

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Vytvoření programové podpory pro organizační  
zajištění konference ŘÍP

Libor Panský

Bakalářská práce

2011

Univerzita Pardubice  
Fakulta elektrotechniky a informatiky  
Akademický rok: 2010/2011

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Libor PANSKÝ**  
Osobní číslo: **I07748**  
Studijní program: **B2646 Informační technologie**  
Studijní obor: **Informační technologie**  
Název tématu: **Vytvoření programové podpory pro organizační zajištění konference ŘíP**  
Zadávající katedra: **Katedra informačních technologií**

### Z á s a d y p r o v y p r a c o v á n í :

Cílem bakalářské práce je realizace systému sloužícího k organizaci vědecko-technické konference. Práce bude zaměřena na návrh a implementaci webové aplikace využívající relační databázi. Aplikace bude mít dvě části - veřejnou, sloužící pro přihlašování účastníků konference a administrativní, určenou pro pořadatele. Aplikace bude umožňovat přihlášení a evidenci všech aktuálně přihlášených účastníků, fakturaci poplatku pro soukromé osoby i organizace, zobrazení a evidenci aktuálního programu konference a výstup do přehledné tisknutelné podoby.

Rozsah grafických prací:

Rozsah pracovní zprávy:

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

Seznam odborné literatury:

**\*PÍSEK, Slavoj. HTML a XHTML - začínáme programovat. Grada Publishing a.s., 2003. 256 s. ISBN 80-247-0571-0**

**\*DRUSKA, Peter. CSS a XHTML - tvorba dokonalých webových stránek krok za krokem. Grada Publishing a.s., 2006. 200 s. ISBN 80-247-1382-9**

**\*GUTMANS, Andi, BAKKEN, Stig Saether, RETHANS, Derick. Mistrovství v PHP 5. Computer Press, 2007. 656 s. ISBN 978-80-251-1519-0**

**\*LACKO, Luboslav. SQL - kapesní přehled. Computer Press, 2005. 96 s. ISBN 80-251-0788-4**

**\*LACKO, Luboslav. Oracle - správa, programování a použití databázového systému. Computer Press, 2007. 576 s. ISBN 978-80-251-1490-2**

Vedoucí bakalářské práce:

**doc. Ing. Milan Javůrek, CSc.**  
Katedra řízení procesů

Datum zadání bakalářské práce: **17. prosince 2010**


Termín odevzdání bakalářské práce: **13. května 2011**



prof. Ing. Simeon Karamazov, Dr.  
děkan



L.S.



Ing. Lukáš Čegan, Ph.D.  
vedoucí katedry

V Pardubicích dne 31. března 2011

## **Prohlášení autora**

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Moravské Třebové dne 20. 07. 2011

Libor Panský

## **Poděkování**

Rád bych poděkoval panu doc. Ing. Milanu Javůrkovi, CSc. za cenné rady, které mi velice pomohly při tvorbě bakalářské práce, za vstřícnost a čas, které této práci věnoval.

## **Anotace**

Práce se zabývá návrhem informačního systému sloužícího k pořádání konference, který má za úkol nahradit stávající aplikaci. V teoretické části jsou popsány technologie, které byly použity k vývoji. V další části je rozebrána konkrétní implementace systému.

## **Klíčová slova**

informační systém, konference, PHP, Oracle SQL, XHTML, JavaScript, CSS

## **Title**

Software development for organizational arrangement of the ŘÍP conference

## **Annotation**

The thesis deals with design of information system serving to organize the conference. The system aims to replace the current application. The technologies used for development are described in the theoretical part. Specific implementation of the system is analyzed in the second section.

## **Keywords**

information system, conference, PHP, Oracle SQL, XHTML, JavaScript, CSS

## Obsah

<b>Seznam zkratk</b> .....	<b>8</b>
<b>Seznam obrázků</b> .....	<b>9</b>
<b>1 Úvodní informace</b> .....	<b>10</b>
<b>2 Požadavky</b> .....	<b>11</b>
<b>3 Použité jazyky</b> .....	<b>12</b>
3.1 XHTML.....	12
3.2 CSS.....	12
3.3 PHP.....	13
3.3.1 OCI8 .....	14
3.4 JavaScript .....	14
3.4.1 jQuery.....	15
3.4.2 Validation Engine .....	16
3.5 SQL.....	17
<b>4 Vývojová prostředí</b> .....	<b>19</b>
4.1 NetBeans.....	19
4.2 Toad Data Modeler.....	19
4.3 Oracle SQL Developer .....	19
<b>5 Servery</b> .....	<b>20</b>
5.1 Apache HTTP Server.....	20
5.2 Oracle Database.....	20
<b>6 Implementace programové podpory konference ŘÍP</b> .....	<b>21</b>
6.1 Popis aplikace .....	21
6.1.1 Charakteristika.....	21
6.1.2 Architektura .....	21
6.1.3 Vzhled.....	22
6.1.4 Uživatelé.....	23
6.2 Adresářová struktura.....	23
6.3 Návrh databáze .....	24
6.4 Funkce .....	24
6.4.1 Registrace .....	24
6.4.2 Přihlášení na konferenci .....	25

