sekundárních zdrojů odborné literatury.

Jednotlivé technologie jsou v práci prezentovány v následující struktuře: na úvod je programovací jazyk představen, poté následuje vizualizace kódu jazyka, u kterých by čtenář měl vidět jak rozdílnost, tak zároveň podobné či shodné znaky. Následně Index TIOBE reprezentující uživatelské preference. Prezentace jazyka dále obsahuje historii, přehled výhod i nevýhod dané technologie. Ke konci teoretické práce je věnována větší pozornost jazyku PHP, který je vybranou technologií pro praktickou ukázkou.

V druhé části práce je zvolen jeden z jazyků, který bude využit pro tvorbu ukázkové aplikace. Ta bude demonstrovat jeho vlastnosti a možnosti využití. Na závěr dojde k prezentaci sekundárního softwaru, který je pro vytvoření demonstrativní ukázky nezbytný.

Vlastní závěr práce patří zhodnocení a bilancování jednotlivých údajů a poznatků.

Práce by měla sloužit jako zdroj informací pro čtenáře, kterým by pomohla v orientaci v dané tématice a přiblížila jim daný problém.

3. Přehled řešené problematiky

Obsahem této části práce je definovat WWW, HTML, charakterizovat programovací skriptovací jazyk, index TIOBE a popsat jednotlivé typy jazyků.

3.1. WWW

Podle Perryho (1996, str. 18) reprezentuje World Wide Web, W3 nebo jen samostatně Web službu internetu založenou na Hypertextu. Často dochází ke ztotožnění WWW se samotným internetem, což je základní chyba, neboť WWW je jeho pouhým nástrojem.

Hlavenka (2002, str. 8) blíže popisuje World Wide Web, jehož zrod se váže s rokem 1989. Zakladatelem této služby byl Tim Berners-Lee. Tento Brit vytvořil WWW při svém působení v CERNu (Evropská organizace pro jaderný výzkum) a zasloužil se o další úspěchy na poli informatiky a informačních technologií. V roce 1994 založil Consorciium W3C, které se zabývá širokým spektrem otázek kolem populárního webu od jazyka HTML, protokolu HTTP až po řešení grafických otázek. Pro tuto práci je však nejdůležitější standardizace WWW.

Hlavenka (2002, str. 2) uvádí, že o WWW, se zprvu uvažovalo jen jako o službě pro zpřístupnění textu. Podpora grafiky se objevila až roku 1993. Prezentační schopnosti WWW však byly počátečním obdobím slabé. Celkovou ideou WWW bylo vytvořit pouhé dílčí služby. V současnosti je tento pohled změněn a WWW představuje globálního giganta. Podporuje celou řadu formátů. Dnes již má velmi dobré prezentační schopnost a dokonce tvoří platformu pro poskytování jiných služeb. Nabízí vyhledávání a účinně spolupracuje s vazbami mezi jednotlivými aplikacemi.

Základními technologiemi WWW jsou:

1. Uniform Resource Locator (URL)
2. HyperText Markup Language (HTML)
3. Hypertext Transfer Protocol (HTTP)

3.1.1. Uniform Resource Locator

Perry (1996, str. 28) nás blíže informuje o pojmu URL, zkratce vytvořené z již zmiňovaného názvu **Uniform Resource Locator**. URL je prostředek pro nalezení jiného

počítače na Webu. Standardní URL se skládá z těchto částí: přenosového formátu, názvu hostitele a cesty. Obecný tvar URL je: protokol, uzel a jméno. Protokol říká, jakým způsobem má být k objektu přistupováno. Uzel je symbolické, spíše standardní doménové jméno, na kterém se objekt nachází. Jméno představuje název objektu včetně přístupové cesty. URL je tedy řetězec znaků, který definuje strukturu. Specifikuje umístění informací na Internetu a ukazuje cestu k těmto informacím.

3.1.2. HyperText Markup Language

Písek (2001, str. 12) prezentuje HTML jako značkovací jazyk, který je určený pro tvorbu www dokumentů. Skládá se ze souboru značek, které se dělí na párové, nepárové a speciální.

Písek (2001, str. 17) definuje strukturu HTML dokumentu, která vypadá následovně: Značka <html> začíná a končí obsah celého dokumentu. Hlavička (header) <head> obsahuje a definuje technické údaje, jako je titulek <Title>, kódování, metadata, která se specializují na: klíčová slova, informace o autorovi a typu kódování, obsahuje i jiné značky, např. link apod. Tělo (body) <body> představuje tělo dokumentu. Slouží k zobrazování veškerého obsahu stránky. Všechny uvedené značky, tzv. tagy, jsou párové.

3.1.3. Hypertext Transfer Protocol

Ullman (2004, str. 121) označuje HTTP jako výměnný protokol, tzv. srdce WWW. Jedná se o vazbu mezi klienty a servery. Kdykoli klient požádá prohlížeč o službu, obdrží odpověď. Využívá transportní služby protokolu TCP, tj. nejpoužívanějšího internetového transportního protokolu.

Metody HTTP:

GET podle autora Procházky (2012, str. 91) představuje požadavek klienta na poskytnutí WWW stránek. Umožňuje i předávat parametry a hodnoty do PHP skriptů.

POST posílá data na webový server, jak uvádí Procházka (2012, str. 91). Používá se při práci s formuláři pro zasílání odpovědí, které mají být dále zpracovány. Má efektivní provázání s metodou GET.

HEAD je požadavek na zaslání hlavičky WWW stránek.

3.1.4. WWW server a WWW klient

World Wide Web vychází z architektury Klient/Server, jenž předpokládá dělbu práce:

- Server (WWW server)
- Klient (WWW prohlížeč, browser)

3.1.4.1. WWW server

Hlavenka (2002, str. 7) nám přibližuje termín WWW serveru a následovně: „Počítač umístěný v síti Internet, který obsahuje data, z nichž jsou sestaveny webové stránky, a který je na žádost poskytuje uživatelům (klientům) webu“.

Každá odpověď má vlastní označení:

1xx – informativní odpověď, záleží na dané aplikaci.

2xx – 200 kladná odpověď (ok), 201 Created (vytvořeno), 202 Accepted (akceptováno).

3xx – očekává další aktivitu ze strany klienta

4xx – chyba na straně klienta, hlásí nějaký problém – 400 Bad request (chybný požadavek), 401 Unauthorized (neautorizovaný přístup), 403 Forbidden (přístup zamítnut), 404 Not found (objekt nenalezen).

5xx – chyba na straně serveru, hlásí problém – 501 Not implemented (neimplementováno), 503 Service unavailable (služba nedostupná).

3.1.4.2. WWW klient

Jedná se o software nebo program, který je instalovaný na lokální počítači nebo mobilním zařízení. Tento tzv. klient zasílá požadavek na vzdálený program, tím se myslí WWW server. Tento server je odpovědný za vyřízení požadavků klientů. Klient se poté rozhoduje o zobrazení objektů různých typů většinou podle svých možností. U některých formátů zvládne zpracování úplně sám. Pokud to sám nezvládne, volí si klient pomocný externí program, většinou helper nebo nesamostatný modul plug-in. Dnešní klienti dovedou pracovat v pozici klienta dalších služeb, kterou je např. FTP. Jedná se o protokol, jehož práce je přenášení souborů mezi počítači (Management Mania, 2013).

Obrázek 1: Vazby mezi WWW klientem a WWW serverem



Zdroj: <http://boss.ped.muni.cz/vyuka/material/puvodni/software.htm>

Technologie na straně klienta jsou rozdělené do tří skupin. První ze skupin je skupina statických technologií. Sem patří XHTML a CSS, které si představíme v kapitole 3.1.5.1. Další skupinu tvoří dynamické technologie, především DHTML. Jedná se o dynamické HTML, které zahrnuje různé technologie a zároveň popisuje, jak spolu tyto různé technologie fungují. V rámci dynamických technologií najdeme ještě Flash (viz kapitola 3.1.5.1.). Stranou od výše popsaných skupin se nachází HTML5 tvořící samostatnou větev.

Na straně serveru jsou známými technologiemi především PHP, ASP.NET a JSP. Do této oblasti serveru spadá i skupina, kterou lze pojmenovat jako skupinu databáze. V této skupině se nacházejí MySQL, MS SQL server a v neposlední řadě také PostgreSQL.

3.1.5. Grafika pro WWW

3.1.5.1. Statická grafika pro World Wide Web

Statická grafika v sobě zahrnuje podle Schafera (2009 str. 199) především formáty JPEG, GIF, PNG, CSS atd. Dále ji rozdělujeme podle jejího využití v grafice. Jako příklad lze uvést soubory pro fotky jpg, jpeg. Na jednoduchou grafiku pak spíše používáme formáty GIF a PNG. Statická grafika také využívá tzv. editory. Jedná se především o Adobe photoshop, GIMP a Zoner photoshop.

Schafer (2009, str. 199) rovněž poukazuje na výhody výše zmíněných formátů. Mezi výhody formátu JPEG patří především jeho vlastnost pro fotografie a formáty podobného charakteru. Další výhodou je výborný a plynulý přechod barev, nastavitelná kvalita a velikost obrázku. Oproti tomu je hlavní nevýhodou zkreslení grafické informace. Dále je používán formát GIF, jenž zvládá obrázky s převážně ostrými přechody barev, jakými jsou např. loga a kresby. Nevýhodou je především omezený počet barev, neboť GIF

dokáže rozlišit max. 265 barev. PNG je velmi univerzální formát, který navíc není kompromitován ztrátou kvality grafického objektu. Nevýhoda tohoto formátu spočívá v absenci tvorby animací.

3.1.5.2. Dynamická grafika pro World Wide Web

Schafer (2009, str. 198) nám přibližuje animovaný GIF, který se stal nepsaným standardem pro malá krátká videa bez zvuku. Ovládl místo na trhu koncem 90. let, obstarával veškerou animaci a svou jednoduchostí získal sympatie široké škály uživatelů. Jeho pokračovatelem byl Adobe Flash, který se zabývá interaktivními animacemi. Do této skupiny se dále řadí JavaScript, Java applet či formátování pomocí dokumentů v CSS viz kapitola 3.2.

