

# WWW stránky pro kurz CŽV na FAI

CŽV course web page

Richard Vytásek



Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav aplikované informatiky

akademický rok: 2008/2009

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Richard VYTÁSEK**  
Studijní program: **B 3902 Inženýrská informatika**  
Studijní obor: **Informační technologie**  
  
Téma práce: **WWW stránky pro kurz CZV na FAI**

Zásady pro vypracování:

1. Vytvořte layout (grafický návrh) celé aplikace. Využívejte technologie PHP a MySQL.
2. Navrhněte a realizujte systém přístupových práv do aplikace.
3. Realizujte moduly řešící jednotlivé činnosti zadané vedoucím bakalářské práce jako jsou: novinky na titulní straně, správa učebních textů, správa rozvrhů, správa frekventantů a absolventů jednotlivých běhů, administrativa.
4. Naplňte hotovou aplikaci daty z kurzů organizovaných v předešlých letech.

Rozsah práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. Mach, J.: PHP pro úplné začátečníky, Computer Press, Praha 2002.
2. Kosek, J.: PHP – tvorba interaktivních internetových aplikací, Grada Publishing 1999.
3. J. Castagnetto, H. Rawat, S. Schumann, Ch. Scollo, D. Veliath: PHP – programujeme profesionálně, Computer Press, Praha 2002.
4. Ullman L.: PHP a MySQL – Názorný průvodce tvorbou dynamických WWW stránek, Computer Press, červen 2004.
5. Škultéty, R.: JavaScript – Kapesní přehled, Computer Press, leden 2006.
6. Prokop, M.: CSS kaskádové styly pro webdesignéry (2. vydání), Computer Press, únor 2005.

Vedoucí bakalářské práce:

**Ing. Tomáš Sysala, Ph.D.**

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

**20. února 2009**

Termín odevzdání bakalářské práce:

**1. června 2009**

Ve Zlíně dne 13. února 2009



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



doc. Ing. Ivan Zelinka, Ph.D.  
*ředitel ústavu*

## **ABSTRAKT**

Cílem této bakalářské práce je vytvoření webové aplikace pro správu studentů, studijních materiálů a novinek pro kurz Celoživotního vzdělávání – Studia k rozšíření odborné kvalifikace, jenž probíhá na Fakultě aplikované informatiky, Univerzitě Tomáše Bati ve Zlíně.

Klíčová slova: Celoživotní vzdělávání, www, css, php, Mysql, Javascript, Zend framework

## **ABSTRACT**

The goal of this bachelor project is to create web application for managing students, educational materials and news for the CŽV course, which takes place at Faculty of Applied Informatics on Tomáš Baťa University in Zlín.

Keywords: Lifetime education, www, css, php, Mysql, Javascript, Zend framework

Rád bych tímto poděkoval vedoucímu mé práce panu Ing. Tomáši Sysalovi Ph.D. za jeho čas a ochotnou pomoc při vypracování této bakalářské práce.

Prohlašuji, že

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

Prohlašuji,

že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval.

V případě publikace výsledků budu uveden jako spoluautor.

Ve Zlíně

.....  
podpis diplomanta

**OBSAH**

<b>ÚVOD</b> .....	<b>9</b>
<b>I TEORETICKÁ ČÁST</b> .....	<b>10</b>
<b>1 HISTORIE WWW</b> .....	<b>11</b>
<b>2 TECHNOLOGIE POUŽÍVANÉ PŘI TVORBĚ WWW STRÁNEK</b> .....	<b>13</b>
2.1 LAMP ARCHITEKTURA:.....	13
2.1.1 Linux .....	13
2.1.2 Apache HTTP Server .....	14
2.1.3 MySQL.....	14
2.1.4 Php.....	15
2.2 DALŠÍ TECHNOLOGIE: .....	16
2.2.1 Javascript.....	16
2.2.2 Css .....	16
2.2.2.1 Syntaxe.....	17
<b>3 MVC FRAMEWORKY</b> .....	<b>18</b>
3.1 HISTORIE .....	18
3.2 VRSTVY.....	18
3.2.1 Model .....	18
3.2.2 View: .....	19
3.2.3 Controller .....	19
3.3 PRINCIP .....	19
3.4 PRAVIDLA A VAZBY.....	19
3.5 MVC A WEB .....	20
3.6 VYBRANÉ PHP MVC FRAMEWORKY .....	20
3.6.1 CakePHP .....	20
3.6.2 Qcodo .....	20
3.6.3 Nette .....	21
3.6.4 Zend Framework .....	21
<b>4 ZEND FRAMEWORK</b> .....	<b>22</b>
4.1 ADRESÁŘOVÁ STRUKTURA .....	22
4.1.1 Application.....	22
4.1.2 Library .....	22
4.1.3 Public.....	23
4.2 TVAR ADRESY .....	23
4.3 DŮLEŽITÉ SOUBORY: .....	24
4.3.1 Index.php.....	24
4.3.2 Bootstrap.php .....	25
<b>5 POUŽITÝ SOFTWARE</b> .....	<b>26</b>

5.1	PSPAD EDITOR 4.5.3 .....	26
5.2	MOZILLA FIREFOX 3.0.X.....	26
5.3	WEB DEVELOPER TOOLBAR PLUGIN 1.1.6.....	26
5.4	EASYPHP 2.0.0.0 .....	26
5.5	IRFANVIEW 4.20.....	27
5.6	GIMP 2.4.6.....	27
<b>II</b>	<b>PRAKTICKÁ ČÁST .....</b>	<b>28</b>
<b>6</b>	<b>POPIS APLIKACE .....</b>	<b>29</b>
6.1	DESIGN.....	29
6.2	DATABÁZE .....	30
6.2.1	Popis jednotlivých tabulek .....	30
6.2.1.1	Tabulka users .....	30
6.2.1.2	Tabulka categories .....	31
6.2.1.3	Tabulka files .....	31
6.2.1.4	Tabulka news .....	31
6.2.1.5	Tabulka pages .....	32
6.2.1.6	Tabulka students .....	33
6.2.1.7	Tabulka settings .....	34
6.2.2	Přístup k databázi přes model .....	34
6.3	STRUKTURA APLIKACE .....	34
6.3.1	ErrorController .....	34
6.3.2	AuthController .....	35
6.3.3	IndexController .....	35
6.3.4	EditController.....	35
6.3.5	AdminController .....	35
6.4	UŽIVATELSKÉ ROZHRANÍ.....	35
6.4.1	Stránky přístupné všem .....	35
6.4.2	Stránky přístupné registrovaným (studentům) .....	37
6.4.3	Stránky přístupné učitelům.....	37
6.4.4	Stránky pro administraci .....	38
	<b>ZÁVĚR .....</b>	<b>40</b>
	<b>ZÁVĚR V ANGLIČTINĚ.....</b>	<b>41</b>
	<b>SEZNAM POUŽITÉ LITERATURY .....</b>	<b>42</b>
	<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK .....</b>	<b>43</b>
	<b>SEZNAM OBRÁZKŮ .....</b>	<b>44</b>
	<b>SEZNAM TABULEK.....</b>	<b>45</b>



## ÚVOD

Celoživotní vzdělávání je jednou z forem získávání odborné způsobilosti v rámci zákonné činnosti vysokých škol ve smyslu §60 zákona č.111/1998 Sb. o vysokých školách. Kvalifikovaní učitelé tímto studiem získávají podle §24, odst. 4 a 5, zákona 563/2004Sb. o Dalším vzdělávání pedagogických pracovníků a §6, odst. 1b vyhlášky č 317/2005 způsobilost k výuce předmětů zaměřených na informační technologie a informatiku k předcházející odborné a pedagogické způsobilosti učitelů základních nebo středních škol.