6.4.3	Práce s měnou.....	29
6.4.4	Aktuality .....	30
6.4.5	Lokalizace.....	30
6.4.6	Fakturace .....	31
6.5	Zabezpečení.....	32
6.5.1	Formuláře .....	32
6.5.2	SQL Injection .....	32
6.5.3	Transakce.....	33
6.5.4	Šifrování .....	34
<b>7</b>	<b>Závěr.....</b>	<b>35</b>
	<b>Použitá literatura a zdroje .....</b>	<b>36</b>
	<b>Příloha A – Use Case Diagram .....</b>	<b>38</b>
	<b>Příloha B – Entity-relationship diagram.....</b>	<b>39</b>
	<b>Příloha C – Obsah příloženého CD.....</b>	<b>41</b>



## Seznam zkratk

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
CSS	Cascading Style Sheets
ČNB	Česká národní banka
DIČ	Daňové identifikační číslo
E-R	Entity-relationship
GB	Gigabyte
GNU GPL	GNU General Public Licence
HTML	Hypertext Markup Language
IČO	Identifikační číslo organizace
IDE	Integrated Development Environment
MIT	Massachusetts Institute of Technology
MS	Microsoft
PHP	PHP: Hypertext Preprocessor
PX	Pixel
PL/SQL	Procedural Language / Structured Query Language
ŘÍP	Řízení procesů
SHA	Secure Hash Algorithm
VAT	Value Added Tax
WWW	World Wide Web
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language

## Seznam obrázků

Obrázek 1 - Zobrazení chybové hlášky pomocí Validation Engine .....	17
Obrázek 2 - Ukázka vzhledu aplikace .....	22
Obrázek 3 - Adresářová struktura.....	24
Obrázek 4 - Kontrola osobních údajů.....	25
Obrázek 5 - Registrační formulář pro zvolení účasti.....	26
Obrázek 6 - Registrační formulář pro přihlášení příspěvku .....	26
Obrázek 7 - Registrační formulář pro zadání ubytovacích údajů.....	27
Obrázek 8 - Registrační formulář pro vyplnění platby a požadovaných služeb.....	28
Obrázek 9 - Registrační formulář pro zadání údajů za účelem vystavení faktury .....	29
Obrázek 10 - Ukázka vygenerované proforma faktury .....	31
Obrázek 11 - Use Case diagram .....	38
Obrázek 12 - Entity-relationship diagram znázorňující strukturu hlavní části aplikace .....	39
Obrázek 13 - Entity-relationship diagram znázorňující strukturu lokalizace aplikace .....	40

## 1 Úvodní informace

Cílem této práce je vytvořit informační systém, který bude sloužit k organizaci vědecko-technické konference. Práce je zaměřena na návrh a implementaci aplikace, která usnadní registraci a správu uživatelů.

Teoretická část se zabývá jazyky, vývojovými prostředími a servery, které byly použity při vývoji informačního systému. Dále jsou zmíněny všechny hlavní aspekty, které by měl systém splňovat.

V praktické části jsou rozebrány jednotlivé postupy, které byly využity při dané implementaci aplikace. Tato kapitola detailně popisuje systém se zaměřením jak na jeho webovou, tak i databázovou část. Pro snadnější pochopení je práce doplněna o E-R diagram<sup>1</sup> databáze.

---

<sup>1</sup> Entity-relationship diagram slouží ke znázornění dat a principu funkce databázového modelu

## 2 Požadavky

Aplikace je určena k registraci účastníků na konferenci Řízení procesů. Požadavkem je nahradit stávající systém, ve kterém je registrace řešena formou zasílání vyplněných přihlášek organizátorům pomocí elektronické pošty. Tento způsob byl shledán jako nevyhovující, a proto byl vyvinut systém nový.

Protože se jedná o konferenci s dlouholetou tradicí, patří mezi základní přednosti aplikace uchování všech zadaných informací v relační databázi, a tedy možnost opakovaného použití údajů při dalších ročnících konference. Uživatelé tak nejsou nuceni vyplňovat stejné údaje při opětovné registraci.