3.1.6. Dílčí shrnutí

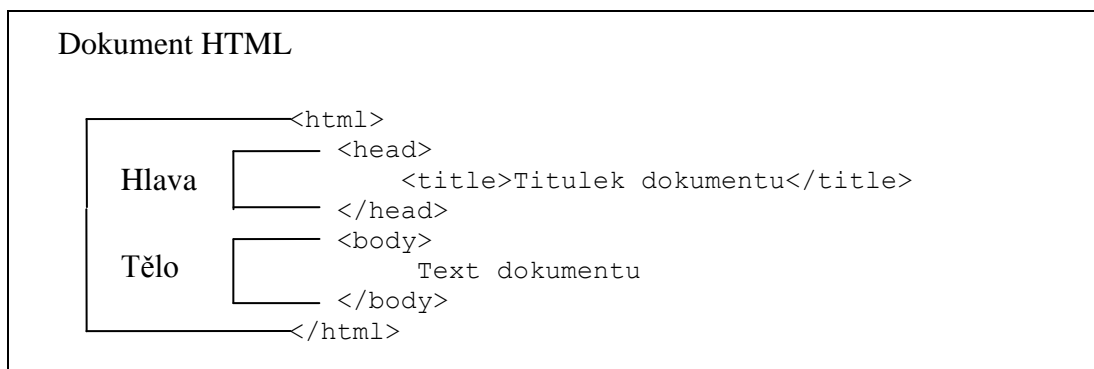
Co je hlavní příčinou úspěchu WWW služby a co způsobilo její masové rozšíření?

Základním předpokladem tohoto úspěchu byla skutečnost, že jí jako předloha posloužilo samotné lidské myšlení. Jedním z nejdůležitějších důvodů je však také její neskromnost. Dokázala zjednodušit internet a přiblížit ho široké veřejnosti. Zaměřila se na atraktivní funkce především pro komerční sféru, kam lze zařadit marketing, obchod a služby. Nelze opomenout ani fakt, že šlo o mimořádnou inovaci, které se podařilo najít volné místo na trhu a plně ho využít.

3.2. HTML

Písek (2001, str. 12) uvádí, že HTML je zkratkou **H**yper**T**ext **M**arkup **L**anguage. Jak již oficiální název napovídá, jedná se o jazyk, který využívá Hypertext. Funguje na principu značek neboli tagů, proto můžeme říct, že jde o značkovací jazyk. Představuje hlavní pilíř pro tvorbu World Wide Web stránek. Tyto stránky jsou většinou v podobě textu obohaceného o výše uvedené značky. Postupem času se HTML vyvíjelo a název jazyka byl upravován o dodatečná písmena. Jako například (D)HTML nebo (X)HTML.

Obrázek 2: Ukázka architektury jazyka HTML



Zdroj: Vlastní tvorba

Schafer (2009, st. 266) považuje DHTML za dynamické HTML. Lze o něm říci, že není „standardem“, což znamená, že nemá standardní předpis nebo normy. Zároveň je možné tvrdit, že se jedná o souhrn nástrojů a postupů umožňujících změny, v tomto ohledu dynamické změny v dokumentech HTML. Mezi jeho zajímavé vlastnosti patří např. animace pohybů prvků na stránce a provádění určité akce podle činnosti uživatele, či dokonce automatizace funkce prohlížeče. Skládá se z několika položek, především XHTML, novější verze již zastaralého HTML. Další položku tvoří CSS (Cascading Style Sheets). CSS představuje kolekci metod pro grafickou úpravu WWW stránek. V neposlední řadě se sem řadí JavaScript, což je objektivně orientovaný programovací jazyk pro WWW stránky. DHTML umožňuje řídit prezentaci dokumentu. Je samozřejmostí, že XHTML musí mít k dispozici skriptovací jazyk, který lze rozdělit do více skupin podle jejich účelu. Nejrozšířenější na tvorbu skriptů je pak právě JavaScript.

Schafer (2009, str. 337) přibližuje novější verzi HTML, a to XHTML. Představuje značkovací jazyk, který pomocí tagů neboli značek vytváří a formátuje dokumenty pro webové aplikace. Dokument se řídí podle pravidel DTD. Jde o nejpoužívanější formát dokumentů. Výhodou XHTML je bezesporu nemožnost formátování, tím pádem je definovaný pomocí CSS. Oproti tomu nevýhodou je náchylnost k chybám ve formátu a malá kvantita prohlížečů, které tuto technologii podporují.

3.2.1. Historie

Hlavenka (2002, str. 24) označuje za počátky HTML rok 1989. Tvůrcem HTML byl dvoučlenný tým, sestávající z autorů Tima Berners- Leea a Roberta Cailliaua. Mimo jiné je Lee otcem služby WWW a také prvního prohlížeče, který pojmenoval po službě WWW – World Wide Web. Tento projekt řadil mezi aktivity organizace CERN, výzkumného centra se nacházejícího se ve Švýcarsku a zabývajících se problematikou fyziky. Pro tvorbu dokumentů pro již zmíněné WWW bylo potřeba vymyslet nástroj, který by uměl formátovat text, vkládat do textu odkazy a umožňovat další práci s textem. První verze, ze které se postupem času stala základna pro dnešní gigantický systém a jež dala webu nový směr, se datuje do roku 1991.

3.2.1.1. Verze

Verze 1.2 vzniká mezi léty 1991 - 1993. Autory verze jsou Dani Connolly a druhý muž týmu Tim Berners- Lee. Tato verze nepodporuje grafické rozhraní. (KOSEK, 2013)

Verze 2.0 se váže k roku 1995. Jedná se o verzi, která již podporuje syntaxe a SGML (jazyk určený pro popis struktur dokumentů). (KOSEK, 2013)

O dva roky později přichází na svět verze s označením 3.2. Zajímavostí je, že se nejedná o verzi s označením 3.0, která byla původně navržena, neboť s ohledem na její složitost nedošlo k jejímu finálnímu schválení. Minulou verzi rozšiřuje o prvky ovlivňující vzhled. Dále byla přidána funkce na zarovnání textu. (KOSEK, 2013)

V roce 1997 nastoupila verze 4.0, která se nejvíce přibližuje k samotné myšlence prvků, vyznačujících význam. Přináší sebou další rozšíření, prvky pro tvorbu tabulek a standardizace rámců. (KOSEK, 2013)

Verze 4.01 opravila verzi předešlou a spolu s ní se přešlo na XHTML (extensible hypertext markup language). Vydána byla v roce 1999. (KOSEK, 2013)

U verze HTML5 (5.0) někteří vývojáři nesouhlasili s vývojem a směrem technologií XHTML. Začali se zabývat možností obnovení HTML a v roce 2010 se dostává na svět nová verze HTML s označením 5.0. Opravuje předchozí verze a podporuje moderní technologie, které vznikly za šestnáct let od prvního využití HTML. (KOSEK, 2013)

HTML 5.1 představuje současnou verzi jazyka HTML, která se začala využívat v říjnu 2014. Opět doplňuje a vylepšuje prvky předchozí verze, především možnost tvorby offline webových aplikací a značky pro audio a video. Součástí této verze se stalo DRM (metody zajišťující ovládání digitálních technologií). (KOSEK, Jiří, 2013)

Další verze HTML 5 s označením HTML 5.2 se plánuje na rok 2018.

3.2.2. Editor pro grafiku HTML

Schafer (2009, str. 296) popisuje grafické editory jako prostředí pro tvorbu webových aplikací v jazyku HTML. Zde rozdělujeme dvě skupiny. Jednu ze skupin tvoří editory, které pracují s HTML na úrovni kódu, jedná se především o editory: Bluefish, Jedit, Notepad, PSPad. Druhou skupinu editorů tvoří tzv. WYSIWYG (What you see is what you get). Sem můžeme zařadit NVÚ, Mercury, Dreamweaver. Liší se od první skupiny tím, že při úpravě webu nemusí pracovat na úrovni kódu, ale v jednodušší grafické formě webových stránek.

3.3. Programovací skriptovací jazyk

Internet a prostředí kolem něho bylo a stále je jedno z nejrychleji se vyvíjejících odvětví informatiky a informačních technologií a pravděpodobně bude i nadále. Nástroje, které jsou využívány k tvorbě webu a webových aplikací, tedy musely projít také určitým vývojem.

Za období zrodu těchto skriptovacích jazyků můžeme považovat konec 80. let. Přesně v této době probíhá vývoj a vytvoření značkovacího jazyka HTML. Prvotní účel tohoto jazyka byl vědecký. Jazyk HTML se rychle rozšířil díky řadě konkurenčních výhod, např. nezávislosti druhu operačního systému na typu počítače, což v té době bylo převratné.

Postupem času s vývojem informačních technologií jeho jednoduchost a text, který byl formulován do několika tagů, nemohl stačit požadavkům vývojářů webových aplikací. Klienti si kladli čím dál větší požadavky a s tím rostla i náročnost na webové aplikace. Už nestačily jen statické údaje, ale důraz se začal klást také na dynamické prvky. Objevily se nápady, jak rozšířit jazyk HTML, např. formou nástaveb a nově se rodících skriptovacích jazyků. Tyto technologie se staly jednou z nejrychleji se rozšiřujících oblastí a dnes se

v praxi jen výjimečně setkáváme se statickými stránkami. Převážná část webových aplikací nyní spadá do dynamického podání.

3.3.1. Charakteristika programovacího skriptovacího jazyka

MacDonald, Szpuszta (2006, str. 34) charakterizují skriptovací jazyk. Slovo skriptovací nám charakterizuje skript, program uložený většinou jako soubor. Můžeme ho označit jako příkaz, který má za úkol vykonat nějakou strojovou instrukci. Skoro každý skriptovací jazyk je interpretovaný. Jedinou výjimku zde tvoří jazyky Java a ASP.NET, které jsou kompilované, a tudíž nepotřebují interpreta, ale překladače, jehož funkcí je přepsat zdrojový kód do strojového kódu. Kompilovaný jazyk tvoří malou základnu ve skriptovacích jazycích. Interpret vedle toho překládá program, až když je strojový kód již v běhu. Výhodou je snadnost zásahu, či úprava programu oproti spolupráci s překladačem. Jazyky spolupracující s interpretem jsou PHP, Python, Perl. Vkládáním algoritmů do značkovacích jazyků nám dává programovací skriptovací jazyk. Programování ve skriptovacím jazyce nám usnadňuje deklarace proměnných, o které se nemusíme starat. Vlastnost, která stojí za zmínění, je práce se složitějšími datovými typy a s asociativními poli. Funguje na principu klíčů a nikoli pomocí posloupnosti celých čísel. Není třeba deklarovat proměnné. Skriptovací programovací jazyky můžeme rozdělit na dva základní typy. První z nich funguje na bázi standardizace HTML a druhá skupina pracuje s webovými prohlížeči.

3.3.1.1. Index TIOBE

Tento žebříček neboli index nelze opomenout. Udává, jaké jednotlivé programovací technologie jsou v oblibě a na kolik procent se využívají. Jelikož index TIOBE zahrnuje všechny programovací jazyky, nikoli pouze skriptovací, na které se práce zaměřuje, je pro účely této práce každému jazyku přidělena hodnota žebříčku a procento využití s ohledem na celý balík programovacích jazyků. (TIOBE Software, 2015)

3.3.2. ASP.NET

Jedná se o technologii využívanou pro tvorbu webových prezentací, jak píšou autoři MacDonald, Szpuszta (2006, str. 31). Technologie pochází od vývojářů Microsoftu a spadá do skupiny .NET Framework. ASP.NET tvoří výjimku v podobě kompilace jazyka. Nejedná se tedy o jazyk interpretovaný.

```
<DOCTYPE html>
<html lang=en">
  <head>
    <meta charit="utf - 8"/>
    <title> Hello World Page</title>
  </head>
  <body>
    <h1> Hello World Page</h1>
    <p> Hello Worl</p>
  </body>
</html>
```

Kód Hello World – ukázka jazyka ASP.NET

Index TIOBE zobrazuje balík .NET na devátém místě s ratingem 2,026 %.

