

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Diplomová práce

Webové skriptování na straně serveru

Petr Mendl

© 2013 ČZU v Praze

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Webové skriptování na straně serveru" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 16.11. 2013

Poděkování

Chtěl bych touto cestou poděkovat Ing. Václavu Lohrovi za jeho odborný dohled a pomoc při zpracování mé diplomové práce.

Webové skriptování na straně serveru

Server side web scripting

Souhrn

Tato diplomová práce popisuje nejčastější skriptovací jazyky využívané při tvorbě webových stránek se zaměřením na konfiguraci a instalaci freewarového prostředí jazyka PHP. Práce je rozčleněna do několika částí, kde v úvodní části je uveden historický vývoj statických HTML stránek s postupným vývojem na dynamické stránkování na straně serveru pomocí PHP. Dále se zabývá popisem nejčastěji používaných skriptovacích jazyků. Hlavní cíl byl zaměřen na vlastní instalaci a konfiguraci skriptovacího jazyka PHP s využitím webového serveru Apache a databázového prostředí MySQL pro tvorbu webových stránek pro společenství vlastníků jednotek našeho domu. V závěru práce je ekonomické vyhodnocení WWW stránek při poskytování služeb web hostingu a porovnání PHP s komerčními produkty od firmy Microsoft..

Klíčová slova:

PHP	HTTP
HTML	APACHE
SCRIPT	FORM
JAVA	CCS
MYSQL	XHTML

Summary

This thesis describes the most common scripting languages used to create websites focused on configuring and installing freeware PHP environment. The thesis is divided into several parts, where the first part deals with the historical development of static HTML pages with the gradual development of the dynamic paging on the server side using PHP. It also deals with a description of the most commonly used scripting language.

The main objective focused on custom installation and configuration of PHP using the Apache web server and MySQL database environment for creating websites for community owners of our house.

The final part of the thesis provides the economic evaluation of websites in providing web hosting and PHP compared with Microsoft commercial products.

Klíčová slova:

PHP	HTTP
HTML	APACHE
SCRIPT	FORM
JAVA	CCS
MYSQL	XHTML

OBSAH

1	ÚVOD	7
2	CÍL PRÁCE A METODIKA.....	11
2.1	CÍL PRÁCE	11
2.2	METODIKA PRÁCE.....	11
3	SKRIPTOVACÍ JAZYKY, DATABÁZE , WEB SERVERY	13
3.1	JAZYK PHP.....	13
3.2.	ASP	20
3.3.	JAVASCRIPT	21
3.4.	MYSQL	21
3.5.	APACHE	22
3.6.	PERL	23
3.7.	WEBOVÉ EDITORY POSKYTUJÍCÍ PODPORU PHP.....	24
4	PRAKTICKÁ ČÁST	25
4.1	POPIS ZADÁNÍ REALIZAČNÍHO PROJEKTU	25
4.2	TECHNICKÁ SPECIFIKACE.....	26
4.3	SOFTWARE	28
4.4	INSTALACE WAMP SERVERU	30
4.5	STRUKTURA A SYNTAXE JAZYKA PHP.....	34
4.6	VYTVORENÍ MYSQL DATABÁZE	34
4.7	PSANÍ PHP, JAVASCRIPT A HTML KÓDU.	36
4.8	ZÁKLADNÍ PHP SKRIPTY STRÁNEK SVJ.....	37
4.8.1	<i>Skript PHP - přístup do systému</i>	<i>38</i>
4.8.2	<i>Skript PHP - registrace.....</i>	<i>40</i>
4.8.3	<i>Skript PHP - změna hesla.....</i>	<i>42</i>
4.8.4	<i>Skript PHP - připomínky a náměty</i>	<i>44</i>
4.8.5	<i>Skript PHP - výpis všech nájemníků(vlastníků) jednotek</i>	<i>47</i>
4.8.6	<i>Skript PHP - zápisy schůzí SVJ.....</i>	<i>50</i>
4.8.7	<i>Skript PHP - návštěvníci stránek.....</i>	<i>52</i>
4.8.8	<i>Skript PHP - nástěnka.....</i>	<i>54</i>
4.8.9	<i>Skript PHP - admin</i>	<i>56</i>
4.9	STATICKE STRÁNKY	59
4.10	ZABEZPEČENÍ WEBOVÝCH APLIKACÍ	60
4.10.1	<i>Cross site scripting.....</i>	<i>60</i>
4.10.2	<i>SQL injection.....</i>	<i>60</i>
4.10.3	<i>Directory Traversal</i>	<i>60</i>
4.11	EKONOMICKÉ VÝSLEDKY	61
5	ZHODNOCENÍ VÝSLEDKŮ	65
6	ZÁVĚR.....	66
7	SEZNAM POUŽITÝCH ZDROJŮ	67
8	PŘÍLOHY	69
9	ZKRATKY A VYSVĚTLIVKY.....	72

1 Úvod

Přechod od statických stránek k dynamickému skriptování na straně serveru prošel dlouhou cestu. V roce 1991 přišel s první definicí jazyka HTML Tim Berners-Lee, jako součástí projektu WWW. Tento projekt měl umožnit vědcům, kteří se zabývali fyzikou vysokých energií, lepší komunikaci a sdílení výsledků výzkumu po celém světě. Celý projekt vznikl v CERNu (Centre Européenne pour la Recherche Nucléaire, Evropské centrum jaderného výzkumu). CERN je velkou institucí, kde měl tento projekt vyřešit problémy se sdílením informací. Tato verze HTML byla popsána v dokumentu HTML Tags. Umožňovala text rozčlenit do několika logických úrovní, zařadit do textu odkazy a obrázky a použít několik druhů zvýraznění textu.

Tim Berners-Lee při návrhu HTML nepředpokládal, že by autoři webových stránek museli tento jazyk znát. První verze WWW softwaru byla napsána pro operační systém NextStep a obsahovala jak prohlížeč, tak i integrovaný editor webových stránek. V roce 1991 byl veřejně zpřístupněn software vyvinutý v CERNu, včetně specifikací jazyka HTML. [1]

Producenti různých prohlížečů doplňovali HTML o nové prvky v důsledku vzrůstu požadavků uživatelů na WWW. Aby byla zachována kompatibilita mezi jednotlivými modifikacemi HTML, vytvořil Berners-Lee pod hlavičkou IETF (Internet Engineering Task Force) návrh standardu HTML 2.0, který zahrnoval všechny, v té době běžně používané, prvky HTML. Verze HTML 2.0 má dvě úrovně. První z nich (Level 1) rozšiřuje předchozí verzi HTML jen velmi málo. Level 2 navíc definuje práci s formuláři. Specifikaci HTML 2.0 nalezneme v RFC dokumentu číslo 1866.

Další rozšíření jazyka známá jako HTML+ zahrnují převážně vytváření tabulek a matematických vzorců. Také se zde objevují prvky, které umožňují lépe kontrolovat výsledný vzhled textu – lepší obtékání obrázků textem a styly dokumentů. Na jaře v roce 1995 vznikl návrh standardu HTML 3.0. Jeho autorem byl Dave Raggett z laboratoří Hewlett-Packard, který formuloval a vytvořil definici HTML+ jako DTD (Document Type Definition) v jazyce SGML.

Někter prvky HTML 3.0, jako např. tabulky, podporovaly novější verze prohlížečů Mosaic a Netscape. Kompletní podporu pro většinu rysů HTML 3.0 nabízel pouze experimentální prohlížeč Arena, který se však nikdy příliš nerozšířil. [2]

Začátkem roku 1996 již bylo jasné, že HTML 3.0 bylo tak velkým skokem vpřed, že se nenašel nikdo, kdo by dokázal implementovat prohlížeč s jeho podporou. Konsorcium W3C, jehož členy jsou významné softwarové firmy, řídilo vývoj standardů Webu. Členové W3C se tedy shodli na vlastnostech, o které rozšíří HTML 2.0, a vytvořili tak HTML 3.2. HTML 3.2 však zdaleka neobsahuje vše z HTML 3.0. Z verze 3.0 zůstaly jen zjednodušené tabulky. Ostatní nové prvky HTML 3.2 jsou jen jakousi směsí, kterou v té době podporovaly nejnovější prohlížeče. Stejně tomu bylo i při vzniku verze 2.0, kdy došlo ke sjednocení jazyka na základě průniku možností těch nejrozšířenějších prohlížečů. Kromě tabulek přibyla ve verzi 3.2 především lepší kontrola formátování včetně mnohem volnějšího výběru použitých druhů písma, což bylo ústupkem požadavkům na graficky perfektně vypadající stránky. Další podstatné rozšíření se týkalo podpory Java appletů. HTML 3.2 neslo kódové označení Wilbur a v lednu 1997 doporučilo konsorcium W3C, že by jej měli všichni používat, aby byla ve Webu zajištěna kompatibilita.

Většina prohlížečů však již od roku 1996/7 nabízela další rozšíření nad rámec HTML 3.2. Jejich používání má dvě protikladné strany – na jednu stranu nebyla tato rozšíření standardizována, a proto jim některé prohlížeče nerozuměly. Na druhou stranu jsou velmi užitečná, protože svým používáním a rozšířením urychlila jejich zařazení do standardu. Mezi taková rozšíření patří především rámy (frames) a klientské skriptovací jazyky (např. Javascript).

Na jaře roku 1997 zveřejnilo W3C další plány na rozšíření HTML pod kódovým názvem Cougar. Cougar v sobě zahrnuje HTML 3.2 společně s běžně používanými konstrukcemi, jako jsou rámy, skripty a obecné vkládání objektů. Dalšími novinkami byla podpora vícejazyčných dokumentů. Návrh HTML 4.0 byl představen W3C začátkem července 1997. Vznikl drobnými úpravami Cougaru a vytvořením jediného komplexního dokumentu popisujícího návrh standardu. V prosinci 1997 pak bylo HTML 4.0 přijato jako standard W3C. Od té doby se na dlouhou dobu samotný vývoj jazyka HTML zastavil. Pouze v prosinci 1999 byl vydán nový standard HTML 4.01, který upravoval některé drobné chyby v předchozí specifikaci.

W3C přišlo s dalším standardem brzy po vydání specifikace HTML 4.0, a to s jazykem XML (eXtensible Markup Language). Ten se od svého vydání v roce 1998 stal ve světě informačních technologií nejčastěji používaným formátem pro výměnu a ukládání dat. XML ovlivnilo i další vývoj jazyka HTML. Na začátku roku 2000 byla vydána specifikace jazyka XHTML 1.0, která měla svoji skladbu odvozenou od XML, nikoli od SGML jako HTML. V podstatě šlo pouze o drobnou změnu. Seznam dostupných značek a jejich význam zůstal stejný, jen se nepatrně změnila skladba. Aby se tato verze, založená na XML, odlišila od předchozích verzí HTML, bylo před název přidáno písmeno „X“.

Na začátku tisíciletí se XHTML považovalo za nejlepší technologii pro klasický web. Vznikly i speciální zjednodušené verze XHTML, které umožňovaly jen některé funkce jazyka XHTML 1.0 a byly určeny pro specifické použití. Například XHTML Basic pro mobilní zařízení, nebo XHTML Print pro jednoduché tiskové výstupy. K popisu modularizace celého jazyka a vytváření vlastních odvozenin vzniklo XHTML 1.1, které do jazyka nepřidávalo žádnou funkčnost, ale umožňovalo snáze vytvářet odnože jazyka XHTML.

Později se ovšem ukázalo, že samotné XHTML nebylo pro další vývoj webu tou nejlepší cestou. Jazyky XHTML 1.0 a 1.1 nepřidaly oproti HTML 4.01 v podstatě žádnou novou funkčnost, která by byla důvodem přechodu od HTML k XHTML. Bohužel se ukázalo, že i prohlížeč Internet Explorer neuměl bezchybně využívat XHTML stránky.

Na další verzi jazyka XHTML 2.0 začala pracovat pracovní skupina ve W3C, snažila se přijít s dalšími zajímavostmi ovšem na úkor porušení kompatibility s předchozími verzemi jazyka HTML a XHTML. Někteří výrobci prohlížečů se s tímto krokem neztotožňovali, protože se přikláněli k postupnému vylepšování stávajících funkčních technologií – tedy HTML 4.01 a XHTML 1.0. V roce 2004 byla založena pracovní skupina WHATWG (The Web Hypertext Application Technology Working Group), která se začala zabývat specifikací platformy pro webové aplikace běžící v prohlížeči. Kromě rozšíření jazyka HTML specifikace zahrnovala i definice důležitých rozhraní pro využití ve skriptovacím jazyku Javascript.

Pokračovat na vývoji HTML dvěma směry bylo samozřejmě nežádoucí. Po dlouhých a složitých diskusích a jednáních se nakonec také W3C přiklonilo k názoru, že vývoj XHTML 2 nikam nepovede, protože výrobci prohlížečů jej ignorují. V roce 2007 se W3C

a WHATWG spojily a pracovní skupina HTML začala pracovat na HTML5, jehož základem byla specifikace vytvořená ve WHATWG.

HTML5 vychází z HTML 4.01 a rozšiřuje jeho funkce o mnoho dalších, kvůli kterým je výhodné tuto nejnovější verzi jazyka používat. Samozřejmě to neomezuje používání XML a XHTML pro zápis stránek.

V tuto chvíli je stále HTML5 ve vývoji a jeho dokončení se předpokládá nejdříve kolem roku 2020. Je třeba ještě mnoho částí dovést k dokonalosti, i když některé jsou již stabilní a jsou podporovány většinou prohlížečů. Pouze čas ukáže, zda se veškerá zdokonalení projeví i v konečné verzi. [3]

V roce 1994 napsal základní část Common Gateway Interface (CGI) v programovacím jazyku C dánský programátor Rasmus Lerdorf. Byla vytvořena pro domácí stránky (Personal Home Page), aby mohly být nahrazeny některé skripty v Perlu, které měly být využívány pro údržbu domovské stránky. Zde vznikl název **PHP**.

Zajišťovalo zobrazení obsahu a zaznamenávalo návštěvnost stránek. Rasmus Lerdorf tento binární kód sloučil s jiným programem Form Interpreter, jehož byl také autorem. Po spojení vznikla kombinace PHP/FI, která tímto spojením rozšířila svoji funkčnost. Umožnila zvětšit implementaci pro programovací jazyk C, další předností byla možná komunikace s databázemi a tak mohly vznikat první jednoduché webové aplikace. PHP, označené jako PHP verze 2, se veřejně poprvé objevilo 8. června 1995 a mělo základní funkčnost jako má dnešní PHP. Vycházelo ze stylu Perlu, a to ve zpracování formulářů, v podobnosti proměnných a možnosti vkládat HTML kód, ale bylo jednodušší, omezenější a méně konzistentní. [4]

2 Cíl práce a metodika

Výsledkem této práce bude ucelený návrh konfigurace a instalace PHP prostředí aplikovatelný v praxi se zaměřením na rešeršní zpracování základního popisu jednotlivých skriptovacích jazyků, které se využívají při vytváření dynamických HTML stránek a podrobněji zaměřen freewarový scriptovací jazkyk PHP. Dílčím cílem je popis základních možností nastavení při instalaci a konfiguraci PHP, včetně vytvoření reálných webových stránek pro společenství vlastníků jednotek našeho domu(SVJ).

2.1 Cíl práce

Tato práce bude hlavně zaměřena na dynamické scritpování na straně serveru, ale autor se zmíní i o scriptovacích jazycích, které jsou vykonávány i na straně klienta (pracovní stanice). Cílem je popsat ucelený postup tvorby dynamických webových stránek od pořízení hardwaru, popisu instalace nutných softwarových balíčků včetně nasazení a importu dat do komerčního prostředí vybraného poskytovatele internetu. V další části bude popsáno ekonomické zhodnocení a časová náročnost realizovaného projektu.

2.2 Metodika práce

Zvolen byl reálný postup instalace nasazení webového serveru s využitím možností dynamického skriptování pomocí jazyka PHP, webového serveru Apache a propojením na databázový systém MySQL včetně doporučeného vývojového prostředí pro programování dynamického skriptování na straně serveru.

Pro analýzu případové studie v této práci bude nutné získat technická data, zejména seznam a počet vlastníků (nájemců) daného společenství vlastníků jednotek. Podstatnou informací pro poskytování informovanosti vlastníků jednotek jsou i vlastní ekonomické a účetní data o vývoji a stavu ekonomiky daného SVJ. Tato data později poslouží pro syntézu řešení technické specifikace a návrhu webových stránek s poskytováním přehlednosti a účelnosti pro vlastníky jednotek SVJ.

Na základě sebraných dat bude provedena reálná instalace, webového serveru využívající PHP prostředí včetně konfigurace a instalace MySQL serveru, který poslouží právě pro načítání dynamicky orientovaných webových stránek. Autor se též zaměří na řádnou instalaci potřebných balíčků, které jsou zapotřebí pro případné napojení dalších databázových systémů jako jsou například databáze ORACLE či MS SQL. V této práci budou vysvětleny základní příkazy jazyka PHP, které jsou důležité pro napojení do databázového systému MySQL, ze které se budou prezentovat informace o jednotlivých vlastnících daných bytových jednotek a jejich ekonomická situace, jako je například případná zadluženost dané bytové jednotky. Veškerá základní konfigurace bude probíhat v laboratorních podmínkách, na pracovním serveru, který má základní technické parametry, které jsou doporučené a vyhovující pro poskytování webových aplikací.