V současné době je pro podporu výuky tohoto kurzu dostupný web na adrese <http://czv.wz.cz/>, jedná se však o statickou stránku, což ústí v řadu nevýhod. Prvním nedostatkem je, že každá aktualizace webu vyžaduje zásah administrátora do zdrojového kódu na serveru, což může být časově neefektivní. Další nevýhoda tohoto způsobu se projeví, pokud potřebuje nějaké úpravy provést učitel. Za současného stavu nemůže změny provést sám, ale musí kontaktovat administrátora.

Tématem této bakalářské práce je naprogramovat aplikaci s víceuživatelským přístupem, jež umožňuje učitelům vkládat novinky a studijní materiály a administrátorovi správu předmětů a studentů.

## **I. TEORETICKÁ ČÁST**

## 1 HISTORIE WWW

WWW je zkratkou pro World Wide Web, zkráceně označované pouze jako Web (v překladu pavučina). Jedná se o soustavu vzájemně propojených dokumentů adresovaných pomocí URL (Uniform Resource Locator) odkazujícího na jednotlivé servery. Webové stránky mohou obsahovat text, obrázky, videa a řadu dalších multimedií. K jejich prohlížení slouží webový prohlížeč. Počátky tohoto mladého média sahají do roku 1989, kdy britský fyzik Sir Tim Berners-Lee při práci pro Evropskou organizaci pro jaderný výzkum (CERN) v Ženevě navrhl systém vzájemně provázaných hypertextových dokumentů, v němž pro procházení mezi stránkami stačí pouze "klikat" na odkazy. Myšlenka hypertextových odkazů nebyla úplně nová, inspiraci našel Lee ve starších projektech (např. Xanadu od Teda Nelsona). Ačkoli původním záměrem bylo pouze vzájemné sdílení informací mezi vědci, systém se díky své jednoduchosti u počítačové komunity brzy uchytil. Již v roce 1990 spatřil světlo světa první webserver a internetový prohlížeč a editor v jednom, jednoduše nazvaný WorldWideWeb. Roku 1993 prohlásil CERN tento systém jako volný pro všechny a bez poplatků, čímž systém získal obrovskou výhodu oproti konkurenčnímu Gopheru. Systém dále rostl neuvěřitelným tempem a objevily se první grafické prohlížeče Midas a Viola, které ale neuspějí v konkurenci výtečného produktu NCSA Mosaic autorů Andreessena a Erica Biny a Marca Andreessena, jenž položil základ všem moderním prohlížečům. Marc Andreessen brzy založil vlastní společnost Netscape s nejvýznamnějším produktem Netscape Navigátor, jenž ovládl asi 90 procent trhu. Na toto zareagovala firma Microsoft, která roku 1995 představila vlastní prohlížeč Internet Explorer, rovněž inspirovaný Mosaicem. Microsoft integroval Internet Explorer do operačního systému Windows čímž dosáhl téměř absolutní dominance na trhu. Psal se rok 1995 a web byl již poměrně rozšířeným médiem, kdy počet serverů přesáhnul 100 tisíc. Trendem posledních let se stalo upravování webových standardů tak, aby web mohl zpracovávat všemožné typy a formáty dat. Web tak nyní zdatně konkuruje ostatním médiím, umožňuje nám číst zpravodajství, poslouchat hudbu, sledovat online videa, chatovat s přáteli a řadu dalších možností. Web je dnes hlavním proudem internetu, čehož důkazem je i to, že pro spoustu laiků se internet automaticky rovná web.

Budoucnost webu je spojována s označením Web 2.0, kdy webové stránky přestávají být pouhým izolovaným zdrojem informací, ale začínají obsahovat i rozsáhlou funkcionalitu. Již dnes existuje řada webových aplikací, do značné míry nahrazující jejich desktopové protějšky. Příkladem lze jmenovat kancelářské aplikace či instant messaging. Dalším výrazným trendem poslední doby jsou sociální sítě umožňující sdílení obsahu mezi uživateli. [2]

## 2 TECHNOLOGIE POUŽÍVANÉ PŘI TVORBĚ WWW STRÁNEK

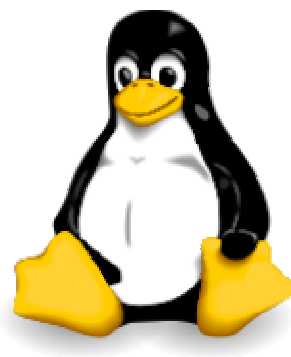
### 2.1 LAMP architektura:

LAMP je akronym pro soubor opensource software používaného pro Webové servery. Každé jednotlivé písmeno vyjadřuje jeden software:

- Linux – operační systém
- Apache – webový server
- MySQL – databázový server
- P bývá jeden z těchto skriptovacích jazyků: Perl, PHP, Python

Přestože autoři těchto programů nezamýšleli jejich vzájemné použití, tato kombinace se stala velmi oblíbenou díky nízkým pořizovacím nákladům a nenáročné instalaci těchto technologií. Tento akronym byl prvně použit roku 1998. [2]

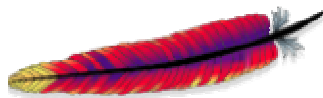
#### 2.1.1 Linux



*Obr.1:Logo Linux*

Linux je označení pro bezesporu nejznámější opensource operační systém. Jedná se o kompletní unixový systém vyvíjený pod licencí GNU GPL od roku 1990. Původně ho začal psát finský student helsinské univerzity Linus Torvalds jako svůj koníček. Postupem času se vývojový tým rozrůstal o řadu nadšenců a postavila se za něj řada společností. Dnes Linux patří mezi nejčastěji nasazovaný operační systém na serverech. [2]

### 2.1.2 Apache HTTP Server



*Obr.2:Logo Apache*

Apache je webový server vyvíjený od roku 1994 pod záštitou společnosti Apache Software Foundation. Tento software je s pokrytím přes 46% trhu nejoblíbenější webserver, jako první roku 2009 překonal mezník v nasazení na 100 milionech serverů. Je dostupný pro celou škálu operačních systémů zahrnující: Unix, GNU, FreeBSD, Linux, Solaris, Novell NetWare, Mac OS X, Microsoft Windows, OS/2, TPF a eComStation. [2]

### 2.1.3 MySQL



*Obr.3:Logo MySQL*

MySQL je RDBMS (relational database management system) systém s více než 11 miliony instalací. Program běží jako víceuživatelský server umožňující běh několika nezávislých databází. Licence tohoto produktu je GNU GPL, jde tedy o volně dostupné opensource. Nasazení tohoto databázového systému není žádná vzácnost, ve spojení s php běží i na velmi rozsáhlých webových aplikacích (např. google). Velkou výhodou je platformní nezávislost.

V současné době, kdy se jedná o odkup společnosti Sun, jenž zaštituje vývoj tohoto systému, firmou Oracle, jež disponuje vlastním databázovým systémem, vyvstává řada otazníků ohledně budoucnosti tohoto software. [2]

### 2.1.4 Php



*Obr.4:Logo PHP*

Php je rekursní akronym pro „PHP hypertext preprocessor“, což znamená, že v plném znění se tato zkratka vyskytuje znovu. Takovéto zkratky nejsou v počítačové branži neobvyklé (např. GNU, LAME, RPM). Jedná se skriptovací jazyk navržený původně pro vytváření dynamických webových stránek. Postupem času však přibylo i rozhraní příkazové řádky, které může být použito i v klasických aplikacích. Vývoj php byl započat roku 1994 Rasmusem Lerdorfem a dnes je zaštitěn organizací The PHP Group. Jde o volně šiřitelný software vydávaný pod PHP Licencí, která není kompatibilní s GPL z důvodu silných restrikcí použití slova „PHP“ v odvozených produktech. Php je nainstalován na více než jednom milionu webových serverů. Mezi jeho přednosti patří zejména platformní nezávislost a kvalitní dokumentace. [2]