3.3.2.1. Historie

MacDonald, Szpuszta (2006, str. 32) nastiňují historii tohoto skriptovacího jazyka, která již podle názvu vychází z technologie ASP (Active Server Pages). Tento předchůdce se od dnešní formy ASP.NET, velmi liší. ASP.NET je založena na technologii CLR (Common Language Runtime). CLR umožňuje programování s funkcí řízení programů napsaných v některém z několika podporovaných jazyků. Výhoda CLR spočívá v široké škále podporovaných jazyků C#, Python, Perl a samozřejmě jazyky s koncovkou NET, jakými jsou VisualBasic.NET či JScript.NET. První verze ASP je datována ke konci roku 1996. V tomto roce nevzniká pouze tato technologie, ale i programovací skriptovací jazyk PHP. Oba tyto moderní skriptovací jazyky mají mnoho společného.

Microsoft přichází s inovací a v 90. letech vyvíjí zcela novou platformu, která dostává název .NET. Jde o soubor technologií vyvinutých Microsoftem.

Aktuální verze této technologie tvoří verze 5.0.

3.3.2.2. Výhody

Nepatrnou výhodou je vícejazyčnost, která tak dává jazyku větší funkčnost, jak tvrdí MacDonald, Szpuszta (2006, str. 38). Potencionálně zásadní výhodou je v objektově orientovaném programování. Jazyk může vytvářet opětovné třídy, standardizovat kód s rozhraním nebo začleňovat užitečné funkce. Další výhodou je podpora na různých platformách a druzích prohlížečů, a to jak těchto vývojově starších, tak moderních. Z těchto důvodů je technologie ASP.NET velmi populární.

3.3.2.3. Nevýhody

Velká nevýhoda podle MacDonald, Szpuszta (2006, str. 32) je závislost na jiné technologii. ASP.NET využívá komponentů, které jsou propojeny s JavaScriptem, ale většina komponent v ASP.NET sice Javascript využívá, ale funguje i bez něj. Za další nevýhodu můžeme označit menší podpora webhostingu oproti zmiňovaným technologiím.

3.3.3. Ruby

Fulton (2009, str. 45) přiřazuje Ruby ke skupině interpretovaných skriptovacích jazyků spadajících do skupiny objektově orientovaných jazyků, které mají neobvykle jednoduchou syntaxi. Proto jednoduchý na naučení a hojně využívaný. I přes jednoduchou syntaxi si však ve výkonnosti nezádá s jinými technologiemi. Těmi jsou například starší skriptovací jazyky Python a Perl, které jsou mu dosti podobné a z jejichž principů Ruby převážně vychází. Z toho je možné vyvodit, že Perl a Python jsou jakýmsi staršími bratry jazyka Ruby. Rozsah tohoto skriptovacího jazyka je poměrně široký, jelikož spadá do více skupin programovacích jazyků. Jako příklad lze uvést OOP – objektově orientované programování. Používá se zejména při tvorbě a psaní skriptů. I přes jednoduchou syntaxi se hodí i pro tvorbu obsáhlejších prací, jako jsou webové aplikace a větší obsahové projekty. Dokonce je vhodný pro práci s grafickým uživatelským rozhraním (GUI).

```
puts " Hello World";
```

Kód Hello World – ukázka jazyka Ruby

Index TIOBE ukazuje na dvacáté místo mezi všemi programovacími jazyky s ratingem 0,873 %.

3.3.3.1. Historie

Podle Fulton (2009, str. 39) je tvůrcem tohoto jazyka Jukihiro Macumoto, japonský programátor, jemuž se v IT oblasti přezdívá Matz. Zaměřil se na práci s jazykem Perl. Kvůli malému výkonu Perlu se však záhy rozhodl přesedlat na Python. Ani s ním však nebyl spokojen kvůli malé podpoře objektivní orientace jazyka. Proto se rozhodl napsat a vytvořit vlastní originální jazyk. Pracovat na něm začal v roce 1993. Trvalo mu další dva roky, než první verze jeho programovacího jazyka byla uvedena do chodu, k čemuž došlo v roce 1995. Výtvar dostal název po drahokamu Rubín, v anglickém překladu Ruby. Nejdříve se novému skriptovacímu jazyku nedařilo prosadit se na trhu. Důvodem byla

chybějící kvalitní část dokumentace v anglickém jazyce. V současnosti je Ruby rychle se rozšiřující interpretovaný jazyk, a to nejen v Japonsku.

Aktuální verze jazyka Ruby je 2.1.3.

3.3.3.2. Výhody

Ruby obsahuje všechny prvky OOP, jak uvádí Fulton (2009, str. 59), což je nesporná výhoda, která umožňuje velké uplatnění jazyka na přeplněném trhu s technologiemi pro tvorbu webových aplikací. Mezi další výhody řadíme přenositelnost jazyka na různé platformy. Jazyk funguje na operačním systému Linux nebo Solaris. Dokonce do této skupiny spadá i operační systém MS Windows. Výhodná je i jeho syntaxe, která díky své jednoduchosti činí Ruby oblíbeným jazykem u programátorů, kteří s programováním teprve začínají.

3.3.3.3. Nevýhody

Hlavní nevýhodou je přetrvávající problém s dokumentací v jiných jazycích. Za nevýhodu můžeme považovat také závislost na interpretovi, jak tvrdí Fulton (2009, str. 40).

3.3.4. Perl

Podle Satrapy (2000, str. 13) spadá Perl do skupiny interpretovaných skriptovacích jazyků. Perl dosáhl na poli této skupiny jazyků věhlasu a popularity. Dařena (2005, str. 22) uvádí výstižné motto charakterizující Perl: „Existuje více způsobů, jak něčeho dosáhnout“. Jeden způsob, jakým si zajistil jazyk slávu, je velká škála možností psaní a tvorby CGI (Common Gateway Interface), což je komunikační protokol skriptů. Tento skript je externí program, který generuje WWW stránky, které po otevření mohou změnit svůj vzhled. Co se týká oblasti práce se soubory, tak se jedná o to nejlepší, co nám tato přední technologie nabízí. Vyjadřovací schopnosti Perlu jsou obrovské. Jazyk funguje na licenci Open source, i proto je velmi oblíbeným jazykem pro tvorbu webových aplikací. Jeho příbuznými skriptovacími jazyky jsou Python a Ruby.

```
#!/usr/bin/perl
Use strict;
print "Hello World";
print"\n";
```

Kód Hello World – v jazyce Perl

Index TIOBE přiděluje jazyku Perl třinácté místo s ratingem 1,390 %.

3.3.4.1. Historie

Dařena (2005, str. 20) uvádí jako zakladatele jazyka kanadského programátora Larryho Walla, který chtěl napsat jazyk vyhovující jeho vlastním potřebám. Jeho cílem bylo přijít s programem, který by byl jednodušší než v té době používaný programovací jazyk C++. Původně byl jazyk Perl navržen na unixovou platformu. V současnosti pracuje ve velké škále operačních systémů jako např. MS-DOS, MS Windows, OS/2 a další.

Aktuální verze, která je dostupná, má označení 5.18.2.

3.3.4.2. Výhody

Dařena (2005, str. 28) uvádí, že výhodou jazyka je jeho snadná interpretovatelnost, která není tolik časově náročná a interpret se stará o rozhraní mezi uživatelem a jazykem. Podle Satrapy (2001, str. 13) je největší výhodou podpora regulárních výrazů. Představují silný nástroj při práci s textovými informacemi. Nesmíme rovněž opomenout práci s asociativními poli, která řeší problémy s vyhledáváním. Za výhodu můžeme také považovat rozsáhlou dokumentaci a odbornou literaturu. Poslední důležitou přednost tvoří jednoduchost. Jazyk je stavěn na jednoduchosti, nemusíme proto podrobně znát jazyk, abychom mohli psát složitější programy.

3.3.4.3. Nevýhody

Autor Dařena (2005, str. 21) vidí nevýhodu jazyka Perl v syntaxi. Tento soubor pravidel není pevně definovaný. U jazyka nelze definovat vlastní datové typy a neexistuje typová kontrola, proto není příliš vhodný pro výuku programování.

3.3.5. Python

Lutz, Ascher (2003, str. 3) tvrdí, že se jedná o jednoduchý dynamický skriptovací interpretovaný jazyk, který patří do kategorie objektivně orientovaného programování, ale i do jazyků procedurálních. Díky tomu ho lze označit za hybridní jazyk, jenž je založený na otevřené licenci Open source (licence poskytnutá vývojářům zdarma pro její rozvoj a vývoj). Jeho tzv. sourozenecké jazyky jsou Perl a Ruby. Je navržen pro tvorbu rozsáhlejších úloh a dokáže využívat také rozhraní GUI – uživatelské rozhraní. Lze jej označit za jazyk vyšší úrovně, což je jeho velkou výhodou. Jeho přehlednost a čitelnost ho řadí před výše uvedené sourozenecké jazyky.

Harms, McDonald (2006, str. 11) informují, že tento skriptovací jazyk je tvořen programovacími balíky. Balík se v jazyce Python skládá ze tří částí. První částí je překladač s PVM (Python Virtual Machine). Druhá část představuje rozsáhlé knihovny, které již obsahují již předdefinované programové struktury. Poslední část balíku Python je textový editor, který v sobě má zabudovaný překladač IDLE (vývojové prostředí pro Python). Na něj navazují způsoby, jak tento překladač využít. Existuje druh využití - shell mode. Tento mód, lze chápat jako režim psaní kódu. Jedná se vlastně o interaktivní režim.

```
%python
>>> print"Hello World"
Hello World!
```

Kód Hello World – ukázka jazyka Python

Index TIOBE řadí Python na osmé místo mezi programovacími jazyky s 2,882 % ratingem.

3.3.5.1. Historie

Procházka (2012, str. 74) poukazuje na nizozemského programátora Guido van Rosse, který navrhl a vytvořil tento programovací jazyk roku 1991. Zajímavostí je název skriptovacího jazyka, který vznikl z názvu hudební skupiny Monty Python.

Aktuální verze jazyka má označení 3.4.