3 Skriptovací jazyky, databáze , web servery

3.1 Jazyk PHP

PHP je skriptovací jazyk pro tvorbu dynamického webu, poprvé se objevuje počátkem roku 1994. Autorem je Rasmus Lerdorf, který vychází z Perlu a cílem je vytvořit jednoduchý systém, který umožní přístup k domácím stránkám. Systém s perlovským kódem zatěžuje server, a proto je přepsán do jazyka C. Soubor těchto skriptů byl vydán pod názvem "Personal Home Page Tools", zkráceně PHP, později ještě "Personal Home Page Construction Kit". Netrvalo dlouho a po spojení systému PHP s programem „Form Interpreter“, kde byl autorem také Rasmus Lerdorf, vznikl PHP/FI 2.0. Tento systém se stal velmi rozšířený a světově proslulý. Vývoj neustal a koncem roku 1998 se objevila další verze PHP 3.0, ta byla lépe vybavenější a mnohem rychlejší než verze 2. Obě verze byly přístupné pod operačními systémy Windows. Počet webů v současnosti, které využívají PHP je cca 150 000. Současná verze PHP 4.0 rychlejší jádro Zend a mnoho nových funkcí. Podle údajů z dubna 2010 je PHP využíváno více jak 15 000 000 doménami a je to bezkonkurenčně nejčastěji používaný modul webového serveru Apache. Zkratka PHP by měla být používána jako "PHP: Hypertext preprocessor", česky přibližně "před zpracovávач hypertextu". V současnosti je možné využít Release Candidate 2 verze PHP 5. Je zde kompletně přepsaná podpora XML, nové objektové rozhraní pro práci s databází MySQL, jádro Zend II a mnoho dalšího. Je to příklad toho, že systém je neustále vyvíjen, zdokonalován a vybavován novými technologiemi. Mezi hlavní důvody, proč využívat dynamické stránky pomocí jazyku PHP jsou tyto důvody:

- PHP je relativně jednoduché na pochopení
- PHP má syntaxi velmi podobnou jazyku C a je tedy většině vývojářů dost blízký
- PHP podporuje širokou řadu souvisejících technologií, formátů a standardů
- je to otevřený projekt s rozsáhlou podporou komunity
- dají se najít kvanta již hotového kódu k okamžitému použití nebo funkční PHP aplikace. Podstatná část z hotového kódu je šířena pod nějakou svobodnou licencí a dá se použít ve vlastních projektech
- PHP si dobře rozumí s webovým serverem Apache

- PHP snadno komunikuje s databázemi, jako je MySQL, PostgreSQL a řada dalších
- PHP je multiplatformní a lze jej provozovat s většinou webových serverů a na většině dnes existujících operačních systémů
- PHP podporuje mnoho existujících poskytovatelů webhostingových služeb

Nevýhody PHP

PHP nemá podstatné nevýhody, ale je zapotřebí s některými počítat a to:

- PHP je interpretovaný, ne kompilovaný jazyk
- kdokoli má přímý přístup k serveru, může nahlédnout do našeho PHP skriptu
- protože je PHP aktivně vyvíjen, v budoucích verzích jazyka se mohou některé funkce změnit nebo se mohou chovat jinak než dosud. [5]

V PHP můžeme napsat:

- internetové obchody
- podnikové informační systémy (ať už intranetové nebo internetové)
- diskusní fóra
- redakční systémy
- firemní prezentace
- dynamické osobní stránky
- vyhledávače a katalogy
- drobnosti typu počítačidla, ankety a mnoho dalších

Níže je uveden historický vývoj jednotlivých verzí jazyka PHP

Řada	Verze	Datum vydání	Poznámka
1.x	1.0.0	8. června 1995	Oficiální název „Personal Home Page Tools (PHP Tools)“. Poprvé byl použit název „PHP“.
2.x	2.0.0	16. dubna 1996	Autor jej považoval za "nejrychlejší a nejjednodušší nástroj" pro tvorbu dynamických webových stránek.
3.x	3.0.0	6. června 1998	Tým vývojářů se z původní jedné osoby rozšířil na více osob. Zeev Suraski a Andi Gutmans přepsali celý základ pro tuto verzi.
4.x	4.0.0	22. května 2000	Přidán pokročilý dvoustupňový systém parse/execute (analyzovat/vykonat) syntaktické analýzy tagu - Zend engine.
	4.1.2000	10. prosince 2001	Byly představeny 'superglobální proměnné' (\$_GET, \$_POST, \$_SESSION, etc.)
	4.2.2000	22. dubna 2002	zakázání register_globals ve výchozím nastavení. Data přijatá prostřednictvím sítě nejsou přímo vkládány do názvů globálních proměnných, uzavírá se tím možnost využití
	4.3.2000	27. prosince 2002	Představeno CLI, jako doplněk CGI.
	4.4.2000	11. srpna 2005	Dodány stránky nápovědy pro skripty phpize a php-config.
	4.4.2008	3. ledna 2008	Oprava některých bezpečnostních prvků a oprava chyb. Měla to být poslední update verze PHP 4. Bezpečnostní updaty budou jen do 8. srpna 2008, pokud to bude nezbytné.
	4.4.2009	7. srpna 2008	Oprava některých bezpečnostních prvků a oprava chyb. Poslední vydání verze ze série PHP 4.4.
	5.0.0	13. července 2004	Zend Engine II s novým objektovým modelováním.
5.x	5.1.2000	24. listopadu 2005	Zlepšení výkonu zavedením kompilátoru proměnných v přepracovaném Engine PHP.
	5.2.2000	2. listopadu 2006	Povolen filtr přípon ve výchozím nastavení.
	5.2.2006	1. května 2008	oprava bezpečnostních děr a chyb
	5.2.2008	8. prosince 2008	oprava chyb
	5.2.2014	22. července 2010	oprava chyb
	5.2.2015	10. prosince 2010	oprava chyb, poslední verze řady 5.2.x
	5.3.2000	30. června 2009	jmenné prostory, oprava chyb, změny ve funkcích a doplňcích
	5.3.2001	19. listopadu 2009	oprava chyb
	5.3.2002	4. března 2010	oprava chyb, přidání podpory SHA-256 a SHA-512
	5.3.2003	22. července 2010	oprava chyb, FPM SAPI
	5.3.2004	10. prosince 2010	oprava chyb, vylepšení FPM SAPI
	5.3.2005	6. ledna 2011	oprava chyb
	5.3.2006	17. března 2011	oprava více než 60 chyb
	5.3.2007	18. srpna 2011	-
5.3.2008	23. srpna 2011	opravuje chybu s MD5	
5.4.2000	1. března 2012	traits, dereference polí, odstranění některých zastaralých funkcí a direktiv	

Princip PHP

V prvopočátcích internetu byly internetové stránky statické. Znamená to, že stránka byla vytvořena, odeslána do prohlížeče a tak se i zobrazila. Vývoj se ovšem nezastavil, bylo potřeba přijít s novými technologiemi, které měly stránky rozpohybovat. Tyto technologie se rozdělily do dvou skupin, na "klientské" a "serverové".

"Klientské" technologie fungují tak, že se spolu s HTML stránkou je prohlížeči odeslán i část programového kódu a ten je spuštěn na "cílovém" počítači stiskem tlačítka, najetím myši na odkaz, nebo otevřením okna prohlížeče a podobně. Klientský kód spouští prohlížeč, což může být nevýhodou, pokud nezná programovací jazyk, ve kterém je kód napsán. Java skript je známým příkladem technologie, která běží na straně klienta.

"Serverové" technologie jsou založeny na jiném principu. Server nejprve sestaví webovou stránku, kterou prohlížeč požaduje a pak ji odešle. Zde se setkáme s flexibilitou těchto technologií, protože sestaví pokaždé jinou stránku v závislosti na tom, co přesně prohlížeč požaduje.

PHP je technologie fungující na serveru. Klasický PHP skript obsahuje část normálního HTML kódu, a část programového kódu. Když webový server obdrží požadavek na zpracování takového skriptu, použije:

- části HTML kódu tak, jak jsou
- části PHP programového kódu provede
- výsledek zkombinuje a odešle prohlížeči

Server dokáže provést jednu i více operací. Výsledek se pak pošle do prohlížeče jako obyčejná HTML stránka.

Příklad 1: Kód je obyčejná HTML stránka a po vyžádání prohlížečem je v něm vidět jen řádek mezi značkami.

```
<HTML>
Já jsem obyčejná stránka. Nic neumím.
</HTML>
```


Příklad 2: Kód je PHP skript. Skládá se z HTML kódu a z PHP kódu. Příkaz echo vypisuje text na standardní výstup a příkaz Date vrací aktuální čas:

```
<HTML>
Toto je úvodní stránka. Aktuální čas je <?echo Date("G:i")?>
</HTML>
```

Stránka nám vypíše aktuální čas na serveru v době, kdy byla generována. To proto, že kód mezi značkami <? a ?> se na serveru nejprve provede a teprve pak se výsledek odešle prohlížeči. Do prohlížeče dorazí něco jako:

```
<HTML>
Toto je úvodní stránka. Aktuální čas je 21:08
</HTML>
```

Prohlížeč nemá představu, co všechno se na serveru proběhlo než, mu byl výstup odeslán, vidí pouze vlastní výsledek. Na serveru může probíhat mnoho věcí (funkcí) – matematické výpočty, přístupy k databázím, formátování, operace s řetězci apod.

Aby bylo možné PHP skript opravdu spustit, musíte ho "prohnat" serverem. Když jej otevřeme v textovém editoru nebo si do něj nahlédneme jako soubor na disku, neuvidíme výsledek, ale zdrojový kód skriptu. To se nám může zaměnit zejména tehdy, když máme PHP skript na vlastním počítači a používáme jej k prohlížení souborů na disku, je to stejný nástroj, jako na prohlížení obsahu sítě internet. PHP soubor proženeme serverem tak, že se na něj zeptáme prohlížeče pomocí adresy „URL“. Například pomocí **http://localhost/skript.php** je možné nařídít serveru, aby skript zpracoval a poslal nám výsledek, zatímco pomocí **file://var/www/html/skript.php** zobrazíme pouze obsah souboru. [6]

PHP skripty nejsou závislé na použitém prohlížeči, protože ten s nimi nepřijde do styku. Veškerou práci provede server a do prohlížeče se prezentuje pouze prosté HTML.

Části PHP kódu se do skriptů vkládají pomocí speciálních značek. V úvahu mohou přicházet následující možnosti kombinace HTML a PHP kódu v jednom souboru:

1. Vlastní soubor může být prostý PHP kód. V tom případě začíná a končí znaky `<? a ?>`
2. V souboru může být kód PHP pouze na jednom definovaném místě.
3. V souboru může být více než jedno místo obsahující PHP kód
4. Soubor se sice může tvářit jako php skript, ale ve skutečnosti je to normální HTML soubor a PHP kód v něm vůbec není obsažen.

PHP soubory se označují nejčastěji příponou `php`, případně `php3`, `php4`, `php5`. Přípona je závislá na konfigurovaném webovém serveru. PHP je tvořen za účelem pro vytváření webových stránek, proto kód jazyka je možné začleňovat přímo do HTML kódů, jen je nutné jej oddělit. K rozdělení PHP kódu a HTML slouží PHP značky: (`<?php a >?`) Tyto značky interpretují, kde `php` kód začíná a kde končí. Všechn kód, který je vložen mezi těmito značkami je interpretován právě jako kód PHP. Lze použít i kratší syntaxe začátku a konce PHP `<?.. PHP kód...?>`, ale aby toto fungovalo zapotřebí to povolit v konfiguračním souboru jazyka `/short_tags/`. PHP využívá základních syntaxí jako například:

- Středník- patří mezi nejdůležitější části skriptu PHP, každou funkci , řádek, deklaraci je nutné oddělit tímto středníkem
- Komentáře- užitečné věci ve skriptu. Je to text , který je zobrazen pouze ve skriptu(zdrojovém kódu), ale je určen pouze autorovi, aby se lépe orientoval, komentář je možné využít jednořádkový `// jednořádkový komentář` nebo i více řádkový komentář:

`/*`

Více řádkový komentář

`*/`

Využívané styly

- XML styl

Tento styl je PHP asi nedoporučovanějším stylem, jeho dostupnost je garantovaná na všech sever. Je obtížné jej adminem vypnout a lze je využívat v souborech XML. Je ho syntaxe je:

<?php ...PHP kód ...?>

- Skriptový styl

Tento styl má nejbližší ke vkládání JavaScriptu nebo VBScriptu jeho syntaxe je:

```
<script type="text/php">... PHP kód ...</script>
```

- ASP styl

Tento styl je obdobý jako v ASP od společnosti Microsoft. Proto, aby v PHP tento styl fungoval je nutností toto nastavit v `asp_tags`. Jeho syntaxe je následující:

```
<%... PHP kód ... %>
```

Pro výpis textu se nejčastěji používá příkaz `echo()`, příkladem může být `echo ("Ahoj");`

Dle mého názoru má jazyk PHP i tzv. „zrádné znaky“ na které je zapotřebí si dát pozor a tyto znaky mohou ohrožovat chod skriptu, jedná se především o znaky `" ' \ $`. Před tyto znaky je nutností vložit zpětné lomítko

Proměnné

V PHP se bez proměnných neobjede téměř žádný skript. Začíná značkou `$` a je zapotřebí dodržovat syntaxi:

-začíná písmenem nebo podtržítkem

Neobsahuje mezery

Píše se bez diakritiky

Je case-sensitive (je rozdíl mezi velkými a malými písmeny)

Jednotlivé proměnné mají své datové typy

Integer tento typ uchovává celá kladná a záporná čísla

Double(float) uchovává desetinná čísla

String je to řetězec nebo znak či sada znaků

Boolean – jedná se o logickou hodnotu, která může nabývat `false` nebo `true`

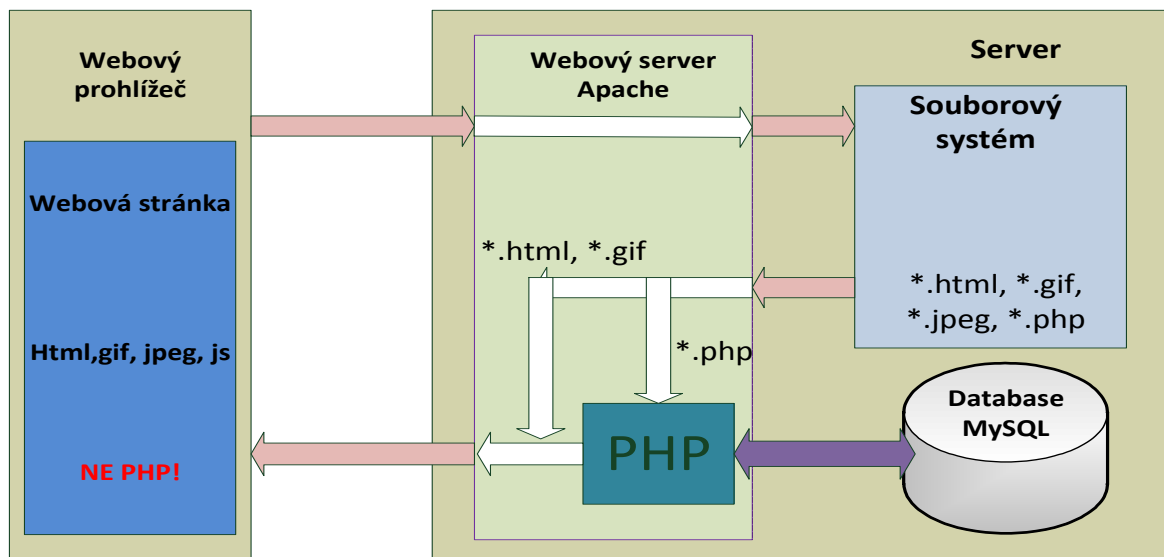
Příkladem výpisu proměnné může být:

```
$zvire="kočka";
```

`Echo ($zvire);` tato funkce vypíše obrazovku `zvire`.

V případě, že bychom chtěli použít souvislý text je možné použít:

`Echo ("toto zvíře je". $zvire);` na obrazovce se nám zobrazí: *Toto zvíře je kočka*



Obrázek 1 - bloková interpretace PHP

3.2. ASP

ASP (Microsoft Active Server Pages) je skriptovací prostředí pro servery, které je možné využít pro tvorbu a provozování dynamických, interaktivních webových aplikací. S využitím ASP, můžeme kombinovat HTML stránky, skripty, a COM komponenty a vytvářet tak interaktivní webové stránky nebo výkonné webové aplikace, které se snadno vyvíjejí i upravují. Na tento druh dynamického skriptování, který je také velmi využívány pro webové servery IEE. Autor se zaměřil na tento druh skriptování, z důvodů komerčního placené verze od firmy Microsoft. Nicméně se domnívá, že patří mezi velmi propracovaný skriptovací nástroj.[8]

3.3. JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, autorem tohoto jazyka je Brendan Eich, který pocházel ze společnosti Netscape.