Verze	Rok	popis
1.0	1995	První oficiálně představená verze
2.0	1996	Umožňovala vkládat html kód. Syntaxe podobná jazyku Perl.
3.0	1998	Vyvíjena týmem namísto jednoho člověka jako předešlé. Komplet předělána
4.0	2000	Přidání pokročilého dvouúrovňového parsovacího systému Zend Engine
5.0	2004	Zend Engine II s objektovým modelem

*Tab.1:Historie php*

## 2.2 Další technologie:

### 2.2.1 Javascript

Javascript je skriptovací jazyk používaný na mnoha webových stránkách. Na rozdíl od php neběží na webovém serveru, ale přímo u klienta v prohlížeči, čímž umožňuje bezprostředně reagovat na různé události. Tento jazyk vychází z mnoha programovacích jazyků, důraz při jeho vývoji byl kladen na jednoduchost a srozumitelnost i pro méně pokročilé programátory. I přes svůj název nijak nesouvisí s jazykem Java. Javascriptové kódy se na webu objevují už od roku 1995, kdy byla jeho podpora oficiálně implementována do prohlížeče Netscape Navigátor. [2]

### 2.2.2 Css

Css je zkratka Cascading Style Sheets neboli Kaskádové styly. Jde o speciální jazyk umožňující definovat vzhledové vlastnosti dokumentů, nejčastěji HTML a XHTML. Hlavní výhodou je oddělení vzhledu a obsahu dokumentu, což ústí v řadu výhod:

- přehlednost kódu
- širší možnosti stylování
- možnost sdílení jednoho stylu pro více stránek
- není nutné opakovat pro každý prvek ty stejné tagy pořád dokola
- omezení velikosti přenášených dat
- možnost definovat více stylů pro stejnou stránku (např. pro tisk, zobrazení na mobilních zařízeních s malým displayem)

Jediným problémem je špatná podpora těchto standartů některými prohlížeči, avšak poslední dobou se díky konkurenci na trhu prohlížečů situace vyvíjí správným směrem. Jazyk byl navržen standardizační organizací W3C, autorem prvotního návrhu byl Håkon Wium Lie. Byly vydány zatím dvě úrovně specifikace CSS1 a CSS2, dokončuje se revize CSS 2.1 a pracuje se na verzi CSS3. V dnešní době je použití kaskádových stylů na každé rozsáhlejší webové prezentaci téměř nutností. [2]



### 2.2.2.1 *Syntaxe*

Jedna položka (pravidlo) obsahuje následující informace:

- Selektor, což je vlastně název ke kterému se dané pravidlo vztahuje. Může být standardní část html stránky nebo námi definovaný, který se volá parametry class či id. Námi definované styly začínají tečkou pro class a mřížkou pro id. Rozdíl spočívá v tom, že prvky označené mřížkou se podle pravidel v dokumentu vyskytují pouze jedenkrát, zatímco prvky označené tečkou se v dokumentu mohou opakovat libovolněkrát.
- Jedna nebo více deklarácí ovlivňujících výsledný vzhled stránky. Tyto jsou vždy umístěny ve složených závorkách a oddělovány středníkem. Deklarace obsahují jejich název a za dvojtečkou pak hodnotu.

### 3 MVC FRAMEWORKY

Model-View-Controller je softwarová architektura, jejímž hlavním cílem je oddělení uživatelského rozhraní řídicí logiky a datového modelu tak, aby změna v jedné části měla co nejmenší dopad na části ostatní. Tato architektura je někdy označována jako Model-2, existuje i architektura Model-1 oddělující pouze datový model a řídicí logiku (má tedy pouze dvě komponenty). Důvod tohoto oddělení spočívá zejména ve snadnější údržbě kódu u dlouhodobých projektů.

Webový aplikační framework je softwarový framework sloužící k tvorbě dynamických stránek, webových aplikací, či webových služeb. Cílem frameworků je usnadnit často prováděné činnosti konané při vývoji webů pomocí knihoven vykonávajících potřebné funkce, nejčastěji například přístup k databázím, správa sessions nebo validace formulářů. [2] [6]

#### 3.1 Historie

Termín MVC byl prvně použit roku 1979 Trygve Reenskaugem v publikaci Applications Programming in Smalltalk-80: How to use Model-View-Controller. Postupem času vzniklo několik odvozených architektur, nejznámější z nich je pravděpodobně Model View Presenter. MVC se však nadále používá a zejména v posledních letech tato architektura dramaticky nabírá na popularitě. Na podzim roku 2002 odhlasovalo W3C použití MVC v architektuře XForms, která má být integrovaná do XHTML 2.0 specifikací, jež jsou momentálně ve fázi návrhu (draft). [2] [6]

#### 3.2 Vrstvy

##### 3.2.1 Model

Modelová vrstva je vlastně datová část aplikace obstarávající práci s daty. Může být realizována například jako pole čísel nebo tabulka v databázi. Nejčastěji se jedná o výběr dat z databáze a jejich uložení do pole. Model jsou tedy data plus business logika aplikace.

### 3.2.2 View

Převádí data reprezentovaná modelem do podoby vhodné k interaktivní prezentaci uživateli. V našem případě se jedná o HTML.

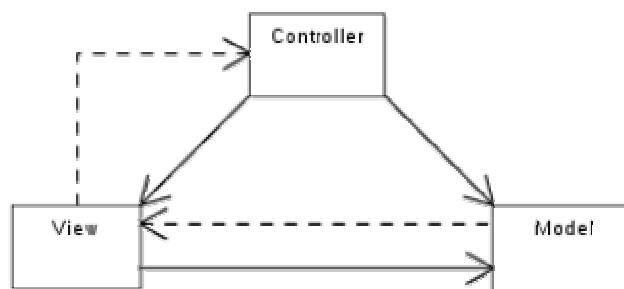
### 3.2.3 Controller

Reaguje na události (nejčastěji pocházející od uživatele) a zajišťuje změny v modelu nebo v pohledu.

## 3.3 Princip

- Je provedena nějaká akce v uživatelském rozhraní.
- O této akci je informován controller.
- Pokud je to nutné, controller přistoupí k modelu a provede v něm potřebné úpravy.
- Zavolá se opět view, aby zobrazil zaktualizované položky.
- Uživatelské rozhraní očekává další příkaz, který by celý koloběh zahájil znovu.

## 3.4 Pravidla a vazby



Obr.5: Vazby mezi jednotlivými vrstvami.

K udržení správného MVC návrhu musíme dodržovat některá pravidla, jednotlivé vazby demonstruje obrázek

- Controller má přímý odkaz na Model, aby mohl upravit jeho data
- Controller má přímý odkaz na View, aby mu mohl předávat parametry apod.
- View má přímý odkaz na Model, aby mohl jeho data zobrazit

- Model NESMÍ nikdy mít přímou vazbu na View nebo Controller, byla by to podstatná chyba v návrhu aplikace podle MVC

Některé aplikace mohou obsahovat nepřímé vazby (označené čárkovaně), což znamená že cílové prvky mohou být upozorněny nějakým mechanismem na změnu dat. [2] [6]

### 3.5 MVC a Web

Webové frameworky existují již několik let. Prvním skutečně použitým frameworkem na webu byl Fusebox vydaný roku 1997. Umožňoval programování v jazyce ColdFusion. Následovala řada dalších, například Struts psaný v Javě. K tomuto frameworku vzniklo několik PHP klonů, jejich zásadním problémem byla však poměrně tragická rychlost. Dalším mezníkem na poli frameworků byl roku 2004 Ruby on Rails založený na v té době nepříliš známém jazyku Ruby. Tímto frameworkem se potom inspirovaly PHP frameworky, jak je známe dnes. [2] [6]