3.3.5.2. Výhody

Lutz, Ascher (2003, str. 4, 5) považují za výhodu jazyka Python, že se jedná o jazyk OOP a s tím spojený polymorfismus, dědičnost i přetěžování operátorů, se kterými umí pracovat. Další výhodou je přenositelnost. Je možné ho spustit skoro na každé platformě. Výkonnost je v Pythonu nezvykle veliká, neboť v něm můžeme programovat jak v „klasických“ jazycích (C, C++, Java), tak jazycích skriptovacích (např. Perl) což nám dává relativně velké pole působnosti. V neposlední řadě je nespornou výhodou Open source licence - skutečnost, že jazyk je skoro zadarmo.

3.3.5.3. Nevýhody

Harms a McDonald (2006, str. 11) spatřují nevýhodu v nízké rychlosti. Jazyk je totiž interpretovaný, a nikoli kompilovaný, který bude vždy rychlejší.

3.3.6. JAVA

Autor Perry (1996, str. 16) představuje jazyk jako objektivně orientovaný programovací jazyk. Oblíbenost a škála působnosti Javy je obrovitá, spadá také do kategorie technologií pro tvorbu webových aplikací, proto tento jazyk nelze vynechat. Do jisté míry se však liší od předešlých jazyků, jako je Ruby a ostatní. Rozdílný je v tom, že ho můžeme považovat za kompilovaný jazyk místo ostatních interpretovaných, které využívají svého interpreta. Kompilovaný jazyk využívá překladače, který přeloží zdrojový kód do kódu strojového. Výhodou je zrychlený průběh programu, ale nevýhodu naopak představuje začátek programu při jeho spouštění. Java je vhodná pro webové úlohy se středním či menším obsahem s postupným rozšiřováním, až do velkých úloh.

```
response.setContentType("text/html; charset=utf-8");
    PrintWriter out = response.getWriter();
    out.println("<html><body>");
    out.println("<h1>Hello World!</h1>");
    out.println("</body></html>");
```

Kód Hello World – ukázka jazyka Java

Index TIOBE uvádí Javu na druhém místě s ratingem 15.345 %.

3.3.6.1. Historie

Perry (1996, str. 22, 23) přibližuje historii jazyka a ukazuje na hlavního tvůrce a otce Jamese Goslinga, který dostal od prezidenta společnosti Sun, Billa Joye za úkol

vymyslet program pro domácí spotřebiče. Vybral si programovací jazyk C++. Se svým týmem Green team pracoval na splnění tohoto cíle přibližně dva roky. Vývojáři po čase zjistili, že zvolený program není pro tyto úlohy vhodný a přešli, na jazyk Pascal, který rovněž neuspěl. Proto se Gosling zabýval myšlenkou vytvořit vlastní program, který pojmenoval Oak a který vychází ze syntaxe C++. V této fázi byl jazyk interpretovaný, ale jak zjistíme z vývoje dalších verzí, od této technologie se později upustilo. Větší rozbor historie nastiňuje kapitola 3.3.6.4.

3.3.6.2. Výhody

Perry (1996, str. 30) spatřuje hlavní výhodu v rychlosti vývoje aplikací. Na Javu se díky její populárnosti a široké použitelnosti zaměřuje velké procento vývojářů, uživatelů, technologů a dalších. To umožňuje pružně reagovat na požadavky trhu prostřednictvím nejdůležitějších novinek, a být tudíž konkurenceschopný. Poslední dobou s rozmachem androidu a mobilních zařízení se jedná převážně o aplikace a webové technologie. Využití Javy je umožněno na široké paletě platform a informačních nosičů. Rozsahem lze Javu využít jak v malých, tak velkých projektech a úlohách. Jednoduchá je i pro programátory – začátečníky, neboť při práci s Javou se dají snadno dohledat vzniklé chyby.

3.3.6.3. Nevýhody

Perry (1996, str. 40) poukazuje na jazyk, který hardwarově náročný, a proto je pro některé uživatele nepoužitelný. Nevýhodou a zároveň výhodou je její rozšíření, jež může být v určitých chvílích více zneužitelná, oproti jazykům, které nejsou tolik na očích a na které se nezaměřuje tolik pozornosti. První verze byla poměrně pomalá, s novějšími verzemi se práce s Javou skokově zrychluje, což může některým uživatelům činit v začátcích problémy.

3.3.6.4. Verze

Roku 1995 se jazyk Oak přejmenovává na jazyk Java. Jméno dostává po kávě, která je ve slangu označení jako Java. (Novotný, 2003)

Roku 1996 přichází do balíku Java Development (zkráceně JDK) 1.0. Jedná se o nástroje pro tvorbu aplikací na platformě Java. Zásadní pro JDK byla práce se slovy `protected` a `private`. (Novotný, 2003)

O rok později, tj. 1998 je vydáno J2SE 1.2., což je označení pro verzi Playground, která se později změnila na Java 2. Přichází rozšíření Unicornu do japonštiny. Pohrává si s grafickým rozhraním pomocí knihovny Swing. (Novotný, 2003)

Verze s označením Kestrel J2SE 1.3, která pracuje s proxy třídami, vzniká v roce 2002. Dochází k vylepšení zvuku pomocí Javasound a dalším zlepšením předchozí verze. (Novotný, 2003)

Roku 2002 sice vzniká J2SE 1.3, ale až roku 2004 vzniká verze Java 5, která je přelomová po stránce označení, když vynechává jedničku a rovnou se nazývá Java 5. Přináší velké množství novinek např. generics, autoboxing/unboxing, výčtové typy, anotace atd. (Novotný, 2003)

S novým desetiletím vzniká verze Java SE 7, která vychází roku 2011. Do obsahu však nebyly zahrnuty veškeré prvky, které vývojáři chtěli v této verzi aplikovat. Ty se později objevují s verzí Java SE 8. (Novotný, 2003)

Java SE 8 představuje doposud nejnovější a nejrozsáhlejší balík Java. Spolupracovali na něm týmy z ORACLE a vývojáři v komunitě OpenJDK Community. Změna se týká nové API pro datum a čas, zdokonalení anotace a kolekce kódu. Podporuje výrazy lambda a poskytuje efektivní podporu vícejádrových procesorů. (Novotný, 2003)

4. PHP

Podle Ullmana (2004, str. 12) je PHP rekurzivní zkratka pro PHP: Hypertext Preprocessor. Původně zkratka vycházela z označení Personal Home Page. Jedná se o programovací skriptovací jazyk na bázi Open source. PHP spolupracuje se serverem Apache a jinými technologiemi pracujícími na obdobných principech. Apache byl v rámci této práce vybrán z důvodu jeho velkého rozšíření a praktičnosti. PHP a server Apache jsou nabízeny pod stejnou licenci Open source, tedy otevřenou licenci s přístupem do zdrojového kódu. Díky svému rozšíření se jedná o jednu z nejvíce používaných skriptovacích technologií. Open source má tu výhodu, že modifikace lze udělat podle potřeb daného uživatele. S tím souvisí řada knihoven a dalších použitelných nástrojů. Můžeme tedy říci, že se jedná o jazyk pro tvorbu či programování dynamických webových aplikací, většinou ve formátu HTML nebo XHTML, který je nejpoužívanější pro tvorbu webových aplikací ve vybraném prostředí, jako jsou např. PSPad nebo Sublime Text 3. O PHP můžeme také tvrdit, že se jedná o vyšší jazyk, který sebou přináší zajímavé výhody a výsledky.

Při použití PHP pro dynamické stránky jsou skripty prováděny na straně serveru k uživateli, kterému přenáší až výsledek jejich činnosti. Gilmore (2007, str. 359) tvrdí, že interpreta PHP skriptu je možné volat pomocí příkazového řádku, dotazovacích metod protokolu HTTP nebo pomocí webových služeb. Syntaxe jazyka je inspirována hned několika programovacími jazyky (Perl, C, Pascal a Java). PHP je nezávislé na platformě. Rozdíly v různých operačních systémech se omezují na několik systémově závislých funkcí a skripty lze většinou mezi operačními systémy přenášet bez jakýchkoli úprav.

Procházka (2015, st. 69) tvrdí, že PHP podporuje mnoho knihoven pro různé účely, např. zpracování textu, grafiky, práci se soubory, přístup k většině databázových systémů (např. MySQL, Oracle, PostgreSQL) nebo podporu celé řady internetových protokolů (např. HTTP, FTP).

Procházka (2015, st. 70) prezentuje PHP jako nejrozšířenější skriptovací jazyk pro web. V současnosti se podílí na doménách skoro z 85 %. Oblíbeným se stal především díky jednoduchosti použití a bohaté zásobě funkcí. V kombinaci s operačním systémem

Linux, databázovým systémem především MySQL nebo vytvořeny pro linux, PostgreSQL a webovým serverem Apache je často využíván k tvorbě webových aplikací. Pro tuto kombinaci se vžila zkratka LAMP neboli spojení Linux, Apache, MySQL a PHP, Perl nebo Python.

```
<html>
  <body>
    <?php
      echo "Hello World"
    ?>
  </body>
</html>
```

Kód Hello World – ukázka jazyka PHP

Index TIOBE zařazuje jazyk PHP na sedmé místo s ratingem 3,170 %.

4.1. Historie

Gilmore (2007, str. 31) uvádí, že historie PHP se začala psát roku 1994, kdy dánský programátor Rasmus Lerdof napsal v programu C binární část skriptů pro své potřeby. Tyto skripty pojmenoval Personal Home Page, což bylo prvopočáteční smysl zkratky PHP. V témže roce vytvořil program FormInterpreter, který poté skloubil dohromady s PHP, čímž vzniká daleko rozsáhlejší verze PHP/FI s větším polem působnosti a funkcností. Výhodou tohoto programu byla možnost komunikace s databázemi. Díky této komunikaci mohly vzniknout základy dynamických webových aplikací. Oficiální představení technologie PHP/FI proběhlo v roce 1995. Ještě v roce 1997 pokrývala PHP/FI jen kolem 1% všech domén, což představovalo přibližně 50 000 domén. Postupem času se však tato technologie dostávala na stále větší množství domén.

Obrat nastává s rokem 1997, kdy vzniká verze PHP 2. Gilmore (2007. str. 31, 32) uvádí, že tato verze se již v takové míře neliší od dnešní verze fungující na stejném základu a systému. Autor verze ho považoval za jeden z nejrychlejších a nejjednodušších nástrojů pro tvorbu dynamických webových stránek. Poté přichází v červnu 1998 verze PHP 3. Pro zajímavost lze uvést, že Lerdof na PHP 3 spolupracoval s izraelskými studenty a vývojáři Zeevem Suraskim a Andim Gutmansem. Ti pátrali po nějaké nové technologii, která by vyřešila jejich problém s elektronickým obchodem. V tomto roce se zastavuje vývoj PHP/FI. Výhodou třetí verze je její konzistentnost, snadná rozšiřitelnost a výkonnější

syntaxe. Starší verze se již, co do pole působnosti, nemohly měřit s třetí verzí, která zajímala v té době odhadovaný podíl všech domén 10 %, což představuje statisíce různých domén.