System dále umožňuje generování proforma faktur, které tvoří nedílnou součást každé přihlášky. Výsledné položky faktury jsou doplněny a sečteny na základě uživatelem zvolených možností.

Celý projekt je lokalizován v českém a anglickém jazyce s možností jednoduchého přidání dalších jazyků.

Původní požadavek byl stanoven také na automatické generování programu konference, od kterého bylo nakonec upuštěno. Program bývá sestavován ručně s ohledem na mnoho faktorů, které nemohou být nahrazeny deterministickým algoritmem.

## 3 Použité jazyky

Každému projektu musí předcházet důkladná analýza, na základě které se volí technologie, které budou při tvorbě použity. Já jsem se rozhodl pro velkou rozšířenost a univerzálnost vypracovat informační systém pomocí kombinace níže uvedených technologií.

### 3.1 XHTML

Jazyk XHTML vychází z jazyka HTML. Zkratka je tvořena anglickými slovy eXtensible HyperText Markup Language, z čehož vyplývá, že se jedná o rozšíření původního jazyka. Ve skutečnosti se nejedná o rozšíření, ale naopak o restrikcí jednotlivých značek. Celý koncept spočívá v převedení jazyka HTML tak, aby vyhovoval tvorbě XML<sup>2</sup> dokumentů se zachováním zpětné kompatibility.[1]

Jak již bylo naznačeno, základ jazyka XHTML tvoří exaktně definované značky, tzv. tagy. Značky se uzavírají mezi dvojicí lomených závorek a vytváří jednotlivé elementy. Každý může obsahovat jeden či více atributů, na základě kterých je dále definován a formátován.[1]

**Příklad elementu:**

```
<body class="telo">Tělo dokumentu</body>
```

Příklad zobrazuje element body, který definuje tělo dokumentu. Jedná se o povinný element každé webové stránky. Text „Tělo dokumentu“ tvoří hodnotu elementu. Počáteční tag je doplněn o atribut class, který specifikuje jméno třídy.[1]

**Změny oproti HTML**

- Dokument musí začínat XML deklarací.[1]
- Nepárové tagy musí být ukončeny lomítkem.[1]
- Značky a atributy musí být psané malými písmeny, protože XML je case sensitive<sup>3</sup>.
- Hodnoty atributů musí být uzavřeny do uvozovek.[1]

### 3.2 CSS

CSS (z ang. Cascading Style Sheets) je jazyk, pomocí kterého popisujeme grafický vzhled webové stránky. Česky bývá označován synonymem kaskádové styly a nabízí nám spoustu možností. Mezi jeho největší přednosti patří oddělení vzhledu dokumentu od jeho struktury a obsahu. Ke změně výsledného designu stránky stačí jednoduše zaměnit soubor

<sup>2</sup> Extensible Markup Language – značkovací jazyk, sloužící k výměně dat a publikaci dokumentů

<sup>3</sup> Citlivé na velikost písmen

s kaskádovými styly za jiný, což umožní uživatelům výběr z několika možností a každý si tak přizpůsobí systém dle vlastních požadavků.[2]

#### **Výhody:**

- Oddělení struktury a stylu dokumentu.
- Rozsáhlejší možnosti formátování vzhledu oproti (X)HTML.[2]
- Grafické vlastnosti elementů mohou být dynamicky měněny pomocí jazyku JavaScript (viz kapitola 3.4).[2]
- Pro každé výstupní zařízení můžeme definovat jiný styl a jednoduše můžeme měnit mezi zobrazením na monitoru, tiskárně, projektoru či displeji mobilního telefonu.
- Možnost vytvoření několika alternativ zobrazení dokumentu a následné nabídnutí změny grafiky uživatelům.

#### **Nevýhody:**

- Drobné chyby v zobrazení v nejčastěji používaných prohlížečích degradují kvalitu kaskádových stylů. Tento problém je postupně řešen vydáváním nových verzí a aktualizací webových prohlížečů.

#### **Příklad tvorby kaskádového stylu:**

```
body {  
    background-color: #A0A0A0;  
    color: black;  
    padding-top: 10px;  
}
```

Příklad nastavuje selektoru body pravidla obsažená v deklaračním bloku, který je uzavřený pomocí složených závorek. Jednotlivá pravidla sestávají z identifikátoru vlastnosti, poté následuje dvojtečka a hodnota vlastnosti. Jednotlivá pravidla jsou oddělena středníkem.[2]

#### **Příklad přiřazení stylu pomocí externího souboru:**

```
<link rel="stylesheet" href="css/graphics.css" type="text/css"  
media="screen" />
```