### 3.6 Vybrané php MVC frameworky

#### 3.6.1 CakePHP

Je poměrně velmi rozšířeným frameworkem s počátky sahajícími do roku 2005. Zajímavostí tohoto frameworku je, že obsahuje vlastní rozhraní příkazového řádku. Cake je určen pro PHP 4 i PHP 5. Díky jeho rozšířenosti je na internetu k dispozici i mnoho českých návodů. Framework je poměrně robustní a jeví se jako vhodný pro práci na velkých i malých projektech. Jeho nevýhodou je, že jsou vyvíjeny paralelně dvě verze, které nejsou navzájem úplně kompatibilní, což může být problémem při hledání různých tutoriálů apod. [2] [6]

#### 3.6.2 Qcodo

Tento framework stojí na dvou na sobě nezávislých hlavních komponentách : Code Generatoru a Qforms. Code Generator vytváří z databáze virtuální objektovou strukturu, přes kterou do ní pak přistupuje aplikace. Qforms je šablonovací systém inspirovaný platformou .NET. Slabinou tohoto frameworku je zatím nepříliš rozsáhlá dokumentace, kde se dá hodně věcí vyčíst pouze z příkladů. QCodo je aktivně udržovaným frameworkem, a jeho vlastnosti se uplatnily i například u NASA. [2] [6]

### 3.6.3 Nette

Jedná se o velmi mladý framework českého autora, který je stále ve fázi vývoje. Obsahuje zatím pouze hrstku modulů a dokumentace oproti ostatním frameworkům také zaostává. Jako svou výhodu uvádí Nette možnost práce s jmennými prostory a routování.

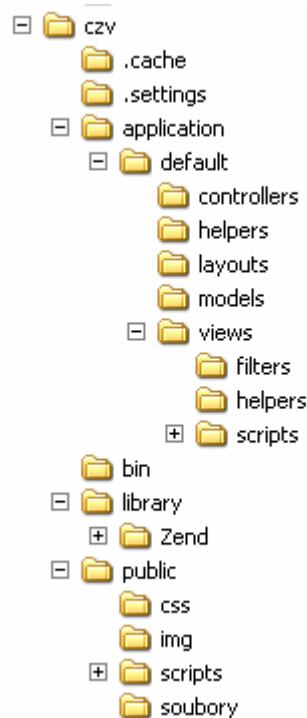
### 3.6.4 Zend Framework

Historie tohoto frameworku sahá do počátku roku 2005, kdy byla představena první veřejná verze. Vývoj tohoto frameworku zaštiťuje samotná společnost Zend Technologies, jejíž zakladatelé Andi Gutmans a Zeev Suraskive ve velké míře ovlivnili vývoj php. Za frameworkem stojí také řada vlivných sponzorů, například IBM, Google, Microsoft a Adobe Systems. Nespornou výhodou tohoto frameworku je velmi propracovaná a přehledná dokumentace. Často bývá doplněna velkým množstvím příkladů, na oficiálních stránkách nalezneme například i různé videotutoriály. Tento framework je dosti rozšířený a komunita kolem něj je velmi rozsáhlá. Na internetu tak existují velká fóra v různých jazycích, jež jsou naplněna řešenými problémy. Zend má velké zastoupení i mezi českými vývojáři. Lze tedy najít mnoho návodů a řešených problémů právě v češtině.

Nezanedbatelnou nevýhodou tohoto frameworku je oproti některým nižší rychlost. Tento problém lze vyřešit použitím php akcelérátoru nebo jej alespoň omezit používáním komponenty Zend\_Cache. [2] [6]

## 4 ZEND FRAMEWORK

### 4.1 Adresářová struktura



Obr.6: Adresářová struktura dle konvence, v praxi se může lišit

#### 4.1.1 Application

Prvním důležitým adresářem je **application** obsahující důležitý soubor bootstrap.php (o něm později), a dále jednotlivé složky jako tzv. namespaces neboli jmenné prostory. Pokud tuto možnost nevyužijeme, používáme pouze podadresář default. V něm nalezneme zejména adresáře controllers, models a views do kterých ukládáme jednotlivé třídy tak aby byly správně zařazeny podle návrhového vzoru MVC.

#### 4.1.2 Library

Další klíčový adresář **library** slouží k uložení vlastních knihoven frameworku. Sem tedy pro instalaci nakopírujeme stažené soubory. Pro úsporu místa můžeme teoreticky ty komponenty které nepoužíváme vymazat, případně si je na vlastní riziko upravit k obrazu svému.

### 4.1.3 Public

Posledním důležitým adresářem je **public**, který bude výchozím adresářem, přístupným z webu. Bude tedy obsahovat výchozí soubor `index.php` a velmi důležitý `.htaccess`, který směřuje definované požadavky právě na `index.php`.<sup>[3]</sup>

```
1 RewriteEngine on
2 RewriteBase /czv
3 RewriteRule !\.(?:is|ico|txt|css|js|php|mhtml|png|jpe|gif|png|css|)$ index.php
```

Obr.7:Příklad nastavení `.htaccess`

Na obrázku uvedený soubor bude všechny znaky ve kterých se nevyskytují řetězce vyjmenované na třetím řádku směřovat právě na `index.php`. Pokud by tento soubor nebyl přítomen, musely by adresy místo např.

```
http://localhost/czv/index/index
```

vypadat takto:

```
http://localhost/czv/index.php/index/index
```

Ve druhém řádku je nastavení prefixu před řetězec, což je vhodné zejména pokud máme definovaný alias v konfiguračním souboru apache, tedy `http.conf`

## 4.2 Tvar adresy

Tvar adresy bude nejlepší vysvětlit na příkladě, mějme tedy adresu:

```
http://localhost/czv/index/index/cat/1/page/1
```

- První částí, tedy `http://localhost/czv/` je standardní **adresa** webové stránky, tou by bylo zbytečné se zabývat.
- Mezi lomítka uvězněný následuje název **controlleru**, který obhospodařuje danou stránku
- Dále následuje **action**, což je jednak název funkce přímo v controlleru, a také název volaného view skriptu umístěného ve `views/scripts`.
- Poslední jsou dvě **proměnné** předané v adresním řádku. První jsou jejich názvy (`cat`, `page`), následované hodnotami (v tomto případě jedničky). K těmto proměnným může controller přistupovat pomocí metody `getParam` vestavěné třídy `Request`. Přestože v příkladu jsou obě hodnoty číselné, není problém takto předat i řetězec.

## 4.3 Důležité soubory:

### 4.3.1 Index.php

```
1 <?php
2 /**
3  * Celoživotní vzdělávání
4  *
5  * @author Richard Vytásek
6  * @version 0.2
7  */
8 error_reporting(E_ALL|E_STRICT);
9 ini_set('display_errors', 1);
10 date_default_timezone_set('Europe/Prague');
11 require '../application/bootstrap.php';
```

*Obr.8:Příklad souboru Index.php*

Na tento soubor jsou směřovány veškeré požadavky z adresního řádku. Hlavní smysl tohoto souboru je napojit na sebe pomocí funkce `require` soubor `bootstrap.php`, zpravidla umístěný v adresáři `application`. Do tohoto souboru můžeme také vložit některé vhodné řádky. Například je doporučeno sem vložit php funkci `date_default_timezone_set`, která je vyžadována specifikací PHP 5.1 a vyššími. Další vhodnou php funkcí je `error_reporting` k nastavení jaké chyby se mají hlásit.