Tohoto jazyka je využíváno jako interpretovaného programovacího jazyka pro WWW stránky, nejčastěji se vkládá přímo do HTML kódu stránky. Ovládá se s ním mnoho interaktivních prvků GUI (tlačítka, textová políčka) nebo se mohou vytvářet animace a efekty obrázků.

Co se týče syntaxe, tak patří do portfolia jazyků C/C++/Java. JavaScript byl v červenci 1997 standardizován asociací ECMA (European Computer Manufacturers Association) a v srpnu 1998 ISO (International Organization for Standardization). Standardizovaná verze JavaScriptu je pojmenována jako ECMAScript a z ní byly odvozeny i další implementace, jako je například ActionScript.

Pro verzi firmy Microsoft se používá názvu JScript. Ten je podporován platformou .NET. Program v JavaScriptu se obvykle spouští až po stažení WWW stránky z Internetu (tzv. na straně klienta), na rozdíl od ostatních jiných interpretovaných programovacích jazyků (např. PHP a ASP), které se spouštějí na straně serveru ještě před stažením z Internetu. Z toho plynou jistá bezpečnostní omezení, JavaScript např. nemůže pracovat se soubory, aby tím neohrozil soukromí uživatele.

JavaScript je možné použít i na straně serveru. První implementací JavaScriptu na straně serveru byl LiveWire firmy Netscape vypuštěný roku 1996, dnes existuje několik možností včetně opensource implementace Rhinola založené na Rhino, gcj, Node.js a Apache.

Kromě DHTML se JavaScript používá k psaní rozšíření pro mnohé aplikace, například Adobe Acrobat. [7]

3.4. MySQL

MySQL byla vytvořena v roce 1995 jako jednoúčelová databáze pro snadné ukládání a především čtení textových dat v internetových aplikacích. Vznik MySQL se spojuje se švédskou společností MySQL AB, která se starala o vývoj, uživatelskou podporu a obchod. Její zakladatelé jsou dva Švédové a jeden Fin: David Axman, Allan Larsen a Michael „Monty“ Widenius. MySQL AB vlastní práva na zdrojový kód, obchodní

značku a doménu mysql.com. Hlavní myšlenkou této firmy bylo vytvoření takového databázového systému, který by byl dostupný všem. Počet instalací MySQL již přesáhl 2 milióny, čímž se stalo MySQL nejpoblárnějším databázovým systémem, přičemž se nedá říci, že je to databázový systém pro chudé, protože mezi spokojené zákazníky patří například Yahoo! Finance, MP3.com, Motorola, NASA, Silicon Graphics nebo Texas Instruments. Databáze MySQL klade hlavní důraz na maximální rychlost, čímž si získala oblibu především u méně obsáhlých internetových projektů, kde je co nejkratší doba odezvy nutností. Na druhou stranu je třeba říci, že za rychlost se platí absencí uložených procedur (stored procedures) nebo spouští (triggerů). Vzhledem k faktu, že poskytovatelé webhostingu mají tento databázový systém za standard, není snad pro programátora PHP lepší volby, a proto jsem ho zvolil i já ke své práci.

Struktura MySQL

V nerelačním databázovém systému jsou všechny informace uloženy v jedné obrovské oblasti v níž jsou jakékoli operace související se získáváním informací mnohem obtížnější a těžkopádnější. MySQL je relační databázový systém, který vám umožňuje rozdělit informace do tabulek, což jsou skupiny souvisejících informací. Každá tabulka se skládá ze samostatných polí nebo-li sloupců, které zastupují jednotlivé části uložené informace. V jednom poli je uloženo například jméno uživatele, v dalším poli třeba jeho příjmení. Jednotlivá pole mohou nést informace různých typů – například text, čísla, kalendářní data a podobně. [5]

3.5. Apache

Apache je softwarový server, tedy program, který běží na hardwarovém stroji připojeném do internetu a zajišťuje obsluhu prohlížečů jednotlivých návštěvníků (posílá jim jednotlivé stránky). Mezi výhody Apache patří zejména dostupnost pro všechny hlavní platformy (Windows, Linux...) a také fakt, že Apache je vyvíjen jako open source (a je tedy k dispozici zdarma).

Společně s PHP (serverové skriptování) a MySQL (databáze) patří Apache k tzv. triádě, trojici programů nejčastěji používaných k vytváření dynamických internetových stránek. Svou roli v tom hraje i poměrně jednoduchá instalace Apache na domácích počítačích, čímž lze zdarma vytvořit velmi kvalitní vývojové prostředí. Nepřekvapí tedy, že podle

statistik z roku 2005 slouží Apache na 69 % všech internetových serverů. Konkrétně podle společnosti Nestcraft (www.netraft.com) provozuje server Apache více než 103 milionů internetových serverů, což je více než mají Microsoft, Sun ONE a Zeus dohromady. Název Apache vznikl z anglického „a patchy server“, tedy záplatovaný server. Název se odvozuje z historie programu, kdy byl Apache mezi správci serverů používán s mnoha různými záplatami a úpravami. [3]

Konfigurace Apache

Konfigurační soubor se jmenuje `httpd.conf` a nachází se v adresáři `/etc/apache`. V tomto souboru se již nacházejí a jsou obsaženy příklady, které jsou bohatě okomentované. Níže jsou uvedeny základní hodnoty nastavení.

```
ServerRoot "/usr"      #kořen serveru, odsud se odvíjí relativní cesty
User "nobody"
Group "nobody"        #pod kým se apache spouští
ServerName "www.example.net"  #jméno serveru, musí být pravdivé
DocumentRoot "/var/www/htdocs" #kořen html dokumentů čili webu
```

3.6. Perl

Perl je programovací jazyk, který primárně sloužil jako pomůcka pro správu systému. Zahrnuje v sobě prvky jazyka C, `awk`, `grep`, `sed`, a Bourne shellu. Perl je velmi dobrým pomocníkem pro zpracování textu. I když je Perl popisován jako programovací jazyk pro správu systému, je možné ho používat i jinde jako třeba pro prototypy složitějších programů a všude tam, kde je až dosud užíván zejména skript psaný v shellu. Slovo „Perl je zkratkou odvozenou z anglického Practical Extraction and Report Language.

Perl vytvořil a dále je rozvíje Larrym Wallem. Perl je volně šiřitelný pod GNU licenci a lze jej používat na celé řadě a operačních systémech. Mezi nejznámější patří zejména velké množství variant OS UNIX 1 nebo VMS a dokonce i starý DOS a jeho následovníci.

Perl se v některých rysech podobá Bourne shellu, `awk`, `sed`, `grep` a navíc je schopen přistupovat k systémovým voláním a ke knihovnám funkcí poskytovaných standardní knihovnou jazyka C a tím zaplňuje prostor mezi shelovskými skripty a programy v jazyce C. Perl není kompilovaným jazykem, ale je rychlejší než většina interpretovaných jazyků.

Před vlastní aktivací skript je Perlův skript nejdříve načten programem perl do paměti počítače a předkompilován do rychlejšího vnitřního formátu. Ve více případech je perlův skript o mnoho svižnější než jeho analogie naprogramovaná v Bourne shellu. [13]

3.7. Webové editory poskytující podporu PHP

K psaní kódu je možné využívat mnoho různých programů. Některé mají více možností jiné zase méně. Níže je uvedena základní tabulka, kde je možné porovnat dostupné možnosti. Obsahuje seznam několika oblíbených editorů seřazených abecedně a nabízí porovnání běžných funkcí, jako je třeba zvýrazňování syntaxe. Mnohé z těchto editorů nabízejí podobné funkce, takže rozhodnutí, který z nich použít je vždy na programátorovi jaké mu vyhovuje více prostředí. [3]

Editor	Zvýrazňování syntaxe	FTP klient	Číslování řádků	Automatické doplňování	Spojení s databází	Ladění kódu v PHP	Integrace CVS/Subversion	Skrývání bloku kódu	cena
Dreamweaver CS4	Ano	Ano	Ano	Ano	Ano		Ano	Ano	15 000 Kč
EditPlus	Ano	Ano	Ano					Ano	1000 Kč
Notepad									Zdarma
PS Pad	Ano	Ano	Ano	Ano				Ano	Zdarma
Zend Studio	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ano	12000 Kč
NetBeans	Ano		Ano	Ano	Ano			Ano	Zdarma
TextPad			Ano						29 USD
PHPED	Ano	Ano	Ano	Ano	Ano	Ano	Ano	Ano	8500 Kč

Tabulka 1 - seznam editorů

4 Praktická část

4.1 Popis zadání realizačního projektu

V praktické části autor vytváří webové stránky pro společenství vlastníků jednotek v domě, ve kterém bydlí. Projekt je vytvořen za účelem poskytování informací vlastníkům domu a nájemníkům jednotlivých bytových jednotek. Na tyto dynamické stránky může též přistupovat kdokoli a získává zde základní informace, které jsou veřejné-dostupné komukoli, dále je zde sekce právě pro vlastníky bytových jednotek. Na tuto část se autor zaměřuje nejvíce. Přístup do této sekce bude umožněn pouze přes přístupové údaje jako je login „přihlašovací jméno“ a „password“ heslo. V této sekci se dále budou nacházet informace jako jsou například seznam vlastníků jednotek, případně nájemníci, kterým vlastníci bytovou jednotku pronajímají, bude zde informovanost o dlužných částkách jednotlivých bytových jednotek, stav účtu, který obhospodařuje výbor společenství, přístup k jednotlivým zápisům z porad společenství a výboru, domovní řád a další věci potřebné pro získání základních informací o stavu vlastněného domu jako celku.

V domu je rozvedena strukturovaná kabeláž do každé bytové jednotky. Celý dům je napojen na poskytovatele internetového připojení přes optický kabel až přímo k poskytovateli, což zaručuje velmi dobrou konektivitu internetu (tak i kabelové televize). Webové stránky budou vytvářeny pomocí jazyka PHP databáze MySQL a webového serveru Apache. Vlastní postup instalace bude autorem popsán níže. Veškeré základní požadavky na vytvoření webových stránek vznikly na základě sjednané schůzky výboru společenství a autorem webových stránek. Ze schůzky byl sepsán zápis s dohodnutými základními požadavky na stránky, hlavní body zápisu a představ výboru do vizuální interpretace byl následující:

- Na hlavní stránce budou zobrazeny aktuální informace o dění SVJ (bude zadávat člen výboru SVJ)
- Možnost hlasovací ankety
- Přihlašovací okno pro přístup do „chráněné sekce“ pouze pro uživatele SVJ
- Možnost vkládání připomínek
- Hlavní menu stránek s odkazem do jednotlivých sekcí viz obrázek níže:

- ❖ SVJ-KONTAKTY – odkaz na kontakty SVJ a jednotlivé členy včetně e-mail adres a telefonů
 - ❖ DOMOVNÍ ŘÁD - odkaz na domovní řád společenství
 - ❖ STANOVY – odkaz na základní dokument SVJ
 - ❖ ZÁPIS SCHŮZÍ SVJ – odkaz pro náhled veškerých dokumentů a zápisů SVJ, jak výboru, tak vlastního společenství. **Přístup pouze pro členy SVJ.**
 - ❖ FINANCE – odkaz na hospodaření SVJ, zobrazení informací o čerpaném úvěru, nákladových a výdajových stránkách hospodaření, platby nájmu vlastníků SVJ včetně případných nedoplatků a aktuálního stavu financí k určitému období. **Přístup pouze pro členy SVJ.**
 - ❖ DŮLEŽITÉ KONTAKTY – odkaz na stránky kontaktů opravárenských firem, kteří zajišťují servis pro SVJ, jako například voda, topení , plyn, výtahy atd..
- HLÁŠENÍ OPRAV – odkaz na stránku, kde bude možné zadávat, nahlašovat opravu. **Přístup pouze pro členy SVJ.** Požadavkem je zaslání e-mailové notifikace členům výboru SVJ.

4.2 Technická specifikace

Hardware

Jako hardwarová platforma byl zvolen server Fujitsu Siemens PRIMERGY TX150 S8
RAM 4Gb 2x500Gb HDD, CDROM



Obrázek 2 - testovací PC

Jedná se o nejnovější dvouprocesorovou platformu Intel a pevných disků SAS 2.0 s velmi nízkou spotřebou energie. Hot-Swap disky umožňují výměnu za provozu. Rozšiřitelnost RAM je až na 96Gb. Tento počítač bude sloužit k základnímu naprogramování PHP skriptů a webových stránek společenství vlastníků jednotek.

Kompletní základní technické parametry jsou uvedeny v tabulce níže:

Čipová sada	Intel® C602
Typ základní desky	D 3079
Typ produktu	Jednopaticový server v provedení tower
Procesor	Procesor Intel® Pentium®, Procesory řady Intel® Xeon® E5-1400, Procesory řady Intel® Xeon® E5-2400
Paměť	2 GB - 96 GB, DIMM (DDR3)
Ochrana paměti	Advanced ECC, Technologie Memory Scrubbing, SDDC (Chipkill™), Podpora okamžité náhrady paměti, Podpora zrcadlení paměti
PCI-Express 3.0 x4 (mech. x8)	2 x Plná výška délka 280 mm
PCI-Express 3.0 x16	1 x Plná výška délka 280 mm
PCI-Express 2.0 x4 (mech. x8)	1 x Plná výška délka 230 mm; upřednostňovaný slot RAID
Sloty PCI	1 x PCI 32 bitů / 33 MHz, 1x dlouhý, 5 V
Poznámky ke slotu	v konfiguraci SAS je obsazen 1x slot PCI-Express modulárním řadičem RAID.
Řadič I/O integrovaný na desce	Intel® C602, 2 porty použity pro zvenku přístupné jednotky 4 porty pro interní pevné disky SATA, RAID 0, 1 a 10 pro Windows i Linux; další možnosti řadiče RAID viz Součásti – Řadič RAID 2x Ethernet 10/100/1000 Mb/s Integrovaný řadič vzdálené správy (iRMC S3, 32 MB přídavné paměti vč. grafického řadiče), Kompatibilní s IPMI 2.0
Standardní výstup napájecího zdroje	500 W (účinnost 90 %, certifikace 80 PLUS Gold)
Výstup za provozu vyměnitelného napájecího zdroje	450 W (účinnost 94 %, certifikace 80 PLUS Platinum)
Hmotnost	16–29 kg
Rack (Š x H x V)	483 x 611 x 177 mm

Provedení tower (Š x H x V)	177 x 651 x 456 mm
Speciální vlastnosti	Nejvyšší výkon mezi jednopaticovými servery, režim tichého provozu pro použití v kancelářském prostředí. Ideální pro databáze, software ERP a CRM a řešení virtualizace.
Standardní záruka	3 roky
Životní cyklus služby	5 let po skončení životnosti produktu

Server byl nainstalovaný s operačním systémem Windows 7 ultimate. Pozn. Je možné využít též open source systém jako je Linux Fedora, důvod zvolení systému Microsoft bude uveden v kapitole 2.10.

4.3 Software

Jeden z hlavních důvodů, proč autor použil operační systém Windows 7 je, že bylo použito aplikace WampServeru ve které jsou zahrnuty veškeré potřebné balíčky potřebných k řádnému běhu webového serveru s použitím skriptovacího jazyka PHP.

WampServer je open source aplikace, která umožňuje vytvářet webové aplikace za použití Apache, PHP a MySQL databáze.

Zde bylo použito WampServe 2.4. Jedná se o instalační balíček, který po instalaci automaticky připraví a nainstaluje tyto potřebné komponenty:

Wamp Server 2.4

Apache 2.4

Mysql 5.6.12

PHP 5.4.16

PHPMy admin 4.0.4

SQLBuddy 1.3.3

XDebug 2.2.3

Vlastní název „WAMP“ Serveru vznikl ze zkratk: “**W**” jako Windows, “**A**” jako Apache, “**M**” jako MySQL, a “**P**” jako PHP.

V našem případě bude využito právě zejména těchto balíčků:

Apache:

Apache je celosvětově nejpoužívanějším webovým serverem, který má výhody především otevřeností svého kódu, multiplatformním využitím (běží jak pod Windows, tak například i pod Linuxem) a cenou. Je totiž poskytován a distribuován zdarma k plnému užití.