Gilmore (2007, str. 32) nastiňuje situaci kolem pokroku další verze s označením PHP 4.0, která vznikla roku 2000. Začalo se pracovat s novým jádrem, označeným a pojmenovaným Zend Engine. Jméno vzniklo ze spojení křestních jmen Zeeva Suraskiho a Andiho Gutmanse. S touto verzí přišlo další rozšíření a změny, např. vyšší výkonnost, super globální proměnné či samotná podpora pro server. Na trhu verze obsáhla 20 % domén, tj. v počtu několika milionů. V tomto stavu byla následně nainstalována na server.

Gilmore (2007, str. 32) uvádí, že v roce 2004 byla prezentována verze PHP 5, která se s různými inovacemi používá do současnosti. Zásadně rozvíjí předchozí verzi a opět je postavena na novém jádru Zend Engine II. Začíná se pracovat s tzv. objektivně orientovaným programováním. Byly přidány destruktory a kontraktory, zaměřuje se na podporu XML, MySQL apod. Roku 2008 se stává PHP 5 jedinou stabilní verzí, která se stále vyvíjí.

Podle autora Procházky (2012, str. 71, 72) je největší výhodou PHP 6 provázání s Unicode. Tato verze se již bude kódovat v označení UTF-16, což nabízí, že jeden nebo dva znaky budou zabírat dva nebo čtyři byte.

4.2. Výhody

Ullman (2004, str. 13) říká, že PHP má vysokou specializaci pro tvorbu webových stránek a jako technologie je z tohoto pohledu nejrozšířenější. Můžeme vytvářet dynamické nebo statické webové aplikace díky prolnavosti zbytku dokumentu ve statickém formátu. Licence v podobě open source dává možnost snadné dostupnosti a rychlého rozvoje. Nabízí se na většině serverů. Co se týká databázových systémů, je další výhodou nativní podpora, která je implementována přímo do prohlížeče. Dnes je běžným standardem, který se objevuje v drtivé většině procesů tvorby webových aplikací.

4.3. Nevýhody

Autor Procházka (2012, str. 70) uvádí jako jednu z hlavních nevýhod skutečnost, že jazyk je interpretovaný a nikoli kompilovaný, což znamená velkou závislost na interpretovi. Rychlost může navíc zatěžovat server. Důležitou nevýhodou bezesporu

představuje pojmenování funkcí, které je v řadě případů nekonzistentní, např. jméno_funkce i jménofunkce. Další nevýhodou jsou vyšší požadavky na webhosting, což je pronájem prostoru pro webové stránky na WWW serveru. Webhosting bývá na PHP náročnější, a to se projevuje i v ceně. Stává se trendem, že nemá podporu SQL a zejména PHP. Jazyk je pomalejší v porovnání s jazyky, které jsou kompilované, což může předznamenat přechod na jinou technologii např. Javu. Jeho volnost může v některých případech vést ke komplikacím zejména při změně programátora. Nový programátor může mít problémy se čtením kódu, který byl vytvořen v minulosti. Předjetí problému se dá kvalitní dokumentací a schématy či popisky.

5. Vlastní webová demonstrace

Obsahem této části práce bylo vytvoření webové prezentace a praktická ukázka PHP prvků charakterizující tuto dynamickou technologii.

5.1. Charakteristika

Webové stránky jsou tvořeny pro menší firmu, zabývající se operacemi s nemovitostmi v podobě prodeje nebo pronájmu nabízených objektů. Firma s názvem Vrchol bydlení se snaží nabídnout zákazníkům ty nevhodnější podmínky pro výběr jejich vysněného bydlení. Charakteristika je jednoduchá, aby co nejlépe demonstrovala základy a principy tvorby webových stránek. Vložený text je Lorem Ipsum. Jedná se o pseudolatinský text, který se využívá v celé IT oblasti jako nepsaný standard pro výplň demonstrativního textu.

5.2. Struktura webu

Rozsah webové aplikace není příliš obsáhlý a struktura webu je velmi jednoduchá. Ihned v úvodu nabídne stránka návštěvníkům záložky k prohlížení a představení struktury webu. Některé ze záložek nabízejí pouze orientační nebo informační data např. Domy k pronájmu. Záložka Kontakty nám již nabízí trochu akčnosti v podobě kontaktního menu. Poslední z nabízených záložek představuje funkci přihlášení se na stránky s menší podzáložkou Registrace. Vše završuje ikona domečku v podobě loga společnosti, která vrací návštěvníka na stránku Domů.

Demonstrativní ukázka webových stránek obsahuje pět hlavních záložek:

Domů – Hlavní stránka webu, která uživatelům nebo návštěvníkům zobrazuje prezentaci webu, strukturu stránek, fotografie a celkové základní informace o webu.

Domy k pronájmu – Tato stránka informuje návštěvníka o nabízených nemovitostech k pronájmu s vizualizací objektu k pronájmu. Vizualizaci lze využít, pokud je návštěvník zaregistrován a přihlášen na svém účtu po přenesení kurzoru myši na fotografii nemovitosti.

Domy k prodeji – Po rozkliknutí je stránka obdobná jako předešlá záložka Domy k pronájmu, ale nemovitosti jsou nabízeny formou prodeje.

Kontakty – Již z názvu zjistíme, že záložka nabízí kontaktní údaje a formulář k zasílání požadavků zákazníků o informace prodejci daného sortimentu.

Přihlášení – slouží k přihlášení ke svému účtu, ze kterého uživateli plynou výhody v podobě nabídky nejrůznějšího sortimentu.

Obrázek 3: Ukázka hlavních položek webu

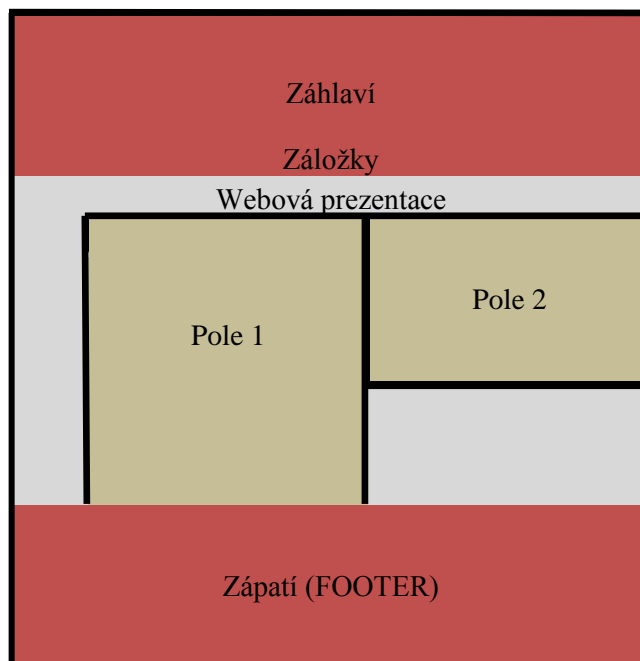


Zdroj: Vlastní zpracování

5.2.1. Layout

Navrhované grafické rozvržení stránky by mělo být tvořeno přehlednými prvky se základními vlastnostmi většiny webů. Horní stránku obrazovky webové aplikace tvoří záhlaví. Jedná se o horní okraj, do kterého bylo vloženo imaginární logo společnosti a jednotlivé záložky (stránky) viz obr. 4. Dále nám layout nabízí pole jedna. Toto pole je vlastně informace v podobě textů, nadpisů, obrázků nebo tabulek s kontakty na společnost (viz záložka Kontakty). Pole dva je tvořeno na podobném principu jako pole jedna, ale je menší, což dle mého názoru pozitivně působí na vzhled stránek. Toto menší pole již neobsahuje tolik funkcí a textu. V názvu Webová prezentace je ukryto pozadí celých stránek, které by mělo na uživatele působit jako graficky doplňující a uklidňující prvek. Poslední složkou je Zápatí neboli Footer. Můžeme ho vidět na dolní hraně schématu ohraničující konec stránky. Je nositelem informací o aktuálním roce a provozovateli webu. Při volbě layoutu stránky byl kladen důraz na centralizaci vizuálního vzhledu webu, tak aby v centru stránek byly z pohledu uživatele nejdůležitější informace. Schéma bylo vytvořeno jako názorná ukázka, jejímž smyslem je co nejvíce přiblížit strukturu webové stránky širšímu spektru čtenářů, a to jak začátečníkům, tak zkušeným a nadšeným vývojářům webových aplikací. Barvy byly zvoleny s ohledem na snadnou orientaci a porozumění schématu, každá jednotlivá složka je výrazná oproti druhé, ale zároveň poskytuje kompletní náhled na schéma jako celek.

Obrázek 4: Grafický layout webových stránek



Zdroj: Vlastní tvorba

5.2.2. Obsah vytvořeného příkladu

Nedílnou součástí a důležitou složkou je vlastní obsah webových stránek. Uživatel, který stránky nevyvíjel nebo není jejich vlastníkem, má možnost vidět pouze ty informace, které mu vývojář webu zobrazí, většinou pouze služby a vizualizace, které nám web prezentuje. Tato část je však limitovaná v oblasti struktur, kódu a celkové ideje stránek. Bez kódu by webové stránky tvořené především syntaxí HTML a dalšími prvky nepředstavovali prakticky nic a jejich funkčnost by byla nulová.

5.2.3. HTML struktura

Z teoretické části již víme, že jazyk HTML je základním stavebním kamenem pro tvorbu webu. Skládá se z jednotlivých tagů neboli značek.

```
<!doctype html>
```

Tato část kódu nám říká, že se jedná o standardní stránku v Internet Exploreru. Tento řádek se umísťuje jako úplně první ještě před základním tagem <html>, který je, jediným ze tří základních prvků pro stavbu a strukturu webových stránek.

```
<html>
```

Značkou `<html>` začíná a končí celá struktura dokumentu. Jedná se o zásadní prvek, který je párový. Na obr. 2 uvidíme princip tohoto HTML tagu a skladbu kostry HTML dokumentu.

```
<head>
  <meta charset="utf-8">
  <title>Hurt</title>
  <link href="style.css" rel="stylesheet" type="text/css">
</head>
```

Hlavička (header) `<head>` obsahuje párovou značku. Vlastní a definuje technické údaje, jakým i jsou titulek `<Title>`, metadata, apod. Tvoří druhý základní prvek pro strukturu internetových stránek. *Meta charset* ukazuje, v jakém kódování je webová aplikace zvolena. *Utf-8* je nepsaným standardem kódování. Třetí řádek vyjadřuje titulek `<title>`. Již podle názvu zjistíme, že se jedná o prostor vymezený pro text vyjadřující nadpis nebo titul webové stránky. Poté následuje tag, který v sobě obsahuje konstrukci kaskádových stylů. Celkově řádek udává formátování a vykreslení vzhledu stránky, které se odkazují na složku `style.css`. Poslední značka uzavírá jednu ze tří důležitých částí HTML dokumentu.

```
<body>
  <?php
    include "layout/menu.php";
  ?>
  <div class="container">
    <h1>Vrchol bydlení</h1>
    <?php include "layout/asides/index_aside.php";
  ?>
  <article class="content">
```

První část tohoto kódu tvoří značka `<body>`. Jedná se o třetí element HTML dokumentu, který je pro tvorbu webu důležitý viz obr. 2. Tag znázorňuje tělo celé technologie HTML. Vkládá se do něj veškerý text určený k zobrazení. Druhý řádek nám ukazuje prvek PHP jazyka. Zobrazuje vložený kód a informuje nás, že načítá obsah z `layout/menu.php`. Značka `div` v pátém řádku poukazuje na obsah `container`, který je vyčleněn pro prostorový oddíl na stránce. Tagy, které svírají *Vrchol bydlení*, označují

stupeň nadpisu. Poslední řádek je podobný řádku třetímu, avšak odebírá obsah z jiné složky.