### **3.3 PHP**

PHP je skriptovací programovací jazyk, který původně vzniknul z anglického označení Personal Home Page. Dnes má zkratka význam PHP: Hypertext Preprocessor. Jazyk je určený k programování dynamických webových stránek, kdy jsou jednotlivé

skripty zpracovány na straně serveru. Nejčastěji bývá začleněn přímo do struktury HTML či XHTML dokumentu, což lze využít při tvorbě webových aplikací. Uživateli je přenesen pouze výsledek zpracování, takže je kód chráněn před nežádoucím kopírováním.[3]

PHP je jedním z nejrozšířenějších skriptovacích webových jazyků. Je nezávislý na platformě<sup>4</sup> a podporuje mnoho knihoven a celou řadu internetových protokolů.[3]

#### **Příklad PHP skriptu:**

```
<?php echo 'Test PHP skriptu.'; ?>
```

Skript je uzavřen mezi dvojicí lomených závorek a otazníků. Na začátku se z důvodu větší kompatibility uvádí formule php. Příkaz echo vytiskne text uzavřený v apostrofech na standardní výstup. Každý příkaz musí být ukončen středníkem.

### **3.3.1 OCI8**

Pro připojení k databázi byla využita PHP knihovna OCI8. Jedná se o modul, který pomocí svých funkcí poskytuje přístup k Oracle databázím. Ke správnému chodu vyžaduje přítomnost klientských knihoven Oracle, které se liší dle verze databáze. K práci byla využita knihovna php\_oci8.dll, která umožňuje přístup do databáze verze 10g ze serveru s operačním systémem MS Windows.

## **3.4 JavaScript**

JavaScript se řadí mezi objektově orientované skriptovací jazyky. Dnes je většinou používán jako interpretovaný<sup>5</sup> programovací jazyk pro webové stránky a je vkládán přímo do HTML kódu. JavaScriptový program bývá obvykle spouštěn až po načtení celé WWW<sup>6</sup> stránky a je zpracován na klientské straně. Z tohoto důvodu vyplývají některá omezení, např. chybí možnost pracovat se soubory, aby nebylo ohroženo soukromí uživatele.[4]

Dalším omezením z důvodu zpracování na klientské straně je nutnost ošetření i jiným jazykem. Je nutné zajistit plnou funkčnost aplikace i pokud má uživatel vypnutý JavaScript nebo jeho prohlížeč není schopen zajistit správnou interpretaci.

#### **Příklad vložení JavaScriptu do dokumentu:**

```
<script type='text/javascript'>
    setTimeout("location.href='index.php';" , 1000);
</script>
```

<sup>4</sup> Platformou rozumíme pracovní prostředí, většinou operační systém

<sup>5</sup> Zdrojový kód je vykonáván interpretem, není nutné jej převádět do strojového kódu

<sup>6</sup> World Wide Web, je označení pro soustavu celosvětově propojených hypertextových dokumentů[5]

Na příkladu je znázorněno jednoduché vložení JavaScriptu do dokumentu. Kód tvoří hodnotu elementu script, ve kterém je pomocí atributu type určeno, že se jedná o JavaScriptový kód. V příkladu je použit příkaz setTimeout, který způsobí přesměrování stránky na adresu index.php s časovou prodlevou 1000 ms.

### 3.4.1 jQuery

jQuery je JavaScriptová knihovna vyvinutá k zjednodušení vytváření HTML a XHTML dokumentů na klientské straně webových aplikací. Jedná se o zdarma dostupný open-source<sup>7</sup> software šířený pod licencemi MIT<sup>8</sup> a GNU GPL<sup>9</sup>. Knihovna je navržena takovým způsobem, aby zjednodušovala procházení a změnu prvků DOM<sup>10</sup>, vytváření efektů a animací, obsluhu událostí, změnu CSS stylů a vývoj AJAXových<sup>11</sup> aplikací.[6]

Použití knihovny je poměrně jednoduché. Na domovské stránce<sup>12</sup> projektu lze stáhnout buď produktivní nebo vývojářskou verzi knihovny, na kterou se poté odkazujeme z požadované stránky.

#### Příklad připojení jQuery:

```
<script src="js/scripts/jquery-1.5.2.js"
type="text/javascript"></script>
```

#### Příklad použití jQuery:

```
<script type="text/javascript">
    $(document).ready(function() {
        $("#ramec").html("Text pomocí jQuery!");
    });
</script>
```

V příkladu je ukázána jednoduchá funkce vytvořená pomocí jQuery. Poté, co je celý dokument načten, je do rámce, který obsahuje atribut id s hodnotou ramec, vepsán text "Text pomocí jQuery!".