### 4.3.2 Bootstrap.php

```
8 set_include_path('.' . PATH_SEPARATOR . '../library' . PATH_SEPARATOR . '../
9
10 require_once 'Initializer.php';
11 require_once "Zend/Loader.php";
12 Zend_Loader::loadClass('Zend_Db_Table');
13 Zend_Loader::loadClass('Zend_Debug');
14 Zend_Loader::loadClass('Zend_Auth');
15 Zend_Loader::loadClass('Zend_Locale');
16 Zend_Loader::registerAutoload();
17 Zend_Locale::setDefault('cs');
18
19 // Vytvoření front controlleru:
20 $frontController = Zend_Controller_Front::getInstance();
21 $frontController->throwExceptions(true);
22 $frontController->registerPlugin(new Initializer('development')); // Change
23
24 // načtení konfigurace:
25 $config = new Zend_Config_Ini('../application/config.ini', 'general');
26 $registry = Zend_Registry::getInstance();
27 $registry->set('config', $config);
28 // nastavení databáze:
29 $db = Zend_Db::factory($config->db);
30 Zend_Db_Table::setDefaultAdapter($db);
31 Zend_Registry::set('db', $db);
32
33 $auth = Zend_Auth::getInstance();
34
35 // Spuštění:
36 $frontController->dispatch();
```

Obr.9:Příklad souboru Bootstrap.php

Tento soubor je možné i přímo sloučit s index.php, častěji se však vyskytují odděleně. Hlavním účelem tohoto souboru je konfigurace frameworku. Zejména zaincludujeme potřebné adresáře a pomocí modulu Loader načteme jednotlivé Třídy. Dalším krokem je vytvoření front controlleru a jeho nastavení. Dále se zde běžně nastaví načítání konfiguračního souboru (obsahujícího nejčastěji přístupové údaje k databázi) a inicializuje se databáze. Posledním krokem je spuštění front controlleru.

## 5 POUŽITÝ SOFTWARE

### 5.1 PSpad editor 4.5.3

Jelikož php, html, css i javaskripty se ukládají ve formě normálního textu, teoreticky pro jejich editaci stačí i obyčejný poznámkový blok obsažený v systému Windows. Tato možnost ale nepatří úplně nejvhodnější, protože poznámkový blok neobsahuje některé pokročilé funkce, zejména pak zvýrazňováním syntaxe, které má velký vliv na přehlednost výsledného kódu. Další problém nastane, pokud se rozhodneme editovat soubor uložený v kódování UTF-8. Mezi řadou editorů jsem si nakonec vybral osvědčený freeware PSpad pocházející od českého autora. Přestože poslední verze tohoto programu je již téměř dva roky stará, svou práci zde odvede poměrně bez problému.

### 5.2 Mozilla Firefox 3.0.x

Tento webový prohlížeč okupující druhou příčku používanosti byl hlavním nástrojem pro testování aplikace. Dílčí testování byla provedena i na prohlížečích Internet Explorer 8.0.x a Google Chrome 2.0.169.1.

### 5.3 Web developer toolbar Plugin 1.1.6

Vynikající doplněk pro prohlížeč Mozilla Firefox. Po nainstalování přidá do aplikace novou lištu s řadou nástrojů, které dokáží poměrně zásadním způsobem urychlit některé úkony vývojáře. Osvědčil se zejména při validaci a vyhledávání chyb.

### 5.4 EasyPhp 2.0.0.0

EasyPhp je soubor programů obsahující webový server Apache, PHP verze 5, MySQL a PHPMyAdmin, tedy všechny nástroje potřebné k testování aplikace na localhostu. Výhodou je že tyto komponenty nenainstaluje jako služby, tzn. že je spouštíme, až když jsou potřeba a neběží nám na pozadí pořád. Do lišty vedle hodin po spuštění navíc přidá ikonku s odkazy na nejdůležitější operace se serverem.

## 5.5 IrfanView 4.20

Je prohlížeč obrázků vynikající zejména svojí bleskovou rychlostí, širokou podporou formátů a rozšiřitelností o pluginy. Umožňuje také provádět základní operace s obrázky, použití našel zejména při ořezávání.

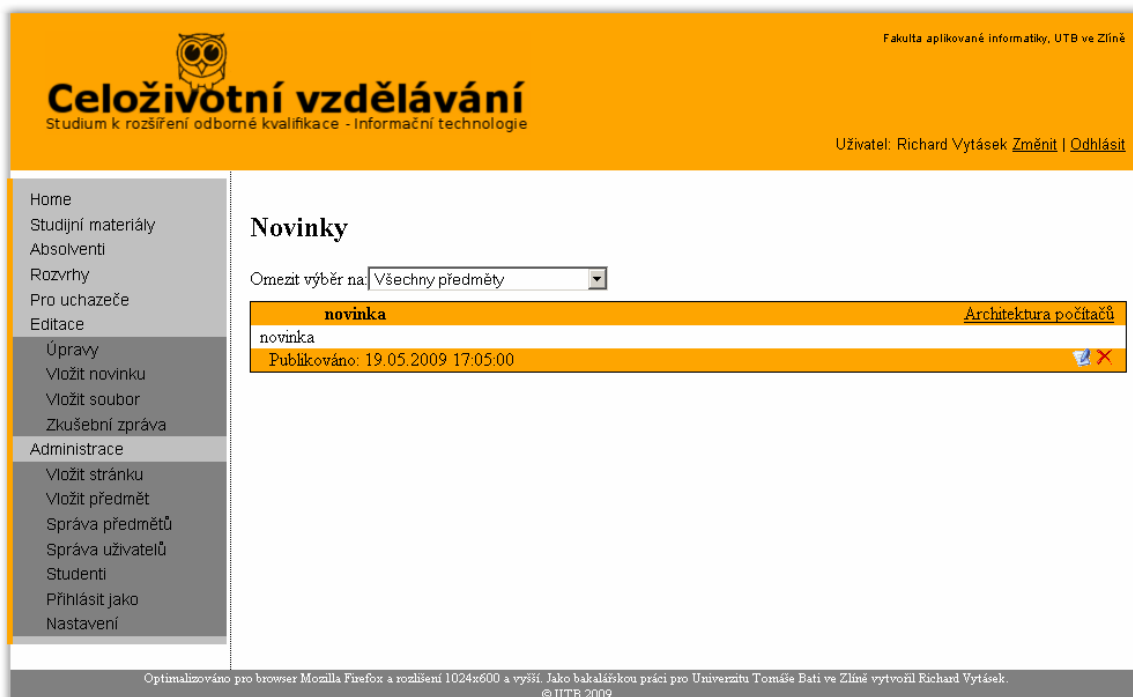
## 5.6 Gimp 2.4.6

Pro pokročilejší práci s bitmapovými obrázky byl použit tento software. Obsahuje nesčetné množství nástrojů, filtrů a hlavně podporuje vrstvy. Velkou výhodou tohoto programu je, že je open source a multiplatformní.

## **II. PRAKTICKÁ ČÁST**

## 6 POPIS APLIKACE

### 6.1 Design



Obr.10: Vzhled webu po přihlášení administrátora

Podle přání vedoucího práce byl vytvořen klasický layout s hlavičkou, patičkou a menu po levé straně. Po dlouhém zkoušení byla nakonec zvolena kombinace barev tvořená oranžovou, bílou a několika odstíny šedi. Vybrané barvy spolu navzájem dobře kontrastují, což přispívá k dobré čitelnosti a snadné orientaci na webu. Hlavičce vévodí jednoduché logo se sovou, která je v přírodě známá svou moudrostí, a proto se již poměrně dávno stala symbolem pro vzdělávání. Formát loga je typu png, nedochází tedy k rozmazání hran, jako např. u jpg. Původním cílem bylo logo průhledné pro jednodušší následné změny designu, ovšem díky špatné interpretaci v prohlížeči Internet Explorer 6.x bylo nakonec od tohoto záměru ustoupeno a pozadí nastaveno pevně na oranžovou. Menu bylo kvůli větší přehlednosti zvoleno dvouúrovňové, po kliknutí na položky Editace či Administrace se zobrazí ještě podmenu s dalšími položkami. Těto vlastnosti bylo dosaženo jednoduchým javascriptem, který mění display vlastnost položek z none na block a naopak.