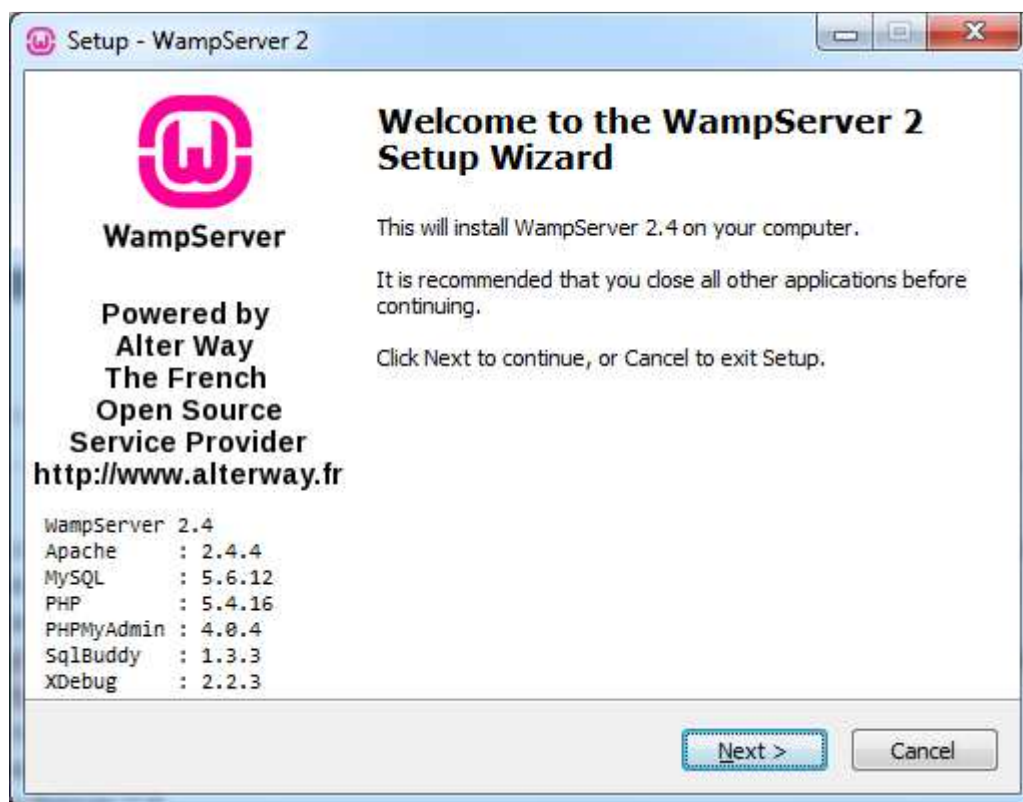
MySql:

MySQL je dalším multiplatformním počinem poskytovaným také zdarma, jedná se o databázový systém, který je nutností každého dynamického webu s proměnným obsahem.

PHP:

PHP jako programovací jazyk umožňující vytváření webů, webových aplikací a práci s databází zakončuje tento nevelký seznam aplikací pro server.

4.4 Instalace WampServeru



Obrázek 3 - instalace WampServer

Program WampServer je open source řešení a je nainstalovaný do adresáře C:\wamp. Při instalaci nás instalace vyzve k zadání defaultního browseru v našem případě byl zvolen Firefox.

V dalším kroku instalace jsme vyzváni k zadání smtp serveru a e-mailu pro Apache.

Po úspěšné instalaci se zobrazí v dolní liště Windows ikona Wamp Serveru. Ikona indikuje dva stavy:

Červeně: Web server je offline

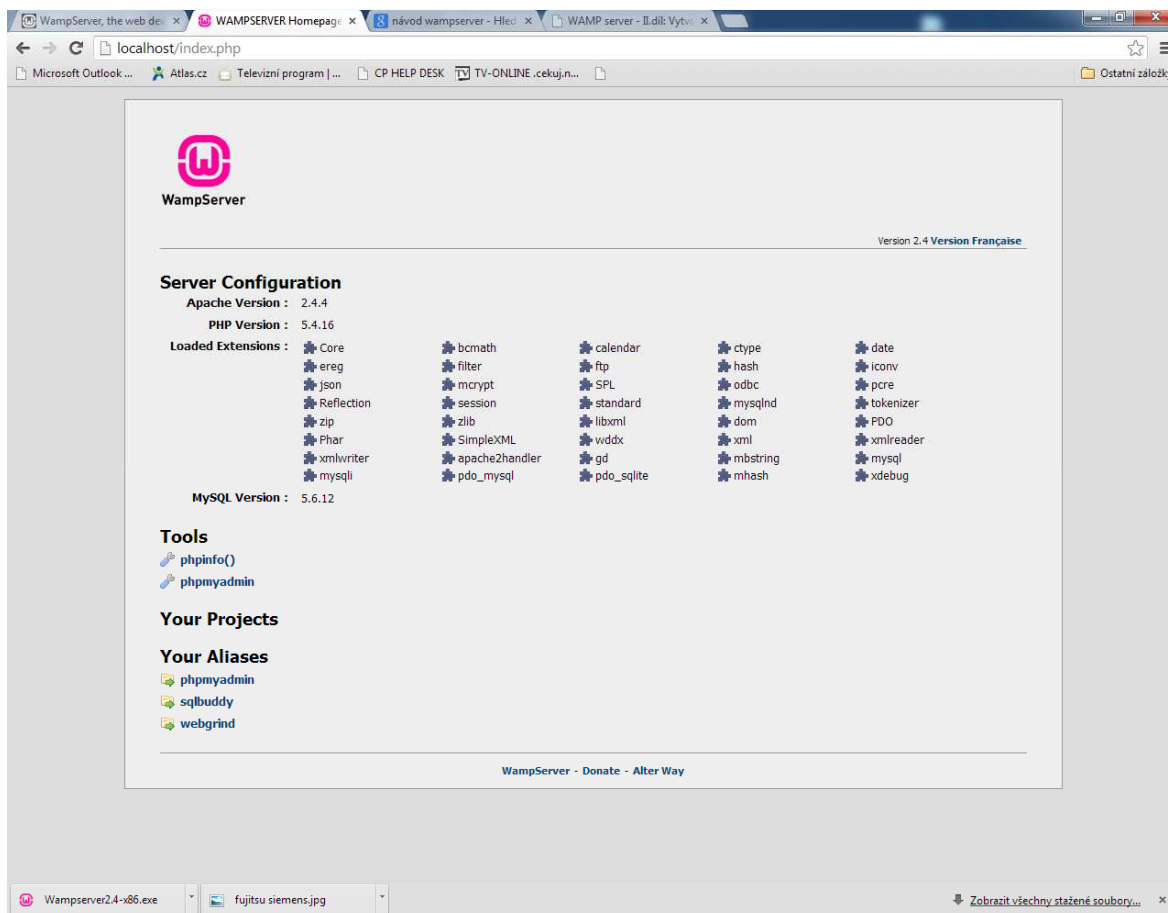
Zeleně: Web server je online

V případě, že po instalaci nenaběhne aplikace WampServeru korektně je „off-line“ je nutné aplikaci restartovat. Menu WampServeru je velmi pěkně přehledné. Je možné provést restart pouze vybrané služby nebo všech služeb pro správných chod WampServeru. Požadovaný postup je interpretován na obrázku níže v menu „Quick Admin“.



Obrázek 4 - menu WampServer

Pokud se nám vše podaří vše korektně nainstalovat zobrazí se nám po zadání do prohlížeče <http://localhost/index.php> stránka nově nainstalovaného WebApache serveru s funkcemi dynamického stránkování za použití PHP skriptů



Obrázek 5 - úspěšně nainstalovaný WampServer

Jednotlivé vlastnosti a chování webového serveru lze modifikovat pomocí parametrů, které se nachází v souboru „httpd.conf. Jedná se zde například o základní parametry:

- *LoadModule php5_module*
"c:/wamp/bin/php/php5.4.16/php5apache2_4.dll" / modul pro funkcionalitu PHP skriptování na webovém serveru apače.
- *DocumentRoot "c:/wamp/www"* / root adresář pro ukládání skriptů, html, xhtml souborů
- *ErrorLog "c:/wamp/logs/apache_error.log"* /soubor, který zaznamenává error hlášky webového serveru, což je důležité v případě zjišťování problémů

Webový server Apache běží jako služba v operačním systému Windows. [18]

V root adresáři www stránek jsou doplněny tyto podadresáře:

_css adresář pro ukládání kaskádových stylů

<code>_images</code>	adresář pro obrázky, které využívají webové stránky
<code>_aktuality</code>	pro ukládání aktuální informací pro společenství SVJ
<code>_dokument</code>	pro ukládání zápisů z porad SVJ a výboru společenství
<code>_namety</code>	jednotlivé dotazy, které vytváří uživatel na stránkách SVJ
<code>...root</code>	zde jsou uloženy vlastní PHP skripty

O správné funkčnosti vlastního skriptovacího jazyku PHP se můžeme přesvědčit zadáním do prohlížeče <http://localhost/index.php?phpinfo=1>

Pokud se zobrazí základní vlastnosti skriptovacího jazyku PHP máme vše správně nainstalované.

Nyní jsou nainstalované všechny komponenty potřebné pro tvorbu dynamických webových stránek v jazyku PHP.

Před vlastním vytvářením skriptů PHP autor doporučuje v souboru `php.ini` modifikovat některá nastavení a to zejména:

`Error_reporting` – nastavení této hodnoty určuje vypisování chybových hlášek, které mohou být vícero druhů:

- `Parse` chyby při čtení kde se jedná o syntaktické chyby
- `Fatal` mohou to být chyby například, když nemáme definovanou nějakou proceduru či funkci
- `Warnings` – Jsou to chyby, kdy se může jednat o závažné, ale PHP se domnívá, že je možné pokračovat dále
- `Notices` – v tomto případě se může jednat například o nevyužívané proměnné v běhu skriptu atd...

`Error_reporting` je možné zapnout v úrovních `E_ALL` nebo pouze chyby a varování `E_ERROR|E_WARNING|E_PARSE`.

Zde se domnívám, že při testování a vytváření je vhodné nastavit hodnotu na `ALL` a až po úplném otestování a nasazení do komerčního prostředí je možné úroveň výpisu chybových hlášek snížit.

4.5 Struktura a syntaxe jazyka PHP

Program v jazyku PHP lze psát stejně jako stránky HTML v textovém editor, jakým je například poznámkový blok, PSPad. Na rozdíl od stránek HTML však mají soubory PHP příponu .php. Tato přípona sděluje server, že soubor musí předat skriptovacímu stroji PHP a teprve výsledek činnosti tohoto stroje lze odeslat zpět do webového prohlížeče klienta. Kód HTML lze psát na webové stránce také uvnitř kódu PHP, což nám umožňuje formátovat text, aniž bychom museli rozbíjet jednotlivé bloky kódu. Kód může být proto mnohem lépe organizován, vývojář je schopen vytvářet efektivnější kód a prohlížeč si žádného rozdílu nemusí všimnout. Kód jazyka PHP lze psát také jako samostatný program bez výstupu HTML. To je užitečné například pro ukládání proměnných obsahujících údaje o připojení k databázi nebo o přesměrování návštěvníků na další stránky webu. Jednou z výhod práce s jazykem PHP je jeho relativně jednoduchá a přímočará syntaxe. Stejně jako v mnoha dalších počítačových jazycích lze i v PHP vykonat stejnou činnost více způsoby.

Webové stránky mají být navrženy, tak, aby na stránky mohl kdokoliv, vidět co je základním obsahem informace o SVJ, kontakty, vypsání domovní řád a další což lze řešit statickými stránkami, ale dalším požadavkem bylo, aby bylo možné:

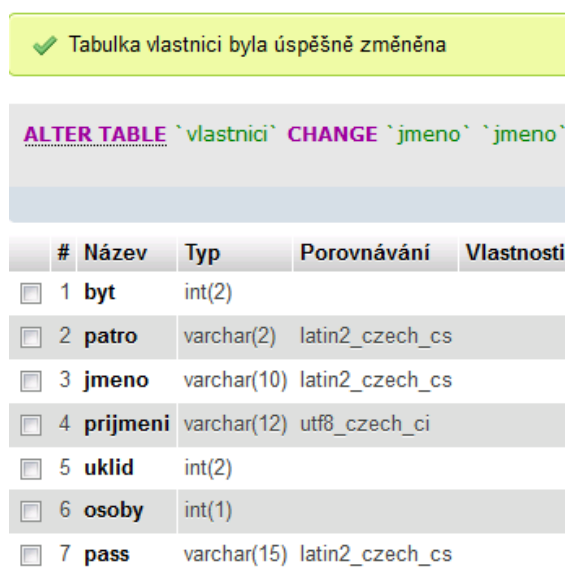
- Přihlásit se do stránek sekce vlastníci bytových jednotek
- Vepisování poznámek a dotazů na výbor SVJ
- Hlasování o případných návrzích/spokojenosti atd...
- Nahlédnutí na všechny vlastníky bytových jednotek včetně výpisu případných nedoplatků nájmu
- Nahlašování závad vlastníky bytových jednotek a e-mailová notifikace v případě vyřešení požadavků.
- Nástěnka přehledných základních informací

4.6 Vytvoření MySQL databáze

Základem pro vytvoření dynamických stránek bylo vytvoření databáze v MySQL. Založení databáze lze provést v MySQL konzoly a nebo pomocí aplikace "WAMP" , které

v sobě obsahuje webovské rozhraní PHP MyAdmin přes, které je možné vytvářet databáze tabulky. Lze zde provádět rovněž zálohy databází, jejich export modifikaci, vest uživatelské účty a mnoho dalšího. V našem případě bylo použito právě webového rozhraní, není zde nutné znát detailně SQL syntaxe a vytváření tabulek je velmi jednoduché.

Vlastní databáze byla nazvána “podparkem” a tabulka “vlastníci” obsahuje tyto položky:



Obrázek 6 - vytvoření tabulky v PHP MyAdmin

Byt- položka udává číslo bytové jednotky

Patro- položka udává patro v jakém se nachází bytová jednotka

Jméno- položka udává jméno vlastníka bytové jednotky

Příjmení- položka udává jméno vlastníka bytové jednotky (v našem případě bude použito jako přihlašovací “login” do stránek

Uklid- položka pro číslo v týdne v roce, kdy má daná bytová jednotka úklid společných prostor.

Osoby- položka pro počet osob bydlících v bytové jednotce. Informace o uživatelích bytových jednotek autor získal od výboru společenství jednotek. Seznam byl předán v papírové podobě, tak naplnění tabulky nebylo možné provést pouhým importem dat např. z format.xls, ale muselo se provést manuální vložení dat.

V případě, že bychom chtěli využít přístupu do MySQL databáze pomocí “console” bylo nutné použít příkazových syntaxí.

Pro přihlášení do console a zadání hesla bylo použito příkazů:

- show databases;* - pro výpis všech databází
- use ic_kamenna8* - příkaz pro připojení se do database ic_kamenna8
- show tables* - příkaz pro výpis všech tabulek v databázi ic_kamenna8
- describe vlastnici* - příkaz pro výpis polí v tabulce vlastnici

```
mysql> describe vlastnici;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| byt   | int(2) | NO   | MUL | NULL    |       |
| patro | varchar(2) | NO   |     | NULL    |       |
| jmeno | varchar(10) | NO   |     | NULL    |       |
| prijmeni | varchar(12) | NO   |     | NULL    |       |
| uklid | int(2) | NO   |     | NULL    |       |
| osoby | int(1) | NO   |     | NULL    |       |
| pass  | varchar(15) | NO   |     | NULL    |       |
| tmp   | varchar(10) | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.12 sec)

mysql>
```

Obrázek 7 - zobrazení struktury tabulky

MySQL je služba, která běží na pozadí pro komunikaci využívá portu 3306, který lze s dalšími parametry modifikovat v `mysql.ini` souboru. Po instalaci MySQL je vytvořen uživatel „root“, který je bez hesla. Je doporučeno heslo nastavit z důvodu bezpečnosti, v našem případě, bylo heslo nastaveno „*heslo123*“

4.7 Psaní PHP, JavaScript a HTML kódu.

Pro psaní webových stránek bylo použito textového editoru PSPad, který v sobě má i kontrolu syntaxí pro XHTML i PHP skriptování. Umožňuje využívat nápovědy dalších skriptovacích jazyků jako je například java. Tento textový editor je freewarová aplikace, tak ji lze bezplatně využívat. Instalace tohoto programu je velmi jednoduchá.

Navíc tento textový editor má v sobě zakomponovaného i FTP klienta což je pak v případě modifikace skriptů velmi dobrá funkce a je možné skripty načítat a ukládat přímo na webový server v a případě, kdy lze použít možnosti FTP připojení k webovému serveru.

Samozřejmostí je i vytváření java skriptů v tomto textovém editoru. Další výhodou je, že je možné pracovat s více otevřenými soubory najednou což v případě vytváření skriptů nejen v PHP je velmi užitečná funkce. Dalším propracovaným editorem může být „NetBeans“, který má v sobě zakomponované též syntaxe pro skriptovací jazyky PHP. NetBeans jsou již mnohem propracovanější, ale v našem případě jich nebylo použito. PSPad není náročný na procesor ani na paměť. V našem případě bylo použito PSPad verze 4.5.7.

4.8 Základní PHP skripty stránek SVJ

Autor se zde nezmiňuje o základních syntaxích HTML, XHTML kódů psaných pro webové stránky, ale klade v této práci důraz na využívání PHP funkcí, které byly využity právě při vytváření webových stránek společenství vlastníků bytových jednotek.

Přihlašovací údaje

Základem pro připojení se do sekce privátních informací na webových stránkách, je přihlášení se do systému pomocí formuláře, kde budou zadány přihlašovací údaje (jméno/heslo). Jako přihlašovací jméno a heslo je použito právě informací z naplněné databáze „*ic_podparkem*“, kde jsou uvedeny veškeré podstatné údaje. Vlastní naplnění databáze bylo provedeno pomocí PHPMyAdmin, kde se naplnila tabulka vlastníků bytových jednotek. Jedná se zejména o údaje uživatelů bytových jednotek.