---

<sup>7</sup> Software s otevřeným zdrojovým kódem, který je možný za určitých podmínek používat a upravovat[11]

<sup>8</sup> Více informací o MIT na adrese [http://cs.wikipedia.org/wiki/Licence\\_MIT](http://cs.wikipedia.org/wiki/Licence_MIT)

<sup>9</sup> Více informací o GNU GPL na adrese [http://cs.wikipedia.org/wiki/GNU\\_General\\_Public\\_License](http://cs.wikipedia.org/wiki/GNU_General_Public_License)

<sup>10</sup> Document Object Model

<sup>11</sup> Asynchronous JavaScript and XML je označení pro technologie vývoje interaktivních webových aplikací, které mění obsah stránek bez nutnosti jejich opakovaného načítání[7]

<sup>12</sup> <http://jquery.com/>



### 3.4.2 Validation Engine

Validation engine je API<sup>13</sup>, které slouží k ošetření vstupu formulářů (X)HTML dokumentů. Pokud není údaj ve formuláři vyplněný nebo je chybně zadaný, skript vytvoří rámec s chybovou hláškou a nedovolí odeslání formuláře do té doby, než je údaj opraven. Vlastnosti chybových hlášek je možné nastavit v externím CSS souboru. Lze nastavit barvu, tvar, styl písma či pozici vzhledem k vstupnímu poli formuláře. Můžeme zvolit z předdefinovaných vzorů ošetření nebo si vytvořit vlastní pomocí regulárních výrazů. Samozřejmě je podpora mnoha jazykových verzí, kdy stačí jednoduše k dokumentu připojit soubor s požadovaným jazykem. Validation Engine využívá ke svému chodu knihovnu jQuery a je šířen pod licencí MIT.[8]

#### Příklad připojení Validation Engine:

```
<script src="jscripts/jquery.validationEngine-cs.js"
type="text/javascript" charset="utf-8"></script>
<script src=" jscripts/jquery.validationEngine.js"
type="text/javascript" charset="utf-8"></script>
```

#### Příklad připojení stylovacího souboru:

```
<link rel="stylesheet" href="css/validationEngine.jquery.css"
type="text/css" />
```

#### Příklad přiřazení validátoru:

```
<input class="validate[required,custom[email]]" type="text"
name="email" id="email" />
```

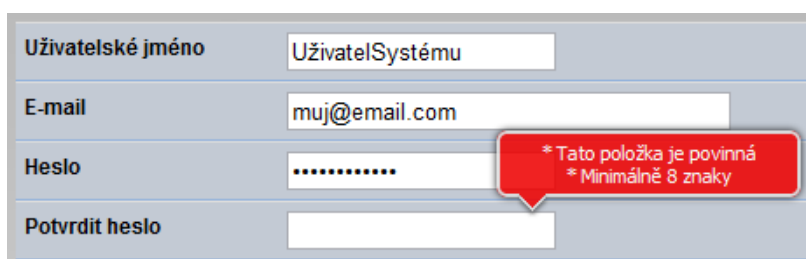
Validátor je přiřazen atributem class. Klíčové slovo required určuje, že je vyžadováno vstup vyplnit, custom[email] vyhledá pravidlo s názvem email v připojeném lokalizačním souboru.

---

<sup>13</sup> Application Programming Interface je rozhraní, které sjednocuje procedury, funkce a třídy do jednoho celku

### Příklad inicializace validátoru:

```
<script>
    $(document).ready(function() {
        $("#registrationForm").validationEngine('init', {
            promptPosition : "centerRight", scroll: false});
        $("#registrationForm").validationEngine('attach');
    });
</script>
```



The image shows a registration form with four input fields: "Uživatelské jméno" (User name) containing "UživatelSystému", "E-mail" containing "muj@email.com", "Heslo" (Password) with masked characters, and "Potvrdit heslo" (Confirm password). A red error message box is positioned over the password field, containing the text: "\* Tato položka je povinná" and "\* Minimálně 8 znaky".