### 6.1.1 Kaskádové styly



Obr.11: Ikonky pro známé typy souborů a pro odkaz vedoucí mimo web

Web používá stylování pomocí css, a protože pravidel je poměrně hodně, jsou uložena v samostatném souboru default.css, na který je odkázáno standartně z hlavičky dokumentu. Zajímavým způsobem pokročilého stylování je využití stylů pro zobrazování ikoněk k souborům. Pomocí zápisu `a[href$='.pdf']` lze omezit použití následujících pravidel pouze na odkazy s určitým typem souborů, dále již není problém nastavit ikonku na pozadí odkazu a text odsadit o požadovanou velikost. Tato feature nefunguje v prohlížeči Internet Explorer od Microsoftu, nejedná se však o funkci nezbytnou, soubory se v něm prostě zobrazí pouze bez ikonky.

## 6.2 Databáze

Databáze hraje v aplikaci velmi významnou roli, figuruje zde jako primární úložiště veškerých údajů. Aplikace využívá celkem 7 tabulek.

### 6.2.1 Popis jednotlivých tabulek

#### 6.2.1.1 Tabulka users

Sloupec	Typ
<b>id</b>	int(11)
<b>username</b>	varchar(30)
<b>password</b>	varchar(50)
<b>real_name</b>	varchar(60)
<b>role</b>	enum('admin', 'teacher', 'user')
<b>student_id</b>	int(11)

Obr.12: Struktura tabulky users

Tato tabulka slouží k uložení přihlašovacích údajů všech uživatelů webu. Obsahuje přihlašovací jméno uživatele, jeho heslo zašifrované funkcí SHA1, celé jméno uživatele, jeho roli a nakonec nepovinnou položku student\_id, která odkazuje na řádek tabulky students. Použité role uživatelů jsou celkem tři: user, teacher a admin.

### 6.2.1.2 Tabulka categories

Sloupec	Typ
<u>id</u>	int(10)
name	varchar(100)

Obr.13: Struktura tabulky categories

Tato tabulka má sloužit pro ukládání předmětů. Obsahuje pouze dva sloupce a to název předmětu a jeho id, na které se odkazují tabulky files a news.

### 6.2.1.3 Tabulka files

Sloupec	Typ
<u>id</u>	int(10)
name	varchar(50)
descr	text
path	varchar(255)
date	datetime
cat_id	int(10)
author_id	int(11)

Obr.14: Struktura tabulky files

V této tabulce jsou uloženy všechny potřebné údaje k jednotlivým souborům a to: jejich název, textový popis, cesta k souboru, datum nahraní a dále id řádků tabulek users a categories pro identifikaci uploadera a přiřazení k předmětu.

### 6.2.1.4 Tabulka news

Sloupec	Typ
<u>id</u>	int(11)
author_id	int(11)
cat_id	int(11)
date_created	datetime
date_updated	datetime
name	varchar(100)
body	mediumtext

Obr.15: Struktura tabulky news

Struktura této tabulky je velmi podobná tabulce files, místo cesty k souboru a popisu však obsahuje sloupeček body, jenž obsahuje html text formátovaný wysiwyg editorem TinyMCE.

#### 6.2.1.5 Tabulka pages

Sloupec	Typ
<b>id</b>	int(10)
<b>name</b>	varchar(30)
<b>body</b>	text
<b>author_id</b>	int(11)
<b>date</b>	datetime

Obr.16: Struktura tabulky pages

Účelem této tabulky je uložení stránek tvořených administrátorem pomocí wysiwig editoru TinyMCE. Mimo těla obsahuje název, id autora a datum a čas poslední úpravy.



6.2.1.6 *Tabulka students*

Sloupec	Typ
<b>id</b>	int(11)
<b>beh</b>	int(11)
<b>absolvent</b>	tinyint(1)
<b>jmeno</b>	varchar(255)
<b>prijmeni</b>	varchar(255)
<b>rozen</b>	varchar(255)
<b>titul</b>	varchar(20)
<b>datum_narozeni</b>	varchar(10)
<b>misto_narozeni</b>	varchar(255)
<b>adresa_ulice</b>	varchar(255)
<b>adresa_mesto</b>	varchar(255)
<b>adresa_psc</b>	varchar(5)
<b>email</b>	varchar(255)
<b>telefon</b>	varchar(14)
<b>vzdelani</b>	tinytext
<b>rok_ukonceni</b>	year(4)
<b>ucitel</b>	tinyint(1)
<b>predmety</b>	mediumtext
<b>zamestnavatel_nazev</b>	varchar(255)
<b>zamestnavatel_ulice</b>	varchar(255)
<b>zamestnavatel_mesto</b>	varchar(255)
<b>zamestnavatel_PSC</b>	varchar(5)
<b>platce</b>	varchar(15)
<b>fakturacni_adresa</b>	text

Obr.17: *Struktura tabulky students*

Nejrozsáhlejší tabulka uchovává všechny potřebné informace o studentech a absolventech kurzu. Klíčové jsou sloupce **beh**, přiřazující studenta k danému běhu kurzu, a **absolvent**, což je příznak udávající, zda je student zdárným absolventem kurzu. K těmto dvěma sloupečkům nemá student z pochopitelných důvodů přístup.

### 6.2.1.7 Tabulka settings

Sloupec	Typ
<b>id</b>	int(11)
<b>name</b>	varchar(20)
<b>value</b>	varchar(20)

Obr.18: Struktura tabulky settings

V této tabulce jsou uchovány dvě zásadní nastavení pro aplikaci, a to aktuální běh kurzu a klíč k registraci. Aktuální běh se automaticky kopíruje do řádků tabulky students při vyplnění dotazníku. Klíč k registraci byl zaveden k zabránění registrací uživatelů v nepravou dobu, čímž by mohlo dojít k přiřazení studenta ke špatnému běhu.

### 6.2.2 Přístup k databázi přes model

Přístup k databázi lze v Zend Frameworku řešit hned několika způsoby. První možností je přístup přes standardní php funkce. Dalším způsobem je použití abstrakční vrstvy Zend\_Db\_Adapter. Toto řešení má nespornou výhodu při migraci na jiný typ databáze, kdy poté v aplikaci stačí prakticky změnit jednu hodnotu. Výčet podporovaných typů databázových systémů je vskutku rozsáhlý. Bohužel použití tohoto způsobu sebou nese i nezanedbatelnou nevýhodu, kterou je negativní dopad na výkon a rychlost. Jelikož je ale u této aplikace velmi nepravděpodobné, že by databáze zpracovávala tisíce požadavků za vteřinu, byla nakonec zvolena druhá možnost. V aplikaci je však použita ještě jedna forma abstrakce, a to Zend\_Db\_Table. Tato komponenta nám umožní si přímo definovat jednotlivé tabulky jako třídy, ke kterým pak můžeme po vytvoření instance přistupovat objektové.

## 6.3 Struktura aplikace

Jak již bylo řečeno v kapitole 4.2, aplikace v Zend Frameworku je tvořena pomocí Controllerů, sdružujících jednotlivé Actions. Tato aplikace využívá celkem 5 controllerů.

### 6.3.1 ErrorController

Tento controller slouží k výpisu chyb. Soubor byl stažen spolu s balíčkem knihoven a při práci na aplikaci do něj už nebylo zasahováno.

### 6.3.2 AuthController

Úkolem tohoto controlleru je spravovat přihlašování, odhlašování a registrace uživatelů. K přihlašování a odhlašování je použita komponenta Zend\_Auth.

### 6.3.3 IndexController

Tento controller má za cíl sdružovat akce dostupné nepřihlášeným uživatelům a uživatelům s oprávněním user. Tyto akce se nejčastěji starají o výpis dat z databáze.

### 6.3.4 EditController

Cílem tohoto controlleru je sdružování akcí dostupných pouze uživatelům s oprávněním teacher nebo admin. Kontrola oprávnění se provádí ve speciální funkci preDispatch(), která je automaticky volána před jakoukoli action. Při nedostatečném oprávnění provede přesměrování na výchozí stránku. Alternativou této možnosti by bylo použití komponenty Zend\_Acl, která je ke spravování přístupových práv přímo stvořená. Pro účely této aplikace je však dostačující i způsob bez Acl.