Dalším důležitým parametrem bude vytvoření session v PHP z důvodu zabezpečení.

Pro napojení do databáze bylo použito v PHP syntaxí:

```
<?php
Mysql_connect (“localhost”,”root”,”heslo123”);
Mysql_selectd_db(“vlastnici”);
?>
```

localhost je IP adresa server kde se nachází náš MySQL server v našem případě, že instalované vše na jednom počítači (prozatím testovacím) tudíž je možné využít jména

„localhost“ nebo loopbackové IP adresy 127.0.0.1 či reálné IP adresy v našem případě 192.168.1.2.

root je uživatelský účet pro přístup k databázi

heslo123 je heslo uživatelského přístupu. Domnívám se, že toto je nevýhodou skriptovacích jazyků, kde je heslo přímo zobrazeno/vepsáno ve skriptovacím souboru. Proto je důležité zabezpečit celý server z důvodu ochrany dat a přímému přístupu k souborům neoprávněným uživatelům.

4.8.1 Skript PHP - přístup do systému

Tento skript umožňuje oprávněným uživatelům přistoupit do systému a mít možnost nahlédnout do zápisů z jednotlivých porad, včetně nahlédnutí o všech nájemcích bytových jednotek. Právě v této části bylo nutné využít funkce v php „`<?php session_start();..... další kód.....?>`“, která právě zabezpečuje vstup na stránky pouze oprávněným osobám, v případě, uživatel není přihlášený dojde na stránce k informativní hlášce „nejsi přihlášen“ a uživateli neumožní přístup k informacím.

Pro kontrolu je-li uživatel přihlášený nebo není, slouží níže uvedený skript, který v případě, když se uživatel po zadání jména a hesla nepřihlásí do systému z důvodu špatně zadaného hesla nebo přihlašovacího jména, tak se tato událost administrátorovi webových stránek zapíše do souboru „*prihlasen.txt*“

```
if (isset($_SESSION['prihlasen'])): echo '<td width="180"
align="center" disabled>Přihlášený uživatel:<br>';
    echo $_POST['uzivatel'].'</td></tr></center></table>';
else:
    if ($_POST['uzivatel']>" " and mysql_num_rows($dotaz)<>1):
        echo '<td><span class="error_login">!!!Neplatny
login!!!</span><br></td></tr></center></table>';
        $prihl=fopen("prihlasen.txt","a+");
        $login=date("d.m.y H:i:s");
        $login.="\\t";
        $login.= $_POST['uzivatel']." nekdo se pokousi";
```

```
$login.="\r\n";
fwrite($prihl,$login);
fclose($prihl);
else:
    echo '<td width="180" align="center"><font
color="blue"><b>Nejsi
přihlášený</font><b><br></td></tr></center></table>';
endif;
endif;
```

Požadavkem administrátora bylo, aby v případě neoprávněného pokusu přístup o přístup přišla notifikační SMS, což bylo provedeno doplněním do skriptu PHP pomocí funkce:

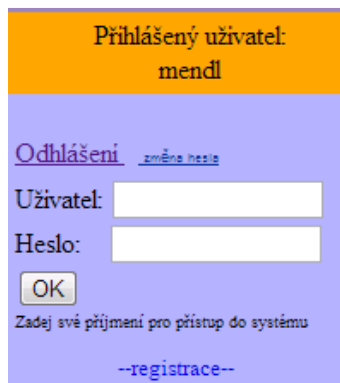
```
mail("00420602102XXX@vodafone.cz","nekdo se pokousi
prihlasit: ".$_POST['uzivatel'], "");
```

Nutností, je , aby byl funkční SMTP server a operátor umožňoval zasílání e-mail/sms notifikací.

Vlastní přihlašování bylo provedeno vyplňováním níže uvedeného formuláře, které po stisknutí tlačítka OK, vyvolával výše uvedeny PHP skript. Na obrázku č.8 je uživatel nepřihlášený na obrázku č.9 je již přihlášený.



Obrázek 8 - přihlašovací okno



Obrázek 9 - zobrazení přihlášeného uživatele

K odhlášení ze systému, slouží hypertextový odkaz „odhlášení“ na stránkách, který vyvolá skript

```
if ($_GET["ci"]==logout):  
    session_unregister("prihlasen");  
endif;
```

a po té dojde k odhlášení uživatele ze systému.

4.8.2 Skript PHP - registrace

Tento skript umožní uživatelům vyplnit registrační formulář se žádostí o přístup do systému. Nyní se jedná pouze o vyplnění jména, příjmení, čísla bytové jednotky a e-mailové adresy, na kterou budou informační údaje o přístupu „jméno/heslo“ zaslány administrátorem stránek po schválení výboru společenství. Množství vyplňovacích údajů je možné rozšířit o další informace. Po kliknutí na hypertextový odkaz „--registrace--“ na obrázku výše se aktivuje webová stránka registrace.php, na které se nachází registrační formulář. Formulář je vyvolán metodou pomocí html syntaxe ve formuláři „target=“_blank“ tzn. Webová stránka se zobrazí na nové záložce prohlížeče.

Jméno a příjmení:	<input type="text"/>
Byt č.:	<input type="text"/>
E-mail:	<input type="text"/> na tuto adresu bude zaslán přístup do systému
	<input type="button" value="odesli"/>

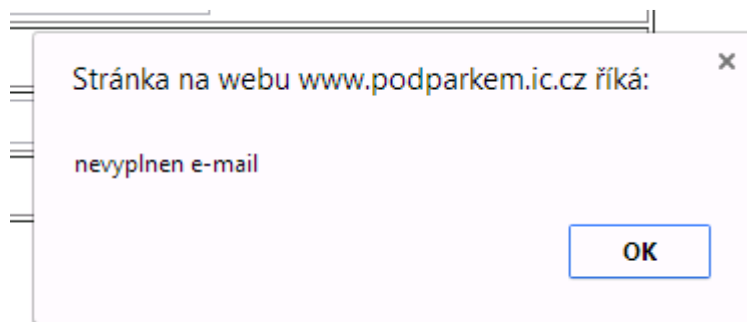
Obrázek 10 - registrační formulář

Je zde zabezpečena i ochrana pomocí javascriptů, že v případě kdy není vyplněno jméno nebo e-mail, tak nebude umožněno odeslání formuláře. Toto zabezpečení bylo provedeno pomocí příkazů java scripitu...

```
<SCRIPT type='text/javascript' language='javascript'>
document.getElementById('ted').focus();
</script>
<SCRIPT type='text/javascript' language='javascript'>
function kontrola()
{
var retval
retval=true;
if ((document.all['jmeno'].value==""))
{alert ("Nevyplněno jméno"); document.all['jmeno'].focus();
retval=false;return retval;}
if ((document.all['email'].value==""))
{alert ("nevyplněn e-mail"); document.all['email'].focus();
retval=false;return retval;}
}
</script>
```

Vlastní aktivaci javascriptu vyvolává <form> pomocí „*onsubmit="return kontrola();"“*

V případě, že je nevyplněna e-mailová adresa a je pokus o odeslání formuláře dostane uživatel informativní upozornění o nevyplněném formuláři:



Obrázek 11 - okno nevyplněné informace

V opačném případě při vyplnění veškerých náležitostí je formulář odeslán administrátorovi webových stránek. A uživatel dostane informaci o tom, že formulář byl řádně odeslán.

4.8.3 Skript PHP - změna hesla

Tento skript je důležitý při zakládání prvních přístupů do systému. Při žádosti o zřízení účtu výše uvedeným formulářem registrace. Musí administrátor vytvořit přes PHP MyAdmin uživateli nové heslo, které mu po té zasílá formou e-mailu, který uživatel vyplnil při registraci. Heslo tedy zná jak administrátor, tak uživatel. Po prvním přihlášení do systému, tak může oprávněný uživatel změnit své heslo pomocí hypertextového odkazu „změna hesla“. Tento odkaz se nachází v pravé části přihlašovacího okna a zobrazí se právě po prvním přihlášení. Po aktivaci hypertextového odkazu se zobrazí nová webová stránka pomocí skriptu PHP. Zde je uživatel vyzván k zadání původního hesla a zadání nového hesla včetně potvrzení nového hesla

zadej staré heslo	<input type="text"/>
zadej nové heslo	<input type="text"/>
zopakuj nové heslo	<input type="text"/>
	<input type="button" value="OK"/>

Obrázek 12 - formulář pro změnu hesla

Jsou zde provedeny kontroly formou java skriptů jako v předešlém případě při registraci nového uživatele tzn. Pokud uživatel nevyplní jakoukoliv položku, je vyzván

k opakovanému zadání. V případě, správného vyplnění a potvrzení změny hesla se veškeré údaje uloží do databáze (tabulky vlastníci) pomocí části skriptu:

```
if ($_GET["pass"]==2):
    $link = mysql_connect("mysql.ic.cz", "ic_podparkem", "xxxxxx");
    // $link = mysql_connect("localhost", "root", "root");
    mysql_select_db("ic_podparkem", $link);
    $dotaz = mysql_query("SELECT * FROM vlastníci where
pass='".$_POST['stare']."' ", $link);
    if (mysql_num_rows($dotaz)==1):
        if ($_POST["nove"]==$_POST["opak"]):
            $sql="update vlastníci set pass='".$_POST[nove]' where
pass='".$_POST[stare]'" ;
            $result=mysql_query($sql);
            $supr='nové heslo uloženo!!!';
        else: $info2='!!!spatne zadane heslo!!!';
        endif;
    else: $info1='!!!puvodni heslo nesouhlasí!!!';
    endif;
endif;
```

Zde si je autor vědom, že se do databáze ukládá heslo v podobě čitelného řetězce. Tzn. v případě, že se přistoupí přímo k databázi například přes PHP MyAdmin je toto heslo čitelné. K těmto účelům před vlastním uložením do databáze je možné využít funkce PHP MD5().

Před vlastním uložením nového hesla do databáze se vyvolá funkce, která přiřadí proměně \$heslo zašifrované heslo následujícím způsobem:

```
$heslo = MD5($_POST[nove]);
```

Pomocí updatu do databáze se pak zapíše heslo, které je zašifrované a nečitelné pro běžného uživatele, který má přístup přímo k datům. Stejným způsobem pak musíme přistupovat v případě, že přistupujeme do systému. Je nutné využívat funkci PHP MD5()

Dalším způsobem jak by bylo možné šifrovat hesla by bylo použitím funkce SHA1()

4.8.4 Skript PHP - připomínky a náměty

Jedním z požadavků bylo, aby se na základní stránce zobrazovali připomínky a náměty. Vlastní vyplňování námětů bylo provedeno pomocí formuláře, který ukládá jednotlivé připomínky a náměty do textových souborů, které jsou uloženy v adresáři `./namety`. Jednotlivé názvy souborů jsou označovány, tak, aby měli jedinečný název a to pomocí datumu a času s označením rok/měsíc/den/hodina/minuta.txt. viz například: `1310417160627.txt`, což značí, že požadavek námět byl vytvořen dne 17.dubna 2013 v 6 hodin 27 minut.

Po odeslání formuláře se zpracovává následující PHP skript, který ukládá jednotlivé náměty do souborů ve formátu výše uvedeném. Do obsahu souboru se nejdříve vepíše datum a čas, který je generovaný PHP skriptem a po té se tam vloží text, který byl napsán ve zvoleném formuláři.

```
<?php
if ($_POST["txt"]!=" ") : echo "Dekujeme Vam za vyplneni vaseho
nazoru ci dotazu";
    echo "<br>"; echo "Dotaz byl odeslan.";
    $zapis="namety/" .date('ymdHis') . ".txt";
    $soubor=fopen("namety/" .date('ymdHis') . ".txt", "x");
/ /fwrite($soubor, "-----
-." ."\r\n");
    fwrite($soubor,date('d/m/y') . " " .date('H:i:s') ."\r\n");
    fwrite($soubor,$_POST["txt"] ."\r\n");
    fwrite($soubor, "-----
-----
." ."\r\n");
    fclose($soubor);
    $soubor=fopen($zapis, "r");
    $date=fread($soubor, filesize($zapis));
    mail("00420602102XXX@vodafoneemail.cz", "| " .$_POST["txt"], "");
endif;
$headers .= "Content-Type: text/plain; charset=windows-1250"; //
Mime typ
```

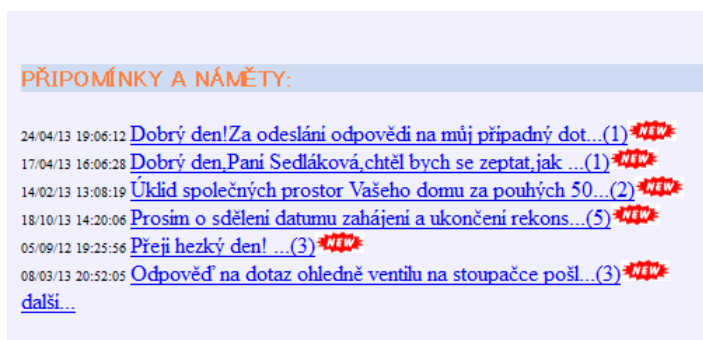
```
$soubor=fopen($zapis,"r");
$precteni=fread($soubor,filesize($zapis));
$soubor=fopen("adresy.txt","r"); //uvedeny email adresy na které
se odesílá info v případě, že byl založen nový námět/připomínka
while (!feof($soubor)){
    $radek=trim(fgets($soubor,4096));
    mail($radek,"nova pripominka",$precteni,$headers);
/* a ted' to odešleme */
mail($recipient, $subject, $message, $headers);
}
fclose($soubor);
?>
```

Celý tento skript je uložen v souboru namety.php, který právě vyvolává vyplněný a odeslaný formulář z webového prohlížeče.

autor si zde po odeslání formuláře nechává zaslat notifikaci formou SMS pomocí příkazu:

```
mail("00420602102XXX@vodafone.cz", "| ".$_POST["txt"], "");
```

Autor se zaměřil na jedinečnost názvu souborů z důvodů, že v případě, kdy někdo chce reagovat na již vytvořené připomínky, které jsou zobrazovány na hlavní stránce "index.php" viz obrázek níže. K jejich interpretaci (připomínek) byl zvoleno též skriptovacího jazyka PHP z důvodu, aby se zobrazovalo na stránce pouze prvních 40 znaků prvního řádku připomínky s datumem založení připomínky, jak je uvedeno níže.



Obrázek 13 - připomínky a náměty

Těchto 50 znaků bylo zvýrazněno hypertextovým odkazem, právě z důvodů případné reakce na danou připomínku. Tuto možnost doplnění reakce na připomínku zabezpečoval níže uvedený PHP script.

```
<?php
if ($handle = opendir('namety')) {
    while (false !== ($filename = readdir($handle))) {
        if ($filename != "." && $filename != "..") {
            // echo "$filename<br>";
            $pole[]=$filename;
        }
    }
    closedir($handle);
}
for ($i=0; $i<count($pole); $i++):
    $soubor=fopen(trim('namety/'.$pole[count($pole)-$i-1]),"r");
    $radecek++;
    while (!feof($soubor)){
        $pocet++;
        $radek=substr(fgets($soubor,4096),0,56);
        if (trim($radek)=="-----
-----"): $dotaz=$dotaz+1; endif;
        if ($pocet==1): echo "<font size=\"1\">".$radek." </font>";
endif;
        if ($pocet==2): echo '<a
href="pripominka.php?file='.$pole[count($pole)-$i-
1].'">'.$radek.'...</a>'; endif;
    }
    echo '<a href="pripominka.php?file='.$pole[count($pole)-$i-
1].'">('.$dotaz.')</a><br>';
    fclose($soubor);$pocet=0;$dotaz=0;
endfor;
?>
```

Ikonka „new“ na obrázku prezentuje, že někdo reagoval na zmíněnou připomínku v průběhu aktuálního dne, kdy se uživatel stránek dívá na stránky SVJ.

4.8.5 Skript PHP - výpis všech nájemníků(vlastníků) jednotek

Požadavkem výboru je, aby bylo možné zobrazit seznam všech nájemníků bytových jednotek, kteří byty obývají a v případě, že uživatel bytu není vlastníkem případně je u něj uvedena dlužná částka, taka by toto bylo označeno a interpretováno na webových stránkách, aby všichni vlastníci jednotek měli přehled o tom, který by je zadlužen-neplatí nájem a nebo je-li byt užíván nájemníkem nikoliv vlastníkem bytové jednotky.