Obrázek 1 - Zobrazení chybové hlášky pomocí Validation Engine

## 3.5 SQL

Zkratka SQL vychází z anglického Structured Query Language a podle názvu napovídá, že se jedná o dotazovací jazyk. SQL slouží ke komunikaci mezi klientskými aplikacemi a databázovými servery. Rozlišujeme mnoho typů serverů, což způsobuje jeden ze závažných problémů jazyka SQL, kdy se implementace liší pro jednotlivé databázové platformy. Na základní rovině příkazů sledujeme kompatibilitu mezi jednotlivými typy databází, ale po podrobném prozkoumání narazíme na jednotlivé nuance. Liší se především implementace nativních datových typů, rozšiřující množiny příkazů a klíčových slov, národní interpretace data a času či lokalizace.[9]

Databáze je strukturované úložiště dat. Všechny informace jsou zde uloženy v různých strukturách a objektech. Každá implementace musí obsahovat popis datových struktur všech údajů a kontrolní mechanismy, které zaručí integritu dat. Údaje jsou obvykle rozděleny do více tabulek, které mezi sebou mají definovány relační vazby.[9]

Tabulky se skládají ze sloupců a řádků. Sloupec definuje datový typ, někdy bývá označován pojmem atribut. Řádky neboli záznamy tvoří kombinaci hodnot sloupců v tabulce. K jednotlivých řádkům přistupujeme obvykle pomocí jednoho sloupce, který obsahuje pouze unikátní hodnoty, tzv. primární klíč. Dále rozlišujeme cizí klíče, což jsou sloupce, které jsou propojené s primárním klíčem v jiné tabulce. Pro přístup k údajům využíváme relační vazby, které jsou vytvářeny pomocí primárních a cizích klíčů.[9]

**Příklad užití SQL příkazu:**

```
SELECT id_aktuality, to_char(datum, 'dd.mm.yyyy hh24:mi:ss'),  
nadpis, text FROM aktuality ORDER BY id_aktuality desc;
```

Příkazem vybereme z tabulky aktuality všechny řádky s údaji o pořadí, datu, kdy byla vložena, nadpisu a celém textu aktuality. Všechny výsledky budou seřazeny v sestupném pořadí, kdy na prvním místě bude nejnovější aktualita. Jedná se o dotaz nad databází Oracle Database.

## 4 Vývojová prostředí

### 4.1 NetBeans

NetBeans je oblíbený vývojový nástroj mnoha programátorů po celém světě. Jedná se o open-source projekt, takže je určen k vývoji aplikací jak pro nekomerční, tak pro komerční sféru. Původně byl vyvinut firmou Sun Microsystems, která je nyní dceřinou společností firmy Oracle. NetBeans IDE<sup>14</sup> poskytuje podporu pro mnoho jazyků, pro práci byla využita podpora pro jazyky PHP a JavaScript.[10]

### 4.2 Toad Data Modeler

Toad Data Modeler je freeware<sup>15</sup> vyvinutý firmou Quest software, který zjednodušuje vývoj datových modelů s podporou mnoha databázových platforem. Byl využit ke znázornění databázové vrstvy celé webové aplikace pomocí E-R diagramu.[13]

### 4.3 Oracle SQL Developer

Jedná se o bezplatný grafický nástroj sloužící ke správě Oracle Database. Usnadňuje procházení databázových objektů, spouštění SQL příkazů, editaci a ladění PL/SQL<sup>16</sup> bloků.[14]

---

<sup>14</sup> Integrated Development Environment, anglické označení vývojového prostředí

<sup>15</sup> Typ aplikace, který autor šíří bezplatně bez možnosti úpravy zdrojového kódu při dodržení specifických podmínek – použití pro osobní potřebu, nekomerční účely atd.[12]

<sup>16</sup> Procedural Language / Structured Query Language je procedurálním rozšířením jazyka SQL od firmy Oracle

## 5 Servery

### 5.1 Apache HTTP Server

Tento server je znám pod jednoslovným názvem Apache a hraje velkou roli na poli webových serverů. Patří mezi nejoblíbenější a nejrozšířenější, což potvrzuje fakt, že se stal prvním webovým serverem, který překonal milník sto milionů provozovaných stránek po celém světě.[15]

Uživatelům poskytuje nespočet možností, které můžeme jednoduše nastavit v hlavním konfiguračním souboru. Všechny funkce jsou implementovány jako samostatné kompilované moduly, které rozšiřují základní funkci jádra serveru. Mezi základní funkce patří např. autentizační moduly, pomocí kterých nastavujeme práva přístupu k celému webu či jednotlivým složkám, podpora SSL<sup>17</sup>, modul mod\_rewrite, sloužící k přepisu URL a další.[15]

### 5.2 Oracle Database

Databáze firmy Oracle patří mezi jednu z nejrozsáhlejších na současném trhu. Spolu s DB2 od firmy IBM a SQL Server od firmy Microsoft tvoří světovou špičku mezi databázovými servery, což se značně odráží na její ceně - za plně vybavenou aplikaci na osmiprocessorovém serveru zaplatíme kolem dvanácti milionů korun.[16]