### 6.3.5 AdminController

Tento controller, podobně jako EditController sdružuje akce dostupné výhradně uživatelům s oprávněním admin.

## 6.4 Uživatelské rozhraní



Pro navigaci na webu slouží menu umístěné klasicky na levé straně stránek. Hlavička webu ukrývá důležité odkazy pro registraci, změnu údajů a linky pro přihlášení nebo odhlášení.

### 6.4.1 Stránky přístupné všem

Pro uživatele bez přihlášení jsou k dispozici pouze funkce umožňující prohlížení informací. Na úvodní stránce se zobrazují novinky, jež je možné pomocí ovládacího prvku select omezit pouze na určitý předmět. Další veřejně přístupnou stránkou jsou studijní materiály, kde, jak nám už název napovídá, jsou umístěny soubory sloužící k podpoře výuky. Před vybranými typy souborů se navíc zobrazuje ikonka reprezentující daný typ souboru. Tato vlastnost se díky definici v css stylech projevuje na všech stránkách webu.

Stejně jako novinky jsou tyto soubory řazeny podle předmětů. Třetí veřejnou funkcí je výpis všech absolventů kurzu. Návštěvník zde má možnost zobrazit všechny absolventy, či omezit výběr pouze na jeden konkrétní běh kurzu. Aplikace byla naplněna i účastníky kurzů předcházejících vzniku tohoto webu. Samozřejmostí u těchto stránek je stránkování, kdy po překročení nastaveného počtu objektů se ty nejstarší už nezobrazují na úvodní straně. Na další stránku se lze přepnout kliknutím na obdélníček s jejím číslem umístěný dole. Poslední veřejně přístupná funkce jsou stránky pro různé účely vytvořené administrátorem pomocí wysiwyg editoru TinyMCE. Mohou tedy obsahovat tabulky, obrázky, odkazy a všechny ostatní standardní html objekty. Administrátor navíc u těchto stránek má odkazy pro úpravu a mazání.

The screenshot shows a web interface with a yellow header. The header contains the logo of an owl and the text 'Celoživotní vzdělávání' (Lifelong Education) with the subtitle 'Studium k rozšíření odborné kvalifikace - informační technologie'. In the top right corner, it says 'Fakulta aplikované informatiky, UTB ve Zlíně' and 'Uživatel: Richard Vytásek [Změnit](#) | [Odhlásit](#)'. On the left, there is a vertical navigation menu with links: Home, Studijní materiály, Absolventi, Rozvrhy, Pro uchazeče, Editace, and Administrace. The main content area is titled 'Novinky' (News) and includes a dropdown menu labeled 'Omezit výběr na' with 'Všechny předměty' selected. Below this is a table of news items:

novinka	Architektura počítačů
novinka	
Publikováno: 19.05.2009 17:05:00	 

The footer of the page contains the text: 'Optimalizováno pro browser Mozilla Firefox a rozlišení 1024x600 a vyšší. Jako bakalářskou práci pro Univerzitu Tomáše Bati ve Zlíně vytvořil Richard Vytásek. © UTB 2009'.

Obr.19: Ukázka stránky novinky

Optimalizováno pro browser Mozilla Firefox a rozlišení 1024x600 a vyšší. Jako bakalářskou práci pro Univerzitu Tomáše Bati ve Zlíně vytvořil Richard Vytásek. © UTB 2009

Obr.20: Ukázka stránky studijní materiály

#### 6.4.2 Stránky přístupné registrovaným (studentům)

Po registraci dostává uživatel oprávnění user. Jedinou funkcí ke které má po přihlášení přístup, je úprava profilu. Na této stránce má uživatel za úkol vyplnit poměrně rozsáhlý dotazník, jehož položky jsou po odeslání uloženy do databáze. Bez tohoto kroku nemůže být uživatel administrátorem na webu zařazen mezi absolventy a rovněž bude chybět ve zkušebních zprávách generovaných touto aplikací.

#### 6.4.3 Stránky přístupné učitelům

Po přihlášení má učitel v menu navíc položku editace po jejímž rozkliknutí se zobrazí podmenu o čtyřech odkazech. První odkaz zavede učitele na stránku, kde může upravovat jím vytvořené novinky a mazat studijní materiály, které na web vložil. Dalším odkazem se učitel dostane na formulář pro vložení novinky, kde vyplní její nadpis, obsah a zvolí správné přiřazení k předmětu. Podobná je i další funkce umožňující učiteli vkládání souboru. Kromě názvu zde může připojit krátký popis k souboru. Poslední funkcí dostupnou učitelům je generování zkušební zprávy. Po vyplnění několika parametrů dojde stiskem tlačítka k vytvoření tohoto dokumentu. U této stránky je potlačeno generování

layoutu a navíc jsou použity odlišné kaskádové styly, takže se hodí pro tisk. Všechny tyto funkce jsou samozřejmě dostupné i administrátorovi webu.

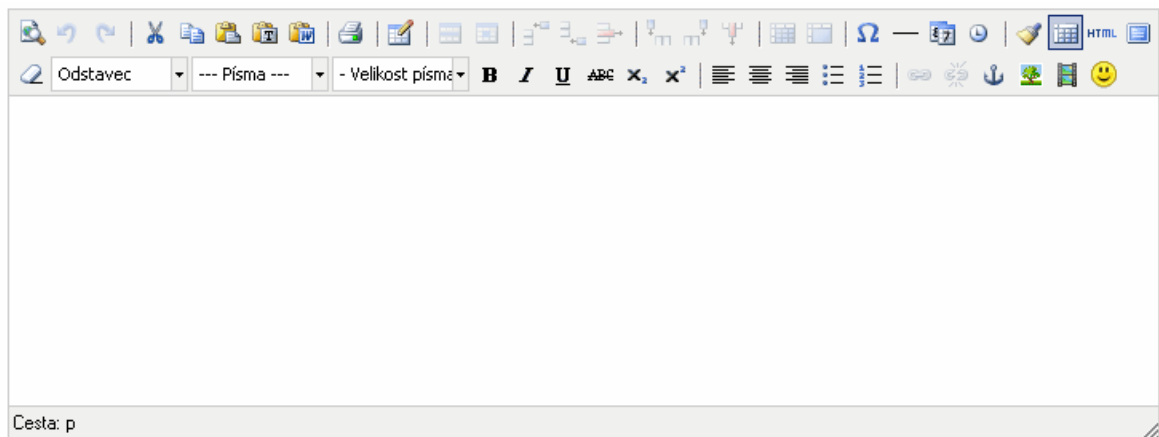
Optimalizováno pro browser Mozilla Firefox a rozlišení 1024x600 a vyšší. Jako bakalářskou práci pro Univerzitu Tomáše Bati ve Zlíně vytvořil Richard Vytásek.  
© UTB 2009

Obr.21: Ukázka stránky pro generování zkušební zprávy

#### 6.4.4 Stránky pro administraci

Stejně jako u učitele, tak i u administrátora se po přihlášení v menu objeví navíc jedna položka. Po rozkliknutí menu administrace se objeví dalších sedm odkazů. A to odkazy na vložení stránky či předmětu, dále správa předmětů umožňující jejich mazání a úpravy. Další položkou je správa uživatelů, která administrátorovi umožňuje měnit role jednotlivých uživatelů a dále se přihlásit jako libovolný účastník. V pořadí pátou položkou je část s názvem studenti. Zde má administrátor možnost si zobrazit profil uživatelů ve formátu vhodném pro tisk a dále zde může označit několik studentů, aby u nich tlačítkem vespod změnil příznak absolventství. Další odkaz, nazvaný přihlásit jako, umožňuje přihlášení pod jiným uživatelským účtem. Tato funkce byla vytvořena přímo na požadavek vedoucího práce. Poslední položka menu, nastavení, slouží k definování dvou proměnných a to aktuálního běhu a klíče k registraci. Proměnná `aktualni_beh` je využita jednak při generování selectů vymezujících běh a je automaticky doplňována do profilů studentů. Druhá proměnná `klic_k_registraci` je vyžadována při registraci uživatelů na webu, což zabrání jednak registraci uživatelů kteří s kurzem nemají nic společného a také předčasným

registracím, kdy by, při registraci uživatele před tím, než administrátor změní aktualni\_beh, mohlo dojít ke špatnému doplnění této proměnné do profilu.