V záložce Kontakt můžeme vidět předpřipravený formulář určený ke kontaktování firmy se zákaznickými požadavky nebo různými speciálními dotazy. Jelikož jsme si většinu značek a tagů vysvětlili na prvních třech záložkách webových stránek, lišících se pouze obsahem textu, fotek a pár dalšími maličkostmi, v následující ukázce uvedeme pouze ty značky, které předešlé tři stránky neobsahovaly.

```
<p>Lorem ipsum...</p>
<h2>Akční nabídky</h2>
<p>Ut suscipit ultrices vel.</p>
<h2>Domy na prodej</h2>
```

První řádek označuje odstavec, který obsahuje pseudolatinský text Lorem ipsum sloužící pro výplň textu. Následující řádek obsahuje nadpis úrovně dva, což v rámci škály jedna až šest hodnotí nadpis jakožto důležitý informační zdroj. Třetí a čtvrtý řádek kopíruje řádky předešlé.

```

  </section>
<section>
  <h2>Domy k pronájmu</h2>
  
  

</section>
<!-- end .content --></article>

<!-- end .container --></div>
  <?php
include "layout/footer.php";
  ?>
```

První řádek prezentuje znak, poukazující na obrázek domu. Za kódem obrázku jsou uvedeny hodnoty, které udávají rozměry obrázku a odkaz na složku, ve které je obrázek umístěn. Element <section> spadá do kategorie značek HTML5 a označuje část oddílu nebo sekce.

```

<tr>
  <td>
    <label>Jméno:</label>
  </td>
  <td> <input type="text" name="firstname" id ="firstname">
  </td>
</tr>

```

Z výše uvedené struktury můžeme vyčíst typickou část tabulky v záložce Kontakty. Tag `<tr>` značí řádek tabulky a je párový proto ho můžeme vidět rovněž na konci kódu s lomítkem. Druhý řádek obsahuje značku, vytvářející buňku tabulky. Jedná se také o párový znak. Následující řádek utváří textové pole s předpisem *Jméno*. Značka `<input type>` reprezentuje vstupní políčko pro text.

```

<select name="platforma" size="1">
  <option value="€90000">€90000
  <option value="€100000">€100000
  <option value="€150000">€150000
  <option value="€500000">€500000
  <option value=">€1million">>€1million
</select>

```

Option value vytváří řetězec, který je odesíláný jako pole. Zobrazuje znaky uvedené za rovnítkem. Na stránkách má zákazník možnost si zvolit si pro něj cenově dostupný dům. Ve vybrané cenové kategorii.

```

<label><input type="checkbox" checked name="skrtatko1" value="ne">1
Pokoj</label>

```

Kód popisuje checkbox s nápisem *1 Pokoj*.

```

<option value="Semi-detached">Řadový
<option value="Detached">Patrový
<option value="Apartments">Přízemí

```

Výše uvedený kód funguje podobně jako při volbě ceny objektu. V tomto případě však zákazník volí typ domu. Kód tedy odesílá text: *Řadový, Patrový, Přízemní*.

```

<textarea rows="6" cols="20" name="longtext">speciální požadavky

```

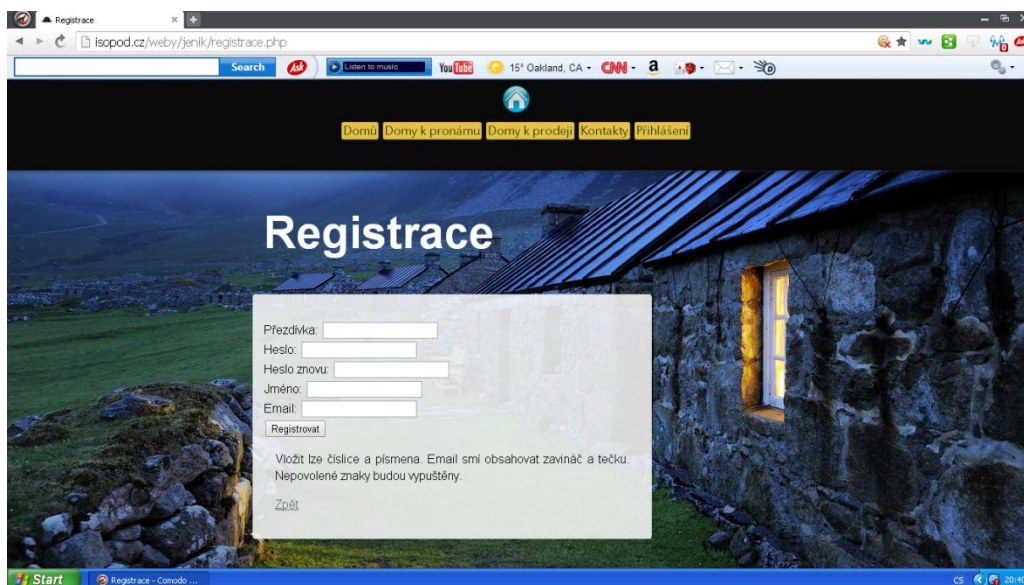

Tato část kódu vytváří vstupní pole pro vkládání dalšího textu. Na stránkách umožňuje zákazníkům do vstupního pole vkládat specifické dotazy.

5.3. Hlavní prvek PHP

Dostáváme se k nejdůležitější části našeho ukázkového příkladu. Jedná se o prvek PHP, který přetváří statické stránky, na webovou dynamickou aplikaci. PHP prvky se vkládají přímo do jazyka HTML. Soubory obsahující technologii PHP mají koncovku `.php`, na rozdíl od klasických HTML souborů.

Nejdůležitější a zásadní prvek se nachází v sekci týkající se přihlášení a registrace v podzáložce, Registrace. Pro správné fungování tohoto skriptu bylo zapotřebí několik složek. V první řadě se jedná se o samotný jazyk PHP, ve kterém byl skript vytvořen na základě jeho syntaxe. Tento jazyk jsme si představili v kapitole 4., proto se obecnému popisu jazyka nebudeme již více věnovat. Další složku tvoří databáze MySQL. Začleněním této složky do webové aplikace můžeme určitá data generovaná jazykem PHP načítat z databáze MySQL. Touto složkou se více zabývá a podrobněji ji vysvětluje kapitola 5.4.1. Třetí důležitou složku tvoří aplikace `phpMyAdmin`. Tato aplikace je důležitým spojem mezi PHP a MySQL prostřednictvím webového rozhraní. Objasněná je v kapitole 5.4.2. Poslední složku, která byla zapotřebí pro tvorbu naší praktické ukázky, tvoří Apache. Jedná se o webový server, bez kterého by samotné použití PHP nemělo žádný význam. Apache a PHP spolu tvoří vynikající tým pro vývoj webových aplikací. O serveru Apache se více dozvídáme v kapitole 5.4.3.

Obrázek 5: Grafické znázornění dynamického prvku



Zdroj: Vlastní tvorba

Při registraci návštěvník vyplňuje následující položky: Přezdívka, Heslo, Heslo znovu, Jméno, Email. Registrující je upozorněn, které znaky umí dynamický prvek přečíst a které naopak vynechá. Po vyplnění zmíněných položek potvrdí uživatel uvedené údaje tlačítkem *Registrovat*. Odeslaný požadavek tímto úkolem přechází na skript, který vidíme níže. Určitě se nabízí otázka, co vlastně skript představuje? Skript rozhodně není posloupnost nějakých znaků, či dokonce jenom pár písmen. PHP skript je vložen do jazyka HTML a sám je ohraničený, takže ho lze při čtení kódu jednoduše rozpoznat. Tyto skripty jsou odesílány na server, a uživatel tak vidí až výsledek. V našem případě uživatel tedy pouze zadá již zmiňované položky. Požadavek je odeslán a zobrazí se výsledek zpracování, a to buď kladný, nebo záporný. Ukázku našeho skriptu pro registraci můžeme vidět níže. Jelikož je skript obsáhlý, dovolil jsem si v ukázce vybrat pro čtenáře pouze tu část skriptu, která je, s ohledem na zde popisovanou funkčnost nejdůležitější.

```

?php

include 'dbcon.php';
include 'functions.php';
sec_session_start();
if(isset($_POST['nick'], $_POST['p'], $_POST['c'], $_POST['name'],
$_POST['email'] ))
{
if ($stmt = $mysqli->prepare("SELECT ida, nick, pass, name, salt FROM
AUTOR WHERE mail = ? LIMIT 1"))
{
$stmt->bind_param('s', $_POST['email']); // Bind "$email" to
parameter.
$stmt->execute(); // Execute the prepared query.
$stmt->store_result();
$stmt->bind_result($user_id, $username, $db_password, $name, $salt);
// get variables from result.
$stmt->fetch();
if($stmt->num_rows == 0) // If the user does not exist
{
if( $_POST['p'] == $_POST['c'] )
{
$nick = $_POST['nick'];
$pass = $_POST['p']; // The hashed password.
$name = $_POST['name'];
$mail = $_POST['email'];
if(registration($nick, $pass, $name, $mail, $mysqli) == true)
{
// Login success
echo "<script>

alert('Successful registration!');
else
{
// Registration failed
echo "<script>

```

S ohledem na značnou obsáhlost budou vysvětleny pouze zásadní prvky skriptu. První podmínka *If* odesílá proměnné např. *nick*, *email* apod. Druhé *If* spolupracuje s tabulkou *AUTOR* a odkazuje se na podmínku. V dalším kroku přijímá databáze proměnné parametry. Dále ve skriptu ve třetí podmínce *If*, jsou parametrům proměnných přiřazeny hodnoty. V rámci posledního kroku, po získání proměnných a ověření správnosti podmínky, je uživatelský účet přijat. V opačném případě, se jedná o neúspěšný pokus.

```
<?php
    include_once 'includes/db_connect.php';
    include_once 'includes/functions.php';
    sec_session_start();
    if (login_check($mysqli) == true) {
        $logged = 'in';
    } else {
        $logged = 'out';
    }
?>
```

Tento skript nám ukazuje princip logování. Uživateli je po registraci, kterou jsme si vysvětlili v předešlém skriptu, přidělen vlastní účet, na který se může přihlásit. Po registraci obdrží heslo a uživatelské jméno. Na obr 6. můžeme obě tyto položky vidět v prostředí phpMyAdmin. Položku s uživatelským jménem zaregistrujeme pod tabulkou AUTOR, která je popsána v kapitole 5.4.2. Do tabulky HESLA se ukládá heslo, s jehož pomocí se přihlašujeme. Skript nad textem zobrazuje podmínku pro přihlášení. Po správném vyhodnocení podmínky se přihlašující dostává do stavu „in“. Pokud algoritmus vyhodnotí přihlášení jako nesprávné, uživatel je stále ve stavu „out“. Tyto dvě položky poté slouží jako vstupní klíč do uživatelského účtu.