K práci byla využita Oracle Database 10g Express Edition, která je k dispozici zdarma, má však několik limitací. Maximální velikost uživatelských dat je stanovena na 4GB<sup>18</sup> a na hostitelském počítači využije maximálně 1GB operační paměti a pouze jedno procesorové jádro. Tato limitace není pro velikost této práce nijak omezující, vystačí na daleko větší projekty.[17]

---

<sup>17</sup> Secure Sockets Layer je protokol, který zabezpečuje komunikaci šifrováním přenášených dat

<sup>18</sup> Gigabyte je jednotkou množství dat, 1GB = 10<sup>9</sup> bytů

## 6 Implementace programové podpory konference ŘÍP

### 6.1 Popis aplikace

#### 6.1.1 Charakteristika

Aplikace je navržena tak, aby co nejvíce zjednodušila práci nejen organizátorům, ale také účastníkům konference. Stěžejní část tvoří registrace na konferenci. Cílem je centralizace vložených údajů s možností opakovaného použití při registraci na další ročník. K tomuto účelu byla použita databáze Oracle.

S odstupem času neshledávám řešení jako zcela vyhovující, daleko lepším řešením by dle mého názoru bylo využití databáze MySQL. Prvním důvodem je fakt, že aplikace nevyužívá ani zlomek možností databáze Oracle, ale hlavním důvodem je podpora databáze MySQL na většině hostingových serverů zcela zdarma. Možným řešením by bylo využití ODBC<sup>19</sup> API, které by vyřešilo rozdíly mezi připojovanými databázemi, takže by programátor při přechodu na jinou platformu nemusel přepisovat sáhodlouhý kód.

Webová stránka nabízí přehled informací o aktuálním pořádaném ročníku konference a všechny důležité údaje. Při práci byl kladen důraz na jednoduchost řešení. Systém je lehce ovladatelný i jednou rukou, na všechna tlačítka a pole se lze přesunout tabulátorem, takže umožňuje práci i handicapovaným uživatelům.

#### 6.1.2 Architektura

Systém je řízen jediným centrálním souborem index.php. Tento soubor detekuje pomocí URL název stránky, která má být zobrazena a otestuje, jestli existuje na serveru. Pokud nalezne webovou stránku s odpovídajícím názvem, načte ji. V opačném případě oznámí uživateli chybu o nenalezení požadované stránky. Tento koncept umožňuje jednoduché přidání nové nebo odebrání existující stránky.

Rámce, které se vyskytují se shodným obsahem na více stránkách, jsou uloženy v samostatných souborech. Soubory jsou poté vloženy na požadovaná místa pomocí php funkce include. Tímto způsobem zajistíme možnost editace příslušného kódu či textu pouze na jednom místě, což usnadní práci při úpravách aplikace.

Při tvorbě byly dodrženy všeobecné normy tvorby kódu, takže je stránka validní dle XHTML i CSS standardů. Zobrazení je shodné ve všech majoritních webových prohlížečích. Testováno v prohlížečích Google Chrome verze 12.0.742.122, Opera verze 11.50 a Internet Explorer verze 9.0.8112.16421.

---

<sup>19</sup> Open Database Connectivity je standardizované rozhraní sloužící k připojení k databázovým serverům

### 6.1.3 Vzhled

Vzhled každé webové stránky hraje klíčovou roli v jeho oblíbenosti. Pokud zvolíme originální, přehledný a příjemně barevně sladěný design, uživatelé se budou rádi vracet. Vytvořit takový vzhled stránky však není jednoduché, tvůrce musí disponovat představivostí a alespoň částečným grafickým cítěním. Za kvalitní návrh vzhledu zaplatíme u profesionálů i několik desítek tisíc korun.

Design aplikace je založen na klasickém třísloupcovém layoutu<sup>20</sup>, který přehledně člení stránku na jednotlivé části. Stránka začíná vycentrovaným logem umístěným v horní části. Pod logem jsou umístěny zmíněné sloupce a spodní část uzavírá zápatí. Menu je umístěné na levé straně webu, pravou stranu tvoří přehled aktualit a výčet mediálních partnerů, kteří se podílí na propagaci konference. Požadovaný obsah stránek je generován v centrální části.

Minimální šířka stránky je stanovena na 1000 pixelů<sup>21</sup>. Šířka byla stanovena s ohledem na standardní rozlišení monitorů 1024x768 px, takže poskytuje komfort při zobrazení drtivé většině uživatelů.



Obrázek 2 - Ukázka vzhledu aplikace

<sup>20</sup> Pojmeme layout označujeme grafické rozvržení určité oblasti

<sup>21</sup> Pixel (zkratka px) je základní jednotkou grafického zobrazení, představuje jeden bod na obrazovce

### 6.1.4 Uživatelé

V systému jsou rozlišeny tři typy uživatelů. Nepřihlášený uživatel má k dispozici veškeré informace, které aplikace nabízí - údaje o aktuálním ročníku, aktualizovaný ceník, seznam uživatelů aplikace, seznam registrovaných účastníků na nejbližší termín konference včetně seznamu přihlášených příspěvků, kontakt na organizátory a informace o místu konání.