K těmto účelům byl vytvořen PHP script seznam.php, jeho část je uvedena níže

```
<?php
    session_start();
    if (!session_is_registered("prihlasen")):
        header("location: index.php?cislo=error");
    endif;
echo '<table border=1 class="text" cellspacing="0" align=center>';
echo '<th align=center colspan=3>2563/28</th><th align=center
colspan=3>2564/30</th><th align=center
colspan=3>2565/32</th></tr>';
$sql="select byt,jmeno,prijmeni,osoby,dluh,najemnik from vlastnici
order by byt,ulice";
mysql_query("SET NAMES 'cp1250'");
$result=mysql_query($sql);

$barva="#B0B0FF";
$i=2;
while ($row = mysql_fetch_assoc($result)) {
    $i++;
    if ($i%3==0): echo '<tr bgcolor="' . $barva . '>';
        $barva = $barva <> 'white' ? 'white' : '#B0B0FF';
    if ($row['byt']==25): echo '<tr bgcolor="gray">'; endif;
    endif;
    if ((trim($row['najemnik']) <> '0') and ($row['dluh']== 0)):
        echo '<td>'.trim($row['byt']).'</td><td width=15
height=25></td><td>'.trim($row['prijmeni']).'
'.trim($row['jmeno']).' </td>';
        endif;
        if ($row['dluh']== 0 and (trim($row['najemnik'])== '0')):
            echo '<td
colspan=2>'.trim($row['byt']).'</td><td>'.trim($row['prijmeni']).'
'.trim($row['jmeno']).' </td>';
            endif;
            if ($row['dluh'] <> 0 and (trim($row['najemnik'])== '0')):
```

```
        echo '<td>'.trim($row['byt']).'</td><td width=15
height=25></td><td>'.trim($row['prijmeni']).'
.trim($row['jmeno']).' </td>';
    endif;
    if ((trim($row['najemnik'])<> '0') and ($row['dluh']<> 0)):
        echo '<td>'.trim($row['byt']).'</td><td width=15
height=25></td><td>'.trim($row['prijmeni']).'
.trim($row['jmeno']).' </td>';
    endif;
//echo '</tr>';
}
echo '</table>';
mysql_close($link);
?>
```

V první části skriptu se provádí kontrola pomocí příkazu `session` ta nám kontroluje je-li uživatel přihlášen do systému v případě, že není, tak se uživateli nezobrazí výpis vlastníků/nájemníků jednotek.

```
session_start();
if (!session_is_registered("prihlasen")):
    header("location: index.php?cislo=error");
endif;
```

V další části skriptu se již provádí (jeli uživatel přihlášen) napojení na databázi `ic_podparkem` a tabulku `vlastnici`. Pomocí dotazu:

```
$sql="select byt,jmeno,prijmeni,osoby,dluh,najemnik from vlastnici
order by byt,ulice";
mysql_query("SET NAMES 'cp1250'");
$result=mysql_query($sql);
```

Je proveden výpis všech vlastníků jednotek s grafickým označením ikonky u uživatele je-li nájemníkem nebo dalším grafickým označením dluží-li vlastník peníze do fondu oprav. Vlastní interpretace PHP a HTML kódu vypadá pak následovně viz. obrázek. Pokud se myší najede ikonku značící zadlužení (červená šipka) zobrazí se dlužná částka což je patrné na obrázku

2563/28		2564/30		2565/32	
1	Ing. Fiala Jan	1	✘ Petermann Lukáš	1	Tomšová Milena
2	❖ Louková Tereza	2	4890,-Kč sef	2	ing.Knežková Ivana
3	Bobek Zdeněk	3	Hlíza Josef	3	Čechová RNDr.
4	RnDr.Vostřelová Dana	4	Flasca Raluca	4	Blažek Miroslav
5	Ing. Tománek Milan	5	❖ Matuška Jiří	5	Kunešová Markéta
6	Lhotský Jan	6	Pešek Petr	6	Hejduk Pavel
7	Trojan Karel	7	Korber Josef	7	Feyfrlíková Jana
8	❖ Mgr. Karas Stanislav	8	❖ Mgr.Karas Stanislav	8	Pelc Zdeněk
9	Staňková Eva	9	Liscová Iveta	9	Ing. Bretfeld Otto
10	Ing. Orava Zdeněk	10	Kroisová Irena	10	Ing. Gajdica Josef
11	Navrátil Jaroslav	11	Knapp Karel	11	Peml Zdenk
12	Koleňová Lenka	12	Sedlák Petr	12	Formanová Jana
13	Mendl Jiří	13	Ing. Poulová Marie	13	Svoboda Václav
14	Hamouz Michal	14	❖ Sl.Trade s.r.o.	14	Peterka Jan
15	Piškanin František	15	Velek Tomáš	15	Kunertová Zdenka
16	Vorlíčková .	16	Čubán Jaroslav	16	Mgr. Volák Lubomír
17	❖ Mudr.Tajchner Lukáš	17	Marková Karolína	17	ing. Vinciková Andrea
18	Novák Petr	18	❖ Jeníková Alena	18	Lacina Miroslav
19	Kutilová Ivana	19	Unčovský Miroslav	19	Semerád Karel
20	❖ Lev Petr	20	❖ Richterová Dana	20	❖ Špecián Miloslav
21	❖ Písařík Jan	21	Salavová Růžena	21	Pešková Tat'jana
22	Vrbíček Bohumil	22	Makovský Miroslav	22	Pinkas Antonín
23	❖ Chlumecký Milan	23	Ing. Mokrý Michal	23	Santrucek Marek
24	Gubčová Alexandra	24	Zelenka Ondřej	24	Razák Václav
25	❖ Thu Ha Bui Thi obchod	25	❖ Jenčo Milan - obchod	25	❖ Perutka obchod
25	❖ Telefonika O2 sklep	25	❖ 5 obchod		

✘ Dlužník do fondu oprav
 ❖ Pronajímaná bytová jednotka

Obrázek 14 - seznam vlastníků SVJ

Požadavkem shromáždění bylo, aby byla možná také interpretace (zobrazení) sumární částky od všech nájemníků, kteří dluží do fondu oprav. Tuto interpretaci lze provést opět dotazováním na pole tabulky, kde jsou uvedeny jednotlivé dlužné částky a provedením sumarizace se výsledek interpretuje do prohlížeče. Tento požadavek byl proveden ve skriptu PHP finance.php viz níže:

```

<?php
    session_start();
    if (!session_is_registered("prihlasen")):
        header("location: index.php?cislo=error");
    endif;
    $link = mysql_connect("mysql.ic.cz", "ic_podparkem", "xxxxxx");
    
```

```
$db=mysql_select_db("ic_podparkem",$link);
$result = mysql_query("SELECT SUM(dluh) AS 'dluh' FROM vlastnici",
$link);
$row = mysql_fetch_assoc($result);
$dluh=$row['dluh'];
?>
```

V tomto případě se jedná o připojení se na databázi a provedení „selectu“ s funkcí „sum“ nad položkou „dluh“ v tabulce vlastnici. Po té je proměnná \$dluh interpretována v html tabulce s dalšími údaji o hospodaření společenství vlastníků jednotek.

STAV FINANČÍ K 30.6.2012		
Dlužná částka od neplatičů	Zůstatek na běžném účtě SVJ	601 037,20
6490,-Kč	Zůstatek na spořicíím účtu SVJ	1 820 421,38
	Zůstatek pokladní hotovosti SVJ	5 000,-
	Zůstatek na běžném účtě BD	97 210,79
	CELKEM	2 523 669,37

Obrázek 15 - stav financí

4.8.6 Skript PHP - zápisy schůzí SVJ

Toto menu je opět dostupné pouze pro vlastníky bytových jednotek, kteří mají přístup do systém webových stránek. Výbor jakož to výkonný organ SVJ zde ukládá veškeré zápisy z jednotlivých porad společenství, tak výboru. Požadavkem bylo, aby k těmto dokumentů měl přístup kdokoliv kdo je vlastníkem bytové jednotky a mohl si přečíst i zpětně historické zápisy z jednotlivých porad, nařízení, vyhlášek atd... Tyto zápisy ukládá jeden z členů výboru do adresáře “/dokumenty. Z tohoto adresáře je pak proveden výpis všech souborů (zápisů z porad), které jsou zde uloženy.

K výpisu těchto souborů slouží script seznam.php viz. níže

```
<?php
    session_start();
    if (!session_is_registered("prihlasen")):
        header("location: index.php?cislo=error");
    endif;
?>
```

```
<table border="0" width="550" align="center" >
<br>
```

```
<?php
$i=1;
//bgcolor="#F0F0FF"
```

```
$barva="#F0F0FF";
if ($handle = opendir('dokumenty')) {
    while (false != ($filename = readdir($handle))) {
        if ($filename != "." && $filename != "..") {
            $i++;
            if ($i%2==0): echo '<tr bgcolor="'. $barva. '">';

                $barva = $barva<>'white' ? 'white' : '#F0F0FF';
            endif;
            echo '<td></td>';
            echo '<td><font size="2"><a
href="dokumenty/'. $filename. '"
target="_blank">'. $filename. '</a></font></td>';
        }
    }
    closedir($handle);
}
?>
```

Na začátku skriptu se opět provádí kontrola je-li uživatel přihlášen do systém, v případě, že ano, pokračuje se dále skriptem, kde dochází k prezentaci výpisu vlastních souborů, které jsou uloženy ve zmiňovaném adresáři “dokumenty”. Pomocí příkazu “while” se provádí výpis všech souborů dokud se nezobrazí poslední soubor po té je cyklus ukončen. Pomocí rozhodovacího cyklu “if” se provádí interpretace barev řádku. Střídá se bílá s fialovou barvou.

Vlastní interpretace vypadá pak následovně



Obrázek 16 - zápisy z porad SVJ

4.8.7 Skript PHP - návštěvníci stránek.

Pro možnost sledování návštěvnosti stránek je vytvořen script, který kontroluje jednotlivé přístupy na stránka jejich přístupy ukládá do souboru "login.txt", který je uložen v root adresáři webových stránek. Při každém načtení stránky dojde k počtu řádek z textového souboru. Po té je proměná \$pocitadlo zobrazena na stránkách jako počet návštěvníků.

```
<?php
    $soubor=fopen("login.txt","r");
    $pocitadlo=-1;
    while (!feof($soubor)) {
        $buffer = fgets($soubor, 4096);
        $pocitadlo++;
    }
?>
```

Pro přesnější identifikaci přístupu na stránky je v PHP použito získání informace o poskytovateli vzdálené IP adresy, která přísluší návštěvníkovi. Tato informace se ukládá do výše uvedeného souboru. Je zde využita funkce setcookie, do které se ukládá proměnná. Tato proměnná má životnost 1hodinu. Po jedné hodině tato cookies zaniká. Tato vlastnost je tam z důvodu, aby návštěvník, který přistoupí na stránky a prohlíží si je, tak aby při každém načtení stránek se nepřipisoval další záznam do souboru login.txt. Pokud bychom tuto cookies nevyužili, tak počet návštěvníků by nebyl až tak přesný, protože každé načtení stránky by znamenalo záznam v souboru a tím pádem navýšení počtu návštěvníků. Skript pro ukládání IP adresy přistupující na webové stránky je následující:

```
<?php
if (!isset($test)):
    setcookie("test", "petr", time()+3600);
    $soubor=fopen("login.txt","a+");
    $adres=date("d.m.y H:i:s");
    $adres.="\t";
    $adres.=( $_SERVER['REMOTE_ADDR'] );
    $adres.="\t";
    $adres.=gethostbyaddr( $_SERVER['REMOTE_ADDR'] );
    $adres.="\r\n";
    //if ($refresh!=1):
    //    mail("00420602102XXX@vodafonemail.cz","|
".gethostbyaddr( $_SERVER['REMOTE_ADDR'] ), "");
    mail("mendl.jiri@umplus.cz","|
".gethostbyaddr( $_SERVER['REMOTE_ADDR'] ), "");
    fwrite($soubor, $adres);
    fclose($soubor);
//endif;

// mail("00420602102XXX@vodafonemail.cz","|
".gethostbyaddr( $_SERVER['REMOTE_ADDR'] ), "");
endif;
?>
```

Soubor “login.txt” neslouží právě jen k funkci počtu návštěvníků stránek, ale zároveň může sloužit jako logovací soubor, kde je možné vidět, které IP adresy přistupovaly ke stránkám včetně záznamu o datumu a času přístupu. Výpis tohoto souboru je pak následující:

```
20.04.13 10:46:24 85.70.1.13.30 30.1.13.broadband3.iol.cz
20.04.13 11:09:17 82.100.11.78 u11-78.static.grapesc.cz
20.04.13 12:15:59 82.100.11.78 u11-78.static.grapesc.cz
20.04.13 17:31:25 62.77.121.120 pool120.tetinet.cz
20.04.13 21:17:02 62.77.121.120 pool120.tetinet.cz
20.04.13 21:17:41 85.160.59.128 ip-85-160-59-128.eurotel.cz
29.04.13 17:59:37 85.160.1.2 ip-85-160-1-2.eurotel.cz
04.05.13 12:06:47 62.77.121.120 pool120.tetinet.cz
06.08.13 06:34:43 213.29.17.19 vba.complus.cz
25.08.13 18:38:57 85.160.25.181 ip-85-160-25-181.eurotel.cz
25.09.13 14:56:21 62.77.121.120 pool120.tetinet.cz
25.09.13 15:58:10 62.77.121.120 pool120.tetinet.cz
25.09.13 16:07:05 127.0.0.1 localhost
```

4.8.8 Skript PHP - nástěnka

Pomocí tohoto skriptu dochází k prezentaci základních informací , které společenství vlastníků jednotek požaduje, aby informovalo všechny uživatele bytových jednotek. Je to vlastně hlavní stránka webu. Výbor zde může prezentovat svoje návrhy, postupy, může uživatele informovat o stavu pokračujících oprav, o svozu objemného odpadu, do informací důsledného dodržování domovního řádu. Koncepce tohoto skriptu je taková, aby administrátor, který vkládá různé informace na stránky mohl vytvořit textový soubor do kterého vepíše základní informace jako je datum případně popis informace, kterou chce mezi vlastníky jednotek předat. Nástěnka je koncipována tak, aby na hlavní stránce byly zobrazovány minimálně 3 zprávy. Každá zpráva má určitý počet znaků, který se na stránce zobrazuje, v případě, že počet znaků (zprávy) je překročen vepíše se do zprávy hypertextový odkaz [..... celý článek.....](#) , po jeho rozkliknutí se zobrazí uživateli kompletní celá zpráva. Jednotlivé soubory administrátor ukládá do adresáře “/aktuality” po jejich vložení script PHP zajistí automatické/dynamické načtení těchto zpráv do webového prohlížeče, není tedy nutné vstupovat do zdrojového kódu HTML.

Naprogramovaný PHP skript aktuality.php:

```
<?php
if ($handle = opendir('aktuality')) {
    while (false !== ($filename = readdir($handle))) {
        if ($filename != "." && $filename != "..") {
            //echo "$filename<br>";
            $pole1[]=$filename;
        }
    }
    closedir($handle);
}
for ($i=0; $i<5; $i++):
    $radek=0;
    echo '<tr><td width="600" valign="top">';
    $soubor=fopen(trim('aktuality/'. $pole1[count($pole1)-$i-1]),"r");
    //$file='aktuality/'. $pole1[count($pole1)-$i-1];
    //echo $file.'<br>';
    while ((!feof($soubor)) and ($radek<>6)){
        $zprava=fgets($soubor,4096);
        if ($radek==0):
            echo '<div class="actual">'. $zprava.'</div>';
        else:
            if ($radek==4):
                echo substr($zprava, 0, 60).'...';
                //echo 'debile jeden velikej';
                echo '<a
href="index.php?cislo=detail&file=' . $pole1[count($pole1)-$i-1].'">
celý článek...</a>';
            else:
                echo $zprava.'<br>';
            endif;
        endif;
        $radek== $radek++;
    }
    fclose($soubor);
    echo '</td></tr>';
    $radek=0;
endfor;
?>
```

V případě, že je požadováno, aby informace obsahovala i obrázek případně tučný text, je zapotřebí mít základní znalosti HTML kódu, protože tuto informaci administrátor vepíše právě do vytvářeného textového souboru, který je ukládán ve zmiňovaném adresáři “/aktuality”.

Vlastní prezentace nástěnky/aktualit ve webovém prohlížeči se po uplatnění PHP skriptu prezentuje následujícím způsobem:

16.5.2013

Zpráva ve věci výtahových strojů.

Dobrý den pani Sedláková,

... [celý článek](#)... Tímto e-mailem Vám chci sdělit následující zprávu.

1.9.13 domovní řád

Prosíme nájemníky, aby dodržovali domovní řád, který byl schválen shromážděním.

Pravidla [domovního řádu](#) jsou k nahlédnutí v levé části obrazovky menu **OBSAH:**

20.5.13 Info

Vážení vlastníci bytových jednotek,

vítáme vás na stránkách SVJ Pod Parkem 2563-2565.

Na těchto stránkách vás rádi budeme informovat o aktuálním dění SVJ. Budete mít zde možnost napsat vaše připomínky a náměty, které nám mohou pomoci ke zlepšení našeho SVJ. Budou zde k dispozici aktuální zápisy ze schůze shromáždění, které si bude moci každý vlastník bytové jednotky stáhnout.

Výbor společenství ... [celý článek](#)...