Obr.22: TinyMCE wysiwyg editor

Fakulta aplikované informatiky, UTB ve Zlíně

## Celoživotní vzdělávání

Uživatel: Richard Vytásek [Změnit](#) | [Odhlásit](#)

Studium k rozšíření odborné kvalifikace - Informační technologie

- Home
- Studijní materiály
- Absolventi
- Rozvrhy
- Pro uchazeče
- Editace
- Administrace

### Správa uživatelů

Vše
 Uživatelé
 Uživatelé s profilem
 Učitelé
 Admini

ID	Login	Jméno	Stávající Role	Změnit roli
4	sysala	Tomáš Sysala	admin	
1	riczie	Richard Vytásek	admin	

Optimalizováno pro browser Mozilla Firefox a rozlišení 1024x600 a vyšší. Jako bakalářskou práci pro Univerzitu Tomáše Bati ve Zlíně vytvořil Richard Vytásek.  
© UTB 2009

Obr.23: Ukázka stránky pro správu uživatelů

## ZÁVĚR

Svou bakalářskou práci jsem rozdělil do šesti částí, z nichž první je krátké shrnutí vývoje webu. Další část je věnována nejrozšířenějším technologiím používaným pro tvorbu dynamického webu, které byly taktéž použity v praktické části. Třetí kapitola se zabývá webovými MVC frameworky, jejichž obliba v posledních letech rapidně roste. Tohoto tématu se rovněž týká čtvrtá kapitola, jenž se věnuje jednomu z nejrozšířenějších frameworků dnešní doby, Zend Frameworku. Pátá kapitola je pouze výčtem software použitého při vývoji zadané aplikace.

Praktická část se věnuje popisu vytvořené aplikace, jak z pohledu uživatele, tak z pohledu učitelů a administrátora webu.

Vytvořený web umožňuje uživatelům s příslušným oprávněním rychlou a efektivní editaci obsahu webu, čímž poměrně zásadním způsobem zlepší jeho aktualizovatelnost a přizpůsobitelnost požadavkům administrátora. Toto by mělo přispět ke zkvalitnění výuky kurzu.



## ZÁVĚR V ANGLIČTINĚ

My bachelor project is divided into six parts. First one is a brief summary of web evolution. Next chapter focuses on most common technologies used for modern dynamic websites of today. These technologies were also used in my practical part of the project. Content of third part is about web application frameworks, whose popularity in recent years grows rapidly. Fifth part is only a list of software used for application development

Practical part is focused on description of created application from both view of the user and view of administrators and teachers.

Created web allows users with proper privileges quick and efficient modification of web content, thus extensively improving updatability and customization to administrator needs. This should contribute to improvement of this course.

## SEZNAM POUŽITÉ LITERATURY

- [1] Allen Rob, Lo Nick - Zend Framework in Action, Copyright 2007 Manning Publications, ISBN: 1933988320.
- [2] Wikipedia [online]. [cit.2009-05-10]. Dostupný z WWW:<<http://wikipedia.org>>
- [3] Pari.cz – Český web (nejen) o Zend Frameworku [online]. [cit.2009-05-10]. Dostupný z WWW: <<http://pari.cz/293/zaciname-se-zend-framework-15>>.
- [4] Root – Velký test php frameworků [online]. [cit. 2009-05-10]. Dostupný z WWW: <<http://www.root.cz/clanky/velky-test-php-frameworku-2008/>>.
- [5] Root – Přehled a vývoj php frameworků [online]. [cit. 2009-05-10]. Dostupný z WWW: <<http://www.root.cz/clanky/prehled-a-vyvoj-php-frameworku/>>.
- [6] Zdroják.cz – Úvod do architektury MVC [online]. [cit. 2009-05-10] Dostupný z WWW: <<http://zdrojak.root.cz/clanky/uvod-do-architektury-mvc/>>.
- [7] Jakpsatweb.cz [online] Dostupný z WWW: <<http://www.jakpsatweb.cz/>>.
- [8] Zend Framework Manuál [online]. Zend Technologies, [cit. 2008-05-05]. Text v angličtině. Dostupný z WWW: <<http://framework.zend.com/manual/en/>>.
- [9] Mach,J.:Php pro úplné začátečníky, Computer Press, Praha 2002.
- [10]Kosek,J.:PHP – tvorba interaktivních internetových aplikací, Grada Publishing 1999.
- [11]J. Castagnetto, H.Rawat, S. Schuman, Ch. Scyllo, D.Veliath: Php – programujeme profesionálně, Computer Press, Praha 2002.
- [12]Ullman L.:PHP a MySQL – Názorný průvodce tvorbou dynamických WWW stránek, Computer Press, červen 2004.
- [13]Škultéty R.: Javascript – Kapesní přehled, Computer Press, leden 2006.
- [14]Prokop M.:CSS kaskádové styly pro webdesignéry (2.vydání), Computer Press, únor 2005.

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

Cern	Conseil Européen pour la Recherche Nucléaire
CSS	Cascading Style Sheet
Gimp	GNU Image Manipulation Program
GNU	GNU's not Unix
GPL	General Public Licence
HTML	Hypertext Markup Language
http	Hypertext Transfer Protocol
Lamp	Linux Apache MySQL Php
MVC	Model View Controller
PHP	Personal Homepage , Php Hypertext Preprocessor
PNG	Portable Network Graphics
Rdbms	Relational Database Management System
SHA	Secure Hash Algorhytm
url	Uniform Ressource Locator
UTF	UCS Transformation Format
W3c	World Wide Web Consortium
www	World Wide Web
WYSIWYG	What you see is what you get
xHTML	Extensible Hypertext Markup Language

**SEZNAM OBRÁZKŮ**

Obr.1 - Logo Linux.....	13
Obr.2 - Logo Apache.....	14
Obr.3 - Logo MySQL.....	14
Obr.4 - Logo PHP.....	15
Obr.5 – Vazby mezi jednotlivými vrstvami MVC.....	19
Obr.6 – Adresářová struktura Zend Frameworku.....	22
Obr.7 – Příklad nastavení souboru .htaccess.....	23
Obr.8 – Příklad souboru Index.php.....	24
Obr.9 – Příklad souboru Bootstrap.php.....	25
Obr.10 - Vzhled webu po přihlášení administrátora.....	29
Obr.11 - Ikonky pro známé typy souborů a pro odkaz vedoucí mimo web.....	30
Obr.12 - Struktura tabulky users.....	30
Obr.13 - Struktura tabulky categories.....	31
Obr.14 - Struktura tabulky files.....	31
Obr.15 - Struktura tabulky news.....	31
Obr.16 - Struktura tabulky pages.....	32
Obr.17 - Struktura tabulky students.....	33
Obr.18 - Struktura tabulky settings.....	34
Obr.19 - Ukázka stránky novinky.....	36
Obr.20 - Ukázka stránky studijní materiály.....	36
Obr.21 - Ukázka stránky pro generování zkušební zprávy.....	37
Obr.22 - TinyMCE wysiwyg editor.....	38
Obr.23 - Ukázka stránky pro správu uživatelů.....	38

## SEZNAM TABULEK

Tab.1 – Historie php.....	15
---------------------------	----