5.4. Sekundární software

Kapitola uvádí charakteristiku jednotlivých sekundárních softwarů, které jsme pro tvorbu naší praktické ukázky potřebovali.

5.4.1. MySQL

Vychází ze zkratky SQL a jedná se o relační databázi. V naší aplikaci představuje prvek, bez kterého bychom nebyli schopni vytvořit dynamické webové stránky. V našem případě pomáhá v rámci skriptu, zabývajícího se registrací. V těchto databázích vytváříme tabulky pomocí prostředí phpMyAdmin viz kapitola 5.4.2. Poté, co se uživatel registruje a zvolí si registrační údaje, jsou údaje zaslány do relační databáze, ve které se uloží, a MySQL je uchovává a pracuje s nimi v prostředí phpMyAdmin.

5.4.2. phpMyAdmin

Jedná se o aplikaci napsanou v jazyce PHP. Tato aplikace je prostředníkem mezi jazykem PHP a databází MySQL. Představuje výhodné prostředí pro pracování s MySQL databázemi. Většinou je nabízena v balíku s pronájmem webhostingu.

Na obr. 6 můžeme vidět stavbu jednotlivých instancí, které nám dopomáhají k zaregistrování našeho uživatele na webové stránky. Pro práci jsou nejdůležitější dvě instance, a to AUTOR a HESLA. Do instance AUTOR se vkládají uživatelská jména, která si uživatel sám zvolí. Tento údaj se ukládá do různých datových textových řetězců, které nám databáze nabízí:

CHAR – Do tohoto typu se ukládají textové řetězce, které mají pevnou délku v rozsahu <1; 2000>. Můžeme zde ukládat znaky v tabulce ASCII, a to jak pro malá a velké písmena, tak číslice.

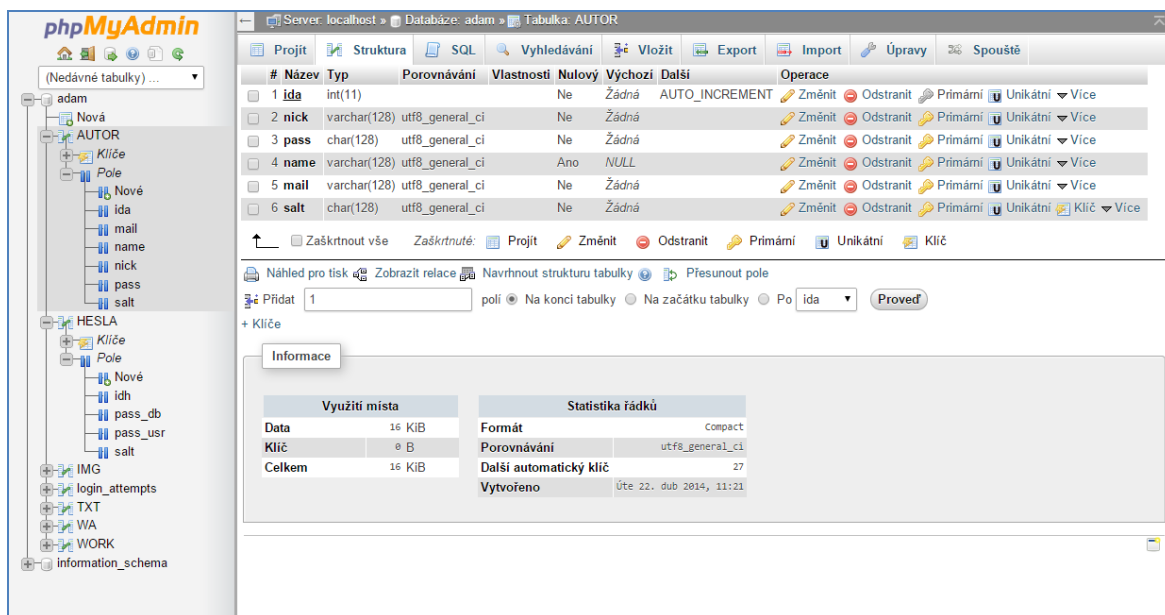
VARCHAR (varying) – Tento typ se liší oproti CHARU v tom, že se ukládá textový řetězec v proměnné délce. Jeho charakteristikou je tedy, reagování na zadanou šířku hodnot. Rozsah je větší, v rozmezí <1;4000>.

INT – Jedná se o jednodušší a omezenější datový typ. Do textového řetězce se ukládají pouze celá čísla.

Instance AUTOR dále zobrazuje rovněž další prvky, které však již nejsou tak důležité. Za zmínku stojí ještě vyzdvihnul kódování UTF-8, které je nepsaným standardem pro tvorbu webových stránek.

Instance HESLA funguje na podobném principu. Ukládají se sem hesla k uživatelským přezdívám.

Obrázek 6: Znárodnění prostředí phpMyAdmin



Zdroj: Vlastní tvorba

5.4.3. Apache

Apache je nedílnou součástí v rámci tvorby dynamických webových aplikací. Jedná se o webový server, který doplňuje PHP jazyk a jejich spojením vzniká duo, které je vysoce používané a tvoří špičku v softwaru pro tvorbu webu. Využívá se ve většině webhostingů ke spolupráci s různými technologiemi. Jeho rozšíření na platformách jako je např. Linux, Microsoft Windows a v neposlední řadě Mac OS, a popularita je hlavním důvodem pro volbu této technologie pro ukázkou v praktické části práce.

5.4.4. Editor

Jde o pracovní prostředí, ve kterém tvoříme zdrojový kód celých webových stránek. Do tohoto prostředí jsou vkládány značky HTML. Námí zvolený editor umí číst kód HTML a vložené dynamické prvky v tomto případě pro PHP jazyk.

Pro tuto webovou aplikaci byl zvolen editor Sublime text 3. Jde o prostředí, které bylo pro práci na tomto příkladu dostačující, přehledné a jednoduché. Pozadí editoru je načernalé a zdrojový kód má světlé barvy. Tagy mají barvu růžovou, což umožňuje větší přehlednost, a vývojář jak neztrácí čas zbytečným hledáním značek. Celý editor si můžeme ukázat na obr. 7. Již na první pohled vidíme, že barvy jsou sladěné a růžová dodává zdrojovému kódu potřebnou přehlednost.

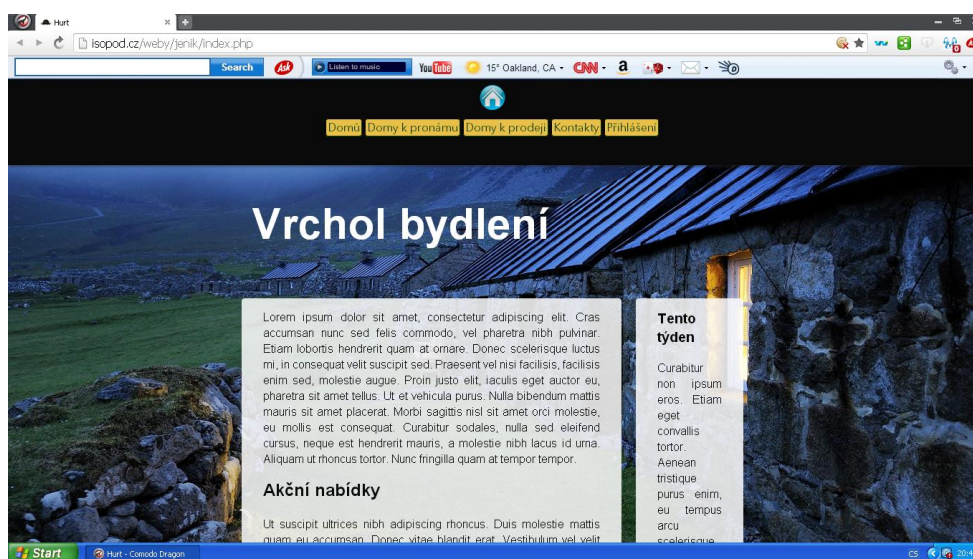
Obrázek 7: Ukázka editoru Sublime Text 3

```
4 <meta charset="utf-8">
5 <title>Hurt</title>
6 <link href="style.css" rel="stylesheet" type="text/css"><!--[if lt IE 9]>
7 <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
8 <![endif]-->
9 </head>
10
11
12 <?php
13 require 'php/functions.php';
14 sec_session_start();
15 ?>
16
17 <body>
18
19 <?php
20 include "layout/menu.php";
21 ?>
22
23 <div class="container">
24
25 <h1>Vrchol bydlení</h1>
26
27
28 <!-- SIDEBAR -->
29 <?php include "layout/asides/index_aside.php";?>
30 <article class="content">
31
```

Zdroj: Vlastní tvorba

Finální zobrazení dynamických webových stránek můžeme vidět na obr. 8. Stránky jsme vytvořili za pomoci technologií, které jsme si představili v teoretické části. Hlavní technologií použitou představovalo PHP a s ním spojený sekundární software. Vnořením různých skriptů a prvků PHP jazyka, využitím databází MySQL, prostředím phpMyAdmin a v neposlední řadě s pomocí serveru Apache byly vytvořeny a aplikovány dynamické prvky v oblasti registrace uživatelů, kde jsme vytvořili databázi hesel a uživatelských účtů. Byl vytvořen jednoduchý příklad dynamické webové stránky.

Obrázek 8: Celkový vzhled webových dynamických stránek




Zdroj: Vlastní tvorba

5.5. SEO


Za zkratkou SEO neboli Search Engine Optimization se skrývá sada nástrojů sloužící k optimalizaci stránky. Vytvořené webové dynamické stránky byly metodou SEO prověřeny a na obrázcích níže můžeme vidět výsledky testu, které lze využít pro další vývoj, či rozšíření testových stránek pro tvorbu zcela nových stránek.