Obrázek 17 - nástěnka

4.8.9 Skript PHP - admin

Posledním požadavkem vedení společenství SVJ bylo, aby mohl provádět editaci vlastníků jednotek jeden z členů výboru společenství. Zadáním bylo navrhnout jednoduchý přístup k databázi, kde bude možné měnit vlastníky jednotek, upravovat jim případné nedoplatky za nájem nebo označovat uživatele bytu zda se jedná o nájemníka či nikoliv. Tato možnost byla naprogramována přístupem na stránky <http://www.podparkem.ic.cz/adm.php>. Tento odkaz na stránky vyvolal administrátorský přístup k databázi po zadání jednoduchého hesla. Po té se zobrazí stránka, kde jsou uvedeni všichni nájemníci. Vzhledem k tomu, že SVJ je složeno ze tří vchodů, tak jednotlivé sloupce označují jednotlivé vchody. Po najetí myši na obrázek „tužky“ jako hypertextového odkazu se dostaneme k editaci záznamu vlastního nájemníka, kde pak můžeme upravovat jednotlivé položky jako je: Jméno, Příjmení, Dlužná částka, počet osob v domácnosti, zda se jedná o nájemníka, či nikoliv atd...Veškeré tyto informace se ukládají do databáze tabulky vlastníci. Opět při vyplňování (editaci) formuláře vlastníka jednotky je použito „javaskriptu“, tak aby administrátor v případě, že nevyplní základní

hodnoty, jako je např: jméno, příjmení, tak mu nebylo umožněno uložení do databáze s prázdnými hodnotami. Vlastní skript pro „jednoduchý“ přístup k editaci stránek pouze se statickým zadáním hesla bylo použito skriptu:

```
<?php
session_start();
if (!session_is_registered("logine")):
    echo 'neoprávněný přístup';
    echo '<br><br><table align=center valign=center><tr><td>
<form action="adm.php" method="post" name="login">
    zadej heslo:
    <input type="password" name="login" size="5" maxlength="5"
id="ted">
    <input type="submit" name="OK" value="OK">
</form></td></tr></table>';
endif;
if ($_POST['login']=='park'):
    session_register("logine");
    header("location: adm.php");
endif;
?>
```

V případě správného zadání hesla v našem případě „park“ se zobrazí stránky s jednotlivými údaji vlastníků jednotek, tak jak je uvedeno výše. Pro interpretaci stránek bylo použito níže uvedeného skriptu:

```
// nacteni najemniku -----
-----
if (((!isset($_GET['polozka'])) or (isset($_GET['ulozit']))) and
(session_is_registered("logine"))):
$sql="select id,byt,jmeno,prijmeni,osoby,dluh,najemnik, last_login
from vlastnici order by byt,ulice";
$result=mysql_query($sql);
echo '<table border=1 class="text" cellpadding="0" align=center>';
echo '<th align=center colspan=5>2563/28</th><th align=center
colspan=5>2564/30</th><th align=center
colspan=5>2565/32</th></tr>';
//echo '<table cellpadding="0">';
$barva="#B0B0FF";

$i=2;
while ($row = mysql_fetch_assoc($result)) {
$log=$row['last_login'];
if ($log=='0000-00-00 00:00:00'): $log='';
endif;
    $i++;
    if ($i%3==0): echo '<tr bgcolor="'. $barva.' ">';
        $barva = $barva <> 'white' ? 'white' : '#B0B0FF';
    endif;

//if ($row['byt']==25): echo '<tr bgcolor="gray">'; endif;
```

```

        if ((trim($row['najemnik'])<> '0') and ($row['dluh']== 0)):
            echo '<td><a href="adm.php?polozka='.$row['id'].'"><td>'.trim($row['byt']).'</td><td width=15
height=25></td><td>'.trim($row['prijmeni']).'
.trim($row['jmeno']).'</td><td class=sedy>'.$log.'</td>';
            endif;

        if ($row['dluh']== 0 and (trim($row['najemnik'])== '0')):
            echo '<td><a href="adm.php?polozka='.$row['id'].'"><td
colspan=2>'.trim($row['byt']).'</td><td>'.trim($row['prijmeni']).'
.trim($row['jmeno']).' </td><td class=sedy>'.$log.'</td>';
            endif;

        if ($row['dluh']<> 0 and (trim($row['najemnik'])== '0')):
            echo '<td><a href="adm.php?polozka='.$row['id'].'"><td>'.trim($row['byt']).'</td><td width=15
height=25></td><td>'.trim($row['prijmeni']).'
.trim($row['jmeno']).' </td><td class=sedy>'.$log.'</td>';
            endif;
        if ((trim($row['najemnik'])<> '0') and ($row['dluh']<> 0)):
            echo '<td><a href="adm.php?polozka='.$row['id'].'"><td>'.trim($row['byt']).'</td><td width=15
height=25></td><td>'.trim($row['prijmeni']).' '.trim($row['jmeno']).'
</td><td class=sedy>'.$log.'</td>';
            endif;
    }
mysql_close($link);
echo '</table>
<br>
<table align=center>
<tr><td></td><td>Dlužník do fondu oprav</td></tr>
<tr><td></td><td>Pronajímaná bytová jednotka</td></tr>
</table>';
endif;

```

Pozn: sloupec hodnota „last_login“, která se načítá do formuláře je pro administrátora pouze informativní údaj jako datum /čas, kdy se naposledy nájemník přihlašoval do

systemu SVJ. V případě, že tato položka není naplněna je zřejmé, že se uživatel, ještě do systému vůbec nepřihlašoval a nebo si nezažádal o přístup, viz. formulář registrace.

Při editaci nájemníka a provedení změn ve formuláři bylo provedeno po stisknutí tlačítka „uložit“ následujícího skriptu pro uložení dat do tabulky vlastníci:

```
//ulozeni editace -----  
-----  
if (($_GET['ulozit']==2) and (session_is_registered("logine"))):  
    $db=mysql_select_db("ic_podparkem",$link);  
    $sql="UPDATE vlastnici set jmeno='". $_POST[jmeno]."',  
prijmeni='". $_POST[prijmeni]."', najemnik='". $_POST[najemnik]."',  
dluh='". $_POST[dluh]."', osoby='". $_POST[osoby]."', last_login='". $_  
POST[login]."' where id='". $_GET[polozka];  
    $result=mysql_query($sql);  
    unset($_GET['polozka']);  
endif;
```

4.9 Statické stránky

Součástí tvorby stránek bylo požadavkem SVJ i interpretace statických stránek, které zobrazují pouze informativní charakter o kontaktech na havarijní službu, prezentaci domovního řád, stanov a odkazu na mapu, kde společenství SVJ sídlí. Představa Statických stránek byla vytvořena výběrem v dokumentu word. Následně se tento dokument uložil s příponou „.html“ a autorem byl poupraven v textovém editoru „pspad“.

V jednom případě odkazu na hlavních stránkách „kde nás najdete“ byl proveden pouhý hypertextový odkaz na stránky www.mapy.cz, kde se návštěvníkovi stránek zobrazila podrobnější mapa, kde se společenství SVJ nachází. Jednalo o programový „.html“ příkaz (syntaxe) následujícím způsobem:

```
<a  
ref="http://www.mapy.cz/#x=132210361@y=138063498@z=15@mm=ZP@sa=s@st=s@ssq  
=pod%20parkem%2028@sss=1@ssp=118854565_121948033_152245157_151684993"  
target="_blank">
```

4.10 Zabezpečení webových aplikací

4.10.1 Cross site scripting

Nazývá se též XSS neboli Cross-site Scripting, je to jedna z nejstarších metod a nastává, když se dynamické webové stránce podstrčí data uživatelem a je zobrazen výstup, při čemž není správně ověřen. Data mají většinou formu odkazů, který může mít v sobě škodlivý obsah a toto je pak šířeno všemi prostředky internetem., Jedná se o způsob kdy útočník pomocí XSS URL podstrčí do stránky vlastní skript a jeho zvládnutí pak kontroluje ve svém prohlížeči. Požadavkem samozřejmostí je, že útočník musí mít minimálně základy HTML, JavaScriptů a obecně dynamického skriptování na straně serveru. Nejvíce ohrožení mohou být stránky, které ukládají data do databáze, jako například formuláře webových stránek kdy nás, vyzívají „, o zapomenutém heslu“ apod. Podstrčením XSS může útočník získat soukromé informace, manipulovat s cookies nebo je odcizit nebo na systému koncového uživatele dokonce si spouštět škodlivý kód.

Nejlepším způsobem jak zjistit zda jsou web aplikace „imunní“ proti této technice je použitím Web Vulnerability Scanner. Prohlédne celý web a hledá chyby, které umožňují útok právě pomocí Cross site scriptingu a na které nás upozorní.

4.10.2 SQL injection

SQL Injection patří mezi hackerská techniky, při které se hacker pokouší webové stránce (aplikaci) vnutit SQL příkazy , které se spustí na databázovém serveru. Hacker vloží do formuláře pole místo požadovaného textu upravený příkaz, který je pak zpracován PHP skriptem a odešle se jako SQL dotaz do databáze. Poškození tímto útokem může být různé. Od celkem malé (při vnučení příkazu SELECT pro zobrazení obsahu databázových tabulek) po fatální (např. příkaz DELETE). U zvlášť zákeřných útoků může dojít ke změně nebo nahrání nových dat.

4.10.3 Directory Traversal

Tento typ útoku může se může vyskytnout z kterékoliv stránky webu, kde jsou uložena nějaká uživatelská data využívána pro činnost se souborovým systémem (formulářová pole nebo upload souboru). Když hacker do formulářového pole vloží

„,..../..../..../etc/passwd” a skript tento řetězec spojí za název adresáře, může být dost pravděpodobné, že může dojít k připojení souboru s hesly a ne požadovaného souboru. V nejhorších situacích je možné, že může docházet i k operacím se soubory (jako je například. přesun nebo mazání). [17]

4.11 Ekonomické výsledky

Vlastní nasazení webového serveru pro dynamické skriptování, může být využito několika způsoby, ten první může být takový, že server se umístí právě v domě společenství vlastníků jednotek, kde je provedena strukturovaná kabeláž po celém domě tří vchodů a je zde zajištěn přístup k poskytovateli internetu, přímo optickým kabelem, což je nejlepší varianta než připojování pomocí bezdrátového připojování k poskytovateli internetu. V případě, že by bylo zvoleno této varianty, je nutné si u poskytovatele zažádat o přiřazení veřejné pevné IP adresy z důvodů, aby bylo umožněno přístupu všem, kteří chtějí tyto webové stránky navštívit. V případě této varianty jsou investice následující:

Varianta č. 1

Fixní pořizovací náklady

Položka	Cena	Pozn.
Server Siemens TX150	26 900,-	
Operační systém Windows 7	4 000,-	
MySQL, Apache, PHP	0,-	WAMP server free balíček
UPS 750W	4350,-	
CELKEM	34 250,-Kč	

Tabulka 2 - fixní náklady

Měsíční náklady

Položka	Cena	Pozn.
Pronájem veřejné IP adresy	190,-Kč/měsíc	
Doména II. Řádu	250,-Kč/rok	

Tabulka 3 - měsíční náklady

Nevýhodou tohoto řešení jsou vyšší náklady na pořízení hardwaru a případné starosti v případě havárie serveru s pořizováním náhradních dílů na opravu jak HW, tak případnou reinstalaci softwaru jako je operační systém Apache, MySQL atd.

Varianta č. 2

Dalším řešením úspory fixních nákladů bylo vedení SVJ nabídnuto řešení, kdy se veškerý potřebný hardware/software tzv. pronajme přímo od poskytovatelů internetů, kteří nabízejí služby hostingu právě pro tyto účely. Tento způsob pronájmu je již velmi běžný. Je nutné jen vybrat vhodného poskytovatele hostingových služeb, který nabízí dynamické skriptování pomocí PHP. Tzn. jsou zde kladeny požadavky, aby hostingový server měl možnost:

- Přístup FTP
- Zabudované PHP
- Webový sever Apache
- MySQL databázi

Průzkumem trhu bylo zjištěno, že již velmi mnoho firem, kteří nabízejí hosting a poskytují výše uvedené požadavky dokonce i zdarma do určitého limitu „diskového prostoru“ případně naplněné databáze. Placené služby jsou v případě překročení řádově 100MB dat diskového prostoru což v našem případě ze začátku nebude překročeno, protože na disk se budou pouze ukládat zápisy z jednotlivých porad a jiné různé dokumenty, ale v budoucnu by mohl být tento limit překročen. Tato varianta bezplatného poskytování hostingu se nazývá „extra lite“.

Doporučením autora, je zaplatit za tuto službu 20,-Kč měsíčně a rozšířit tyto služby. Název této služby je „extra lite plus a nabízí: Kalkulace cen za první rok

Cena za hosting	12měsíců x20= 240,-Kč
Cena za doménu	0,-Kč
Cena celkem	240,-Kč

Tabulka 4 - první rok náklady

Cena na další roky

Cena za hosting	12měsíců x20= 240,-Kč
Cena za doménu	160,-Kč
Cena celkem	400,-Kč

Tabulka 5 - další roky náklady

V ceně této služby je zahrnuto:

- 4Gb prostoru FTP
- 1Gb dat pro MySQL možnost vytvořit až 10 databází
- PHP 5.3
- Dostupnost > 99,9%
- Periodické denní zálohy
- Vzdálená administrace a mnoho dalšího
- Statistiky přístupů

Společenství SVJ se rozhodlo právě pro tuto variantu, kterou nabízí firma HOSTING 90 s.r.o

Po zaplacení veškerých poplatků, byl zprovozněn přístup hostingovou společností a autor po té provedl zkopírování veškerých PHP skriptů pomocí FTP klienta na diskový prostor poskytovatele služeb. Naplněná databáze MySQL byla vyexportována pomocí PHP My Admina ze serveru Siemens TX150 do sql souboru a po té naimportována na nový MySQL server poskytovatele služeb.

PHP Open Source – Microsoft ASP

Investice na pořízení scriptovacího jazyka PHP jsou 0,-Kč v porovnání se s prostředím dynamického skriptování od společnosti Microsoft, kde jsou veškeré balíčky placené.

Níže jsou uvedeny základní potřebné balíčky pro dynamické skriptování na straně severu s použitím Open Souce řešení a řešení komerčního placeného za použití Windows 2012 serveru od společnosti Microsoft.

Jedná se pouze o orientační ceny

	PHP		Microsoft ASP	
Operační systém	Linux Fedora	0,-Kč	Windows 2012 server	9162,-Kč
Databázový systém	MySql	0,-Kč	MS SQL	V ceně
Web server	Apache	0,-Kč	ASP	V ceně
CELKEM		0,-Kč		9162,-Kč

Tabulka 6 - srovnání cen PHP a ASP

Autor tímto nechce znevýhodňovat cenovou politiku firmy Microsoft, ale snaží se nastínit, že pro prvotní programování dynamického skriptování na straně serveru není nutné vynaložit jakékoliv náklady na pořízení softwaru , který slouží k výše uvedeným účelům. Komerční softwarové produkty od společnosti Microsoft jsou placené, jsou zde vyšší náklady, ale na druhou stranu mají možná vyšší podporu a programování stránek je více pohodlnější.

5 Zhodnocení výsledků

Tato práce popsala základní seznámení s možnostmi využití jazyka PHP, ve vlastní praktické části bylo názorně předvedeno, jakým způsobem je možné využít skriptovacího jazyka PHP při vytváření stránek s dynamickým obsahem. Bylo ověřeno, že vlastní instalace potřebných balíčků pro tento skriptovací jazyk je velmi jednoduchá a není nijak náročná na hardware, který je potřebný pro využití funkcionalit tohoto skriptovacího jazyka. V případech kdy bylo zapotřebí vyplňovat i obsah formuláře na straně klienta, bylo nutností použít javascriptů z důvodu kontroly správného vyplnění. Bez této kontroly by byl formulář odeslán a zapsán do databáze bez jakýchkoliv údajů. Vlastní projekt navržení webových stránek s dynamickým skriptováním trvala přibližně 1 týden. Další den byla prováděna instalace softwaru, jako je operační systém a konfigurace Apache, PHP a MySQL. Po otestování a ověření veškeré funkcionality jednotlivých skriptů a přístupů na webové stránky se web server přenesl do komerčního prostředí k poskytovateli internetu(provider). Nebylo zde nutné manuálně vyplňovat veškeré údaje znovu do databáze, ale provedl se pouhý export/import MySQL databáze, zkopírování skriptů a dalších potřebných souborů pomocí FTP klienta. Potvrdilo se, že aplikace WampServer, je velmi dobrým pomocníkem při nasazování webových stránek se zaměřením na dynamické skriptování, neboť je velmi jednoduché otestovat a zprovoznit funkcionalitu stránek na běžné pracovní stanici a až po otestování kompletní řešení nasadit do komerčního prostředí. Bylo zjištěno, že poskytovatele hostingu nabízejí téměř stejné služby, ale za zcela jiných cenových podmínek, někteří nabízejí hosting zdarma s omezenou kapacitou diskového prostoru, jiní s menším měsíčním poplatkem, ale za to s rozšířeným diskovým prostorem a možností vytváření několika databází. V našem případě postačila pouze jedna databáze s několika tabulkami. V průběhu instalace byly zjištěny i drobné neduhy softwarového balíčku „Wamp Server“ a to, že když je tato služba spuštěná (běží PHP, MySQL, Apache), tak není možné využívat softwaru pro komunikaci „skype“. Autor se domnívá, že se nejedná až tak o závadu „softwarový bug“, ale zřejmě jedna z aplikací Wamp Serveru“ využívá pro komunikaci stejného UDP portu, který je používán softwarem „skype“.