Obrázek 9: SEO optimalizace


Zdrojový kód


 Zdrojový kód má optimální velikost. Příliš velká stránka zatěžuje vyhledávač stahováním přebytečných dat, a ten pak těžko určuje relevantní obsah. Navíc ho návštěvníci dlouho stahují.


Obsahová část


 Stránka obsahuje **právě jeden nadpis h1**.

Sémantika a přístupnost


 Stránka neobsahuje vnořené tabulky.

 Netextové elementy mají alternativní obsah.


 Na stránce je použito pouze správné sémantické zvýrazňování textu.

 Stránka obsahuje málo odstavců. Bylo by lepší text více strukturovat.

SEO Servis školení a tvorba webu

 Nevíte co výsledek testu znamená? Nevíte jak odstranit popisované chyby? Zkuste [školení SEO Servisu](#) nebo nám pošlete poptávku na [vytvoření webových stránek](#).

Ikonka pro vás

Pokud se s výsledkem testu chcete pochlubit, můžete na svůj web umístit odkaz na výsledek tohoto testu.
Zde máte k dispozici ikonku pro přidání: 

Přidejte ikonku na svůj web:

```
<a href="http://seo-servis.cz/source-zdrojovy-kod/11402120" title="SEO Servis, optimalizace pro vyhledavače."></a>
```

Zdroj: <http://seo-servis.cz/>

6. Zhodnocení výsledku

V rámci představení jednotlivých programovacích jazyků určených pro tvorbu webových prezentací byly prezentovány technologie, které bylo možné zvolit pro náš praktický příklad. Každý programovací jazyk má svá pro a proti. Každý má svá specifika, kterými chce nalákat uživatele. V rámci této práce byl vybrán programovací skriptovací jazyk PHP. Tato technologie je na špičce současného trhu používaných technologií, proto byla implementována a zahrnuta do praktické části práce. Rovněž nelze opomenout jednoduchost implementování těchto dynamických prvků do základního jazyka HTML.

Co se týká praktického příkladu webové aplikace, cílem bylo především osvětlení základní problematiky, a proto byl kladen důraz na jednoduchost a přehlednost. Vytvořené webové stránky v sobě zahrnují HTML jazyk, CSS (kaskádovité styly) a prvky pro tvorbu dynamických stránek v podobě PHP architektury. Použitý jazyk spolupracuje s databázovým systémem MySQL, v prostředí phpMyAdmin a s využitím serveru Apache. Ukázka byla vytvořena v Sublime Text 3, který je svojí dostupností, jednoduchostí a cenou vhodným editorem pro vývoj této webové aplikace.

Jelikož web byl primárně vytvořen za účelem o přiblížení tematiky tvorby webových aplikací s dynamickými prvky, je vhodné zhodnotit rovněž finanční náročnost projektu. Webová aplikace byla tvořena především s použitím softwaru fungujícího na Open source licenci. Jelikož PHP i server Apache spadají pod tuto licenci, finanční náročnost je v tomto směru nulová. phpMyAdmin je přidělen pořízením webhostingu a MySQL je zdarma dostupný pod licencí GPL (General Public License). Jediné vynaložené finanční prostředky souvisí s pronájmem webhostingu a domény. Tyto náklady jsou však díky poptávce a nabídce na trhu v této oblasti velmi příznivé. Zákazník může náklady stlačit k nule díky pronájmu webhostingu a domény zdarma. Tyto pronájmy však nenabízejí tak konformní služby pro klidné vyvíjení projektu. Tento projekt je vytvořen s využitím pronájmu obou dvou prvků v řádech stokorun. Finanční návratnost spočívá ve vlastní funkci webu, který prezentuje a nabízí sortiment za účelem zisku.

Cílem teoretické části práce bylo porovnat jednotlivé technologie pro tvorbu webu s dynamickými prvky a vybrat jednu z těchto technologií. Cíl této části bakalářské práce byl splněn.

Praktický příklad je jednoduchý, graficky méně náročný a struktura webových stránek je srozumitelná, aby byly čtenářům co nejvíce přiblíženy pro čtenáře principy a základy tvorby webových aplikací. Tento cíl bakalářské práce byl splněn.

7. Závěr

Po přiblížení problematiky spojené s oblastí internetu byla představena internetová služba World Wide Web. Služba představuje jeden z nejdůležitějších stavebních kamenů v oblasti IT a webových stránek. Před představením jednotlivých skriptovacích jazyků se autor podrobně seznámil se základní strukturou a fungováním značkovacího jazyka HTML. Skriptovací jazyk byl obecně charakterizován a jednotlivé vybrané technologie byly detailně přiblíženy a porovnány jejich výhody a nevýhody. Z těchto údajů zjišťujeme, že každý jazyk má své vlastnosti, kterými může zaujmout. Většina představených technologií se vyznačuje podobnými rysy, např. Open source nebo OOP, ale zjišťujeme i rozdílnost v podobě jednoduchosti (Ruby), podpory CGI skriptů (Perl) nebo pole působnosti, ve kterém má navrch jazyk Java. Dále byla popsána historie jejich vzniku a vývoje. Vzhledem k dobré dostupnosti, jednoduchosti a nízkým pořizovacím nákladům byla pro praktickou aplikaci zvolena technologie PHP. Je však třeba poznamenat, že i ostatní uvedený konkurenční software určený pro webové dynamické stránky dosahuje vysoké kvality. Pro vzájemné porovnání jednotlivých jazyků byl index TIOBE, který je všeobecně hojně využíván pro demonstraci uživatelských preferencí. V rámci hodnocení dle tohoto indexu bylo PHP pokořeno pouze Javou, která však zahrnuje i oblasti programování, ve kterých se PHP nevyužívá. Dále byl vytvořen demonstrativní příklad webové aplikace. Pomocí tohoto příkladu bylo možné představit strukturu a základní principy pro tvorbu webové aplikace. Autor použil implementaci prvků charakterizující technologii PHP, která patří do nepsaného standardu tvorby webových stránek. Na stránky je kladen důraz z hlediska jednoduchosti a přehlednosti, s méně náročnou grafickou úpravou.

Projekt může sloužit jako odrazový můstek k rozsáhlejšímu a složitějšímu webovým aplikacím. Vytvořený web může být rozšiřitelný téměř neomezenými prvky i technologiemi, které jsou dostupné na trhu. Může být prvním krokem vstupu do světa velkých firem a pozváním do současného i budoucího IT prostředí, které v sobě uchovává obrovskou moc v podobě IT technologií, jež budou udávat trend každodenního života i v následujících letech. Bakalářská práce může rovněž posloužit jako cenný zdroj informací vývojářům i jiným uživatelům, kteří by se rádi blíže seznámili nebo již zvažují využití některého ze skriptovacích programovacích jazyků pro svou práci, či zájmům.

8. Zdroje

8.1. Tištěné publikované dokumenty

DAŘENA, František. *Myslíme v jazyku Perl*. 1 vyd. Praha: Grada Publishing, a.s., 2005. ISBN 80-247-1147-8

FULTON, Hal. *Ruby, kompendium znalostí pro začátečníky i profesionály*. 1 vyd. Brno: ZONER software, s.r.o., 2009. ISBN 978-80-7413-018-2

GILMORE, W. Jason. *Velká kniha PHP a MYSQL 5*. 1 vyd. Brno: ZONER software, s.r.o., 2007. ISBN 80-86815-53-6

HARMS, Daryl, MCDONALD Kenneth. *Začínáme programovat v jazyce Python*. 1 vyd. Brno: Computer Press, 2006. ISBN 80-72226-799-X

HLAVENKA, Jiří. *Vytváříme WWW stránky a spravujeme moderní web site*. 6 vyd. Praha: Computer Press, 2002. ISBN 80-7226-748-5

LUTZ, Mark, DAVID, Ascher. *Naučte se Python*. 1 vyd. Praha: Grada Publishing, a.s., 2003. ISBN 80-247-0367-X

MACDONALD, Matthew, SZPUSZTA, Mario, FREEMAN, Adam. *ASP.NET 4 a C#sharp 2010*. 1 vyd. Brno: ZONER software, s.r.o., 2011. ISBN 978-80-7413-131-8

MACDONALD, Matthew, SZPUSZTA, Mario. *ASP.NET 2.0 a C# - tvorba dynamických stránek profesionálně*. 1 vyd. Brno: ZONER software, s.r.o., 2006. ISBN 80-86815-38-2

PERRY, J. Paul. *Java tvorba dokonalých WWW stránek*. 1 vyd. Praha: Grada Publishing, spol. s.r.o., 1996. ISBN 80-7169-415-0

PÍSEK, Slavoj. *HTML - tvorba jednoduchých internetových stránek*. 1 vyd. Praha: Grada Publishing, spol. s.r.o., 2001. ISBN 80-247-0094-8

PROCHÁZKA, David. *PHP 6 začínáme programovat*. 1 vyd. Praha: Grada publishing, a.s., 2012. ISBN 978-80-247-3899-4

SATRAPA, Pavel. *Perl pro zelenáče*. 2 vyd. Praha: Neocortex s.r.o., 2001. ISBN 80-86330-02-8

SCHAFER, M Stevan. *HTML, XHTML a CSS*. 1 vyd. Praha: Grada Publishing, a.s., 2009. ISBN 978-80-247-2850-6

ULLMAN, Larry. *PHP a MySQL* 1 vyd. Brno: Computer Press, 2004. ISBN 80-251-0063-4

8.2. Elektronické zdroje

KOSEK, Jiří, *Historie a vývoj HTML* [on-line]. Vystaveno 2013 [cit. 2015-2-27].

Dostupný z WWW: <http://htmlguru.cz/uvod-historie.html>

MANAGEMENT MANIA, *Architektura klient – server* [on-line]. Vystaveno 22. 07. 2013 [cit. 2015-2-27]. Dostupný z WWW: <https://managementmania.com/cs/architektura-klient-server>

NOVOTNÝ, Luděk, *Historie a vývoj jazyka Java* [on-line]. Vystaveno 2003 [cit. 2015-2-27]. Dostupný z WWW: <http://www.fi.muni.cz/usr/jkucera/pv109/2003p/xnovotn8.htm>

SEO SERVIS, *Analýza zdrojového kódu stránek* [on-line]. [cit. 2015-2-27]. Dostupný z WWW: <http://seo-servis.cz/>

STRACH, Jiří, *Základní software* [on-line]. [cit. 2015-2-27]. Dostupný z WWW: <http://boss.ped.muni.cz/vyuka/material/puvodni/software.htm>

TIOBE SOFTWARE, *TIOBE index for February* [on-line]. Vystaveno 19. 2. 2015 [cit. 2015-2-27]. Dostupný z WWW: <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>