6 Závěr

Existuje mnoho instalačních balíčků, které dokážou nainstalovat veškeré komponenty potřebné pro dynamické skriptování na straně serveru, tak jako software WampServer dalším v pořadí je například Xampp, který umožňuje nainstalovat dokonce o něco více komponent. V základu tohoto balíčku jsou, Apache 2.2.14, MySQL 5.1.41, php 5.3.1 PHPMyAdmin 3.2.4, Pearl 5.10.1, FileZilla FTP server 0.9.33 a Mercury Mail Transport System verze 4.2. Projekt implementace webových stránek SVJ byl vytvořen jednoduchými PHP skripty pro základní dotazování MySQL databáze s použitím skriptů pracujících na straně klienta pomocí „javaskriptů“ Autor nabyl z implementovaného projektu zkušeností a zvažuje webové stránky předělat do prostředí PHP verze 6, kde tato verze již nabízí objektově orientované programování což , ale sebou obnáší hlubší znalosti problematiky PHP jazyka. V průběhu implementace a studia jazyka PHP autor zjistil, že existuje mnoho dalších způsobů jak vytvářet podobné webové stránky a to například formou „redakčních systémů“, které ve vlastním důsledku poskytují podobné vlastnosti a funkce, kterých se autor snažil vytvořit vlastními základními znalostmi. Instalace redakčního systému je velmi jednoduchá a nabízí sofistikovanější a přehlednější formu prezentace aktuálních informací na jakémkoliv téma. Jedním z nich může být právě informování vlastníků bytových jednotek o ekonomickém a kulturním dění v domě ve kterém bydlí. Realizace projektu programování webových stránek SVJ byla autorovo první instalací, kde se potýkal s prvotními programátorskými problémy jak se napojovat pomocí jazyka PHP do databáze, jakým způsobem provádět exporty importy dat tabulek nebo jakým způsobem provést vlastní migraci dat do komerčního prostředí webového hostingu poskytovatele internetu. Z důvodů ochrany osobních dat jsou některé informace o uživatelích bytových jednotek smyšlené jako například nedoplatky seznam vlastníků bytových jednotek nebo nedoplatky nájemníků prezentovaných na obrázcích. Závěrem lze konstatovat, že stránky jsou i nadále využívány společenstvím vlastníků jednotek k poskytování informací o stavu SVJ. Vzhledem k úspěšnosti a návštěvnosti stránek je v současnosti rozpracován další část projektu na administraci stránek SVJ.

7 Seznam použitých zdrojů

1. **Boroznyk, Timothy.** *PHP6, MySQL, Apache vytváříme webové aplikace.* Brno : CPRESS, 2009. ISBN 978-80-251-2767-4.
2. **Lecky-Thomson, Ed and D. Nowicki, Steven.** *PHP 5 programujeme profesionálně.* Brno : CPRESS, 2009. ISBN 978-80-251-3127-5.
3. **Bláza, Jiří.** *PHP 5 začínáme programovat .* Praha : GRADA, 2005. 80-247-1146-X.
4. **Jakub, Vrána.** *1001 tipů a triků pro PHP.* Praha : COMPUTER PRESS, 2011. 9788024711461.
5. **Ögg, Bernd and Kofler, Michael.** *PHP 5 a MySQL 5.* Praha : COMPUTER PRESS, 2007. 9788025118139.
6. *Jak psát web.* [Online] [Cited: 08 13, 2013.] <http://www.jakpsatweb.cz/>.
7. *JavaScript pro webové vývojáře.* Brno : COMPUTER PRESS, 2009. 9788025125090.
8. **Bellinaso, Marco.** *Webové programování v ASP.NET 2.0.* Brno : COMPUTER PRESS, 2013. 9788025118931.
9. **Mikul'ák, Martin.** *Programování WWW stránek pro úplné začátečníky.* Brno : COMPUTER PRESS, 2011. 9788025132524.
10. **Miroslav, Cyroň.** *CSS – kaskádové styly.* Praha : GRADA, 2005. 80-247-1420-5.
11. **Slavoj, Písek.** *html.* Praha : GRADA, 2010. 978-80-247-3117-9.
12. **Oliver, Leiss and Jasmin, Schmidt.** *PHP v praxi.* Praha : GRADA, 2010. 978-80-247-3060-8.
13. Linuxsoft. [Online] [Cited: 08 31, 2013.] <http://www.linuxsoft.cz/php/>.
14. PHP triky. [Online] [Cited: 06 18, 2013.] <http://php.vrana.cz/>.
15. Acunetix. [Online] [Cited: 05 16, 2013.] <http://www.acunetix.com/>.
16. Wamp Server. [Online] [Cited: 01 13, 2013.] <http://www.wampserver.com/en/>.
17. **Allen, Harpen.** *Hacking–manuál hackera* Praha : GRADA, 2008. 978-80-247-1346-5.
18. *Apache documentation.* [Online] [Cited: 10 12, 2013.] <http://httpd.apache.org/docs/2.4/>.
19. *PHP documentation.* [Online] [Cited: 05 14, 2013.] <http://php.net/docs.php>.

Seznam obrázků

Obrázek 1 - bloková interpretace PHP	20
Obrázek 2 - testovací PC	26
Obrázek 3 - instalace WampServer	30
Obrázek 4 - menu WampServer.....	31
Obrázek 5 - úspěšně nainstalovaný WampServer.....	32
Obrázek 6 - vytvoření tabulky v PHP MyAdmin	35
Obrázek 7 - zobrazení struktury tabulky.....	36
Obrázek 8 - přihlašovací okno	39
Obrázek 9 - zobrazení přihlášeného uživatele	40
Obrázek 10 - registrační formulář.....	41
Obrázek 11 - okno nevyplněné informace	42
Obrázek 12 - formulář pro změnu hesla	42
Obrázek 13 - připomínky a náměty	45
Obrázek 14 - seznam vlastníků SVJ	49
Obrázek 15 - stav financí	50
Obrázek 16 - zápisy z porad SVJ.....	52
Obrázek 17 - nástěnka.....	56

Seznam tabulek

Tabulka 1 - seznam editorů.....	24
Tabulka 2 - fixní náklady	61
Tabulka 3 - měsíční náklady	61
Tabulka 4 - první rok náklady.....	62
Tabulka 5 - další roky náklady	62
Tabulka 6 - srovnání cen PHP a ASP	63

8 Přílohy

Ukázka webových stránek, společenství vlastníků jednotek, které využívají PHP (dynamického skriptování na straně serveru) a částečně java skriptů popsaných v diplomové práci

The screenshot shows a web page for a housing cooperative (SVJ) with the following elements:

- Header:** "Pod parkem 2563-2565, Ústí nad Labem, 400 12" and "SVJ Pod parkem 2563-2565". A user status indicator says "Nejsi přihlášený".
- Left Navigation Menu:** OBSAH, SVJ-KONTAKTY, DOMOVNÍ ŘÁD, STANOVY, ZÁPISY SCHŮZÍ SVJ, FINANCE, HLÁŠENÍ OPRAV, ARCHIV NÁSTĚNKY, DŮLEŽITÉ KONTAKTY.
- Main Content Area:**
 - 16.1.2012:** "Zpráva ve věci výtahových strojů." "Dobrý den paní Sedláková, ... [celý článek...](#) Tímto e-mailem Vám chci sdělit následující zprávu."
 - 1.7.09 domovní řád:** "Prosíme nájemníky, aby dodržovali domovní řád, který byl schválen shromážděním. Pravidla [domovního řádu](#) jsou k nahlédnutí v levé části obrazovky menu **OBSAH:**"
 - 20.4.08 Info:** "Vážení vlastníci bytových jednotek, vítáme vás na stránkách SVJ Pod Parkem 2563-2565. Na těchto stránkách vás rádi budeme informovat o aktuálním dění SVJ. Budete mít zde možnost napsat vaše připomínky a náměty, které nám mohou pomoci ke zlepšení našeho SVJ. Budou zde k dispozici aktuální zápisy ze schůze shromáždění, které si bude moci každý vlastník bytové jednotky stáhnout. Výbor společenství ... [celý článek...](#)"
- Right Side:**
 - Přihlášení:** Fields for "Uživatel:" and "Heslo:" with an "OK" button. Below it: "Zadej své příjmení pro přístup do systému" and "--registrace--".
 - ANKETA:** "Jste spokojeni s novými výtahy?" with a progress bar for "ANO" (14) and "NE" (1), and a "hlasuj" button.
 - Subscription:** "máte zájem o zaslání odpovědí na dotazy pomocí e-mailu? napište svůj e-mail: email@email.cz" with an "OK" button.
 - Reminders:** "zde zasílejte vaše připomínky:" followed by a text input area.
- Bottom Section:** "PŘIPOMÍNKY A NÁMĚTY:" with a list of recent posts:
 - 15/09/13 11:28:25 [Dobrý den, chtěla bych se zeptat jak to bude s úkl...](#) (1) **NEW**
 - 17/04/11 16:06:28 [Dobrý den, Paní Sedláková, chtěl bych se zeptat jak ...](#) (1)
 - 14/02/11 13:08:19 [Úklid společných prostor Vašeho domu za pouhých 50...](#) (2)
 - 05/09/10 19:25:56 [Přeji hezký den! ...](#) (3)
 - 18/10/09 14:20:06 [Prosím o sdělení datumu zahájení a ukončení rekons...](#) (5)
 - 24/04/09 19:06:12 [Dobrý den! Za odeslání odpovědi na můj případný dot...](#) (1)
- Footer:** "Počet návštěvníků: 1580" and navigation icons for "Další", "Předchozí", "Zvýraznit", and "Rozlišovat velikost".

SVJ Pod parkem 2563-2565

Pod parkem 2563-2565
Ústí nad Labem
400 12

Přihlášený uživatel:

OBSAH:

SVJ-KONTAKTY

DOMOVNÍ ŘÁD

STANOVY

ZÁPISY SCHŮZÍ SVJ

FINANCE

HLÁŠENÍ OPRAV

ARCHIV NÁSTĚNKY

DŮLEŽITÉ KONTAKTY

01-5.Zapisy z vyboru.doc	06.Zapis z vyboru ze dne 18.12.2007.doc
07.Zapis z vyboru ze dne 14.1.2007.doc	08.Zapis z vyboru ze dne 21.1.2008.doc
09.Zapis z vyboru ze dne 28.1.2008.doc	10.Zapis z vyboru ze dne 30.1.2008.doc
11.Zapis z vyboru ze dne 5.2.2008.doc	13.Zapis z vyboru ze dne 13.2.2008.doc
14.Zapis z vyboru ze dne 18.2.2008.doc	15.Zapis z vyboru ze dne 25.2.2008.doc
16.Zapis z vyboru ze dne 3.3.2008.doc	17.Zapis z vyboru ze dne 10.3.2008.doc
18.Zapis z vyboru ze dne 17.3.2008.doc	19.Zapis z vyboru ze dne 27.3.2008.doc
20.Zapis z vyboru ze dne 7.4.2008.doc	21.Zapis z vyboru ze dne 14.4.2008.doc
22.Zapis z vyboru ze dne 28.4.2008.doc	23.Zapis z vyboru ze dne 19.5.2008.doc
24.Zapis z vyboru ze dne 26.5.2008.doc	25.Zapis z vyboru ze dne 2.6.2008.doc
26.Zapis z vyboru ze dne 9.6.2008.doc	27.Zapis z vyboru ze dne 16.6.2008.doc
28.Zapis z vyboru ze dne 23.6.2008.doc	29.Zapis z vyboru ze dne 30.6.2008.doc
30.Zapis z vyboru ze dne 7.7.2008.doc	31.Zapis z vyboru ze dne 14.7.2008.doc
32.Zapis z vyboru ze dne 21.7.2008.doc	33.Zapis z vyboru ze dne 28.7.2008.doc
34.Zapis z vyboru ze dne 4.8.2008.doc	35.Zapis z vyboru ze dne 11.8.2008.doc
36.Zapis z vyboru ze dne 25.8.2008.doc	37.Zapis z vyboru ze dne 8.9.2008.doc
38.Zapis z vyboru ze dne 15.9.2008.doc	39.Zapis z vyboru ze dne 22.9.2008.doc
40.Zapis z vyboru ze dne 29.9.2008.doc	Mimoradny Zapis ze shromazdeni ze dne 20.3.2008.doc
Zapis ze shromazdeni ze dne 27.11.2007.doc	Zapis ze shromazdeni ze dne 31.1.2008.doc
41.Zapis z vyboru ze dne 6.10.2008.doc	60.Zapis z vyboru ze dne 18.5.2009-.doc
62.Zapis z vyboru ze dne 22.6.2009.doc	43.Zapis z vyboru ze dne 20.10.2008.doc
Mimoradny zapis ze dne 16.10.2008.doc	44.Zapis z vyboru ze dne 3.11.2008.doc
45.Zapis z vyboru ze dne 10.11.2008.doc	46.Zapis z vyboru ze dne 24.11.2008.doc
47.Zapis z vyboru ze dne 1.12.2008.doc	48.Zapis z vyboru ze dne 8.12.2008.doc
49.Zapis z vyboru ze dne 15.12.2008.doc	OPRAVA-Zapis ze shromazdeni ze dne 2.12.2008.doc
50.Zapis z vyboru ze dne 12.1.2009.doc	51.Zapis z vyboru ze dne 19.1.2009.doc
52.Zapis z vyboru ze dne 26.1.2009.doc	53.Zapis z vyboru ze dne 16.2.2009.doc
54.Zapis z vyboru ze dne 2.3.2009.doc	55.Zapis z vyboru ze dne 9.3.2009.doc
56.Zapis z vyboru ze dne 23.3.2009.doc	57.Zapis z vyboru ze dne 20.4.2009.doc
61.Zapis z vyboru ze dne 1.6.2009.doc	Mimoradny zapis ze shromazdeni ze dne 8.4.2009.doc
58.Zapis z vyboru ze dne 27.4.2009.doc	59.Zapis z vyboru ze dne 11.5.2009-.doc
70.Zapis z vyboru ze dne 21.12.2009.doc	63.Zapis z vyboru ze dne 17.8.2009.doc
64.Zapis z vyboru ze dne 14.9.2009.doc	65.Zapis z vyboru ze dne 21.9.2009.doc
66.Zapis z vyboru ze dne 12.10.2009.doc	67.Zapis z vyboru ze dne 26.10.2009.doc
Zapis ze shromazdeni ze dne 1.10.2009.doc	68.Zapis z vyboru ze dne 2.11.2009.doc
69.Zapis z vyboru ze dne 23.11.2009.doc	71.Zapis z vyboru ze dne 4.1.2010.doc
72.Zapis z vyboru ze dne 15.2.2010.doc	72.Zapis z vyboru ze dne 1.3.2010.doc




[Kde nás najdete](#)

Počet návštěvníků: 1580

Odhlášení [změna hesla](#)

Uživatel:

Heslo:

Zadej své příjmení pro přístup do systému

--registrace--

ANKETA

Jste spokojeni s novými výťahy?

ANO 14

NE 1

máte zájem o zaslání odpovědi na dotazy pomocí e-mailu?
napíšte svůj e-mail:

zde zasílejte v

Další [↑](#) [Předchozí](#) [Zvýraznit](#) [Rozlišovat velikost](#)

☐ Sázení na sport ☐ Originální dárky ☐ email client ☐ Doména CZ za 125 Kč/rok ☐ nábytek -40% ☐ Dedikovaný server za 450 Kč ☐ Stahování souborů ZDARMA!

Pod parkem 2563-2565
Ústí nad Labem
400 12

SVJ Pod parkem 2563-2565

Nejsi přihlášený

OBSAH:
SVJ-KONTAKTY
DOMOVNÍ ŘÁD
STANOVY
ZÁPISY SCHŮZÍ SVJ
FINANCE
HLÁŠENÍ OPRAV
ARCHIV NÁSTĚNKY
DŮLEŽITÉ KONTAKTY



[Kde nás najdete](#)

Počet návštěvníků: 1580

nejsi přihlášen!

Přihlášení:
Uživatel:
Heslo:
OK
Zadej své příjmení pro přístup do systému
--registrace--

ANKETA
Jste spokojeni s činností výboru?

ANO 23
 NE 3

hlasuj

máte zájem o zaslání odpovědí na dotazy pomocí e-mailu?
napíšte svůj e-mail:

OK

zde zasilejte vaše připomínky:

odeslat

ance

Další ↑ Předchozí ↩ Zvýraznit ☐ Rozlišovat velikost

9 Zkratky a vysvětlivky

SVJ	společenství vlastníků jednotek
MySQL	freewarová databáze
PHP	skriptovací jazyk
Apache	webový server
Javascript	skriptovací jazyk
Web hosting	pronájem prostoru u poskytovatele internetu
HTML	Hyper text markup language
http	hypertext transfer protokol
FTP	File transfer protokol