Po registraci a přihlášení dostane uživatel možnost zadat či upravit své kontaktní údaje, přihlásit se na daný ročník konference, pokud jsou přihlášky povoleny, a uložit nebo vytisknout vygenerovanou proforma fakturu.

V případě, že má uživatel nastavená práva administrátora, může upravit informace o aktuálním ročníku konference nebo přidat další ročník. Dále má možnost přidat, upravit či smazat aktuality, které přehledným způsobem informují uživatele o nejnovějších změnách v systému. Samozřejmostí je přístup ke všem uživatelským přihláškám a příspěvkům na aktuální ročník konference.

Přehled jednotlivých uživatelských rolí zobrazuje Use Case Diagram<sup>22</sup>, umístěný v příloze A.

## 6.2 Adresářová struktura

Aplikace je rozdělena do několika adresářů, které obsahují soubory, zajišťující všechny potřebné funkce ke správnému chodu systému. V kořenovém adresáři je umístěn řídicí soubor index.php, který pracuje s ostatními soubory, uloženými v podadresářích. Základní struktura je zobrazena na Obrázku 3.

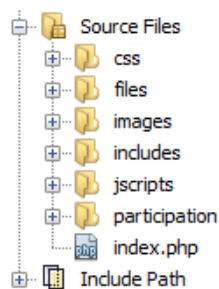
**Struktura adresářů zahrnuje:**

- css - všechny soubory s kaskádovými styly
- files - dokumenty přístupné uživatelům aplikace
- images - použité obrázky
- includes - zdrojové kódy php souborů
- jscripsts - zdrojové kódy JavaScriptových souborů
- participation - prezentace a příspěvky účastníků konference

---

<sup>22</sup> Diagram zobrazuje systém z vnějšího pohledu a pomáhá určit rozsah a hranice celého projektu





Obrázek 3 - Adresářová struktura

## 6.3 Návrh databáze

Databázovému návrhu předcházela důkladná analýza, která měla za úkol vytýčit všechny hranice systému a určit požadovaný rozsah datových struktur, které budou reprezentovat uložená data.

Aplikace do databáze ukládá uživatelská přihlašovací jména a hesla, všechny texty umístěné na webových stránkách, přiděluje uživatelům role s oprávněními, nastavuje všechny důležité údaje o aktuálním ročníku konference a ukládá přihlášky uživatelů a jejich prezentace, určené na aktuální konferenci. K dispozici byla přihláška z minulých ročníků, která obsahovala několik desítek vyplňovaných údajů, což předpovídalo, spolu s předchozími faktory, že bude databáze obsahovat větší množství relací.

Celý koncept byl navrhován s ohledem na budoucí rozšiřitelnost, kterou uplatníme především v možnosti snadného přidání dalšího jazyka k aktuální české a anglické lokalizaci. E-R diagram, vyjadřující celý koncept databáze, je k dispozici v příloze B.

## 6.4 Funkce

### 6.4.1 Registrace

Většina funkcí systému je k dispozici všem návštěvníkům. Pokud se uživatel rozhodne přihlásit na konferenci, musí nejdříve projít registračním procesem. Za úspěšnou registraci je považována taková, která splní vstupní požadavky. Prvním z požadavků je přihlašovací jméno, které se skládá minimálně z 8 znaků. V případě, že je uživatelské jméno již použito, bude uživatel upozorněn a požádán, aby zadal jméno nové. Spolu se jménem je nutné zadat uživatelské heslo, které je pro zvýšení bezpečnosti vyžadováno zadat dvakrát. Poslední podmínkou je uvedení emailu, který je kontrolován dle platného emailového tvaru. Pokud jsou správně vyplněny všechny vstupní údaje a heslo je správně potvrzeno, je registrace úspěšná, data jsou uložena do databáze a uživatel se ihned může přihlásit. Původně byl zvažován koncept dokončení registrace pomocí emailového ověření, ale tento krok byl vynechán, aby uživateli přinesl větší komfort.

## 6.4.2 Přihlášení na konferenci

Přihláška na konferenci, jak již bylo naznačeno, tvoří stěžejní část celé práce. Při registraci je uživatel tázán na mnoho údajů, které jsou při správném vyplnění formuláře uloženy do databáze. Uložené informace může uživatel využít při registraci na další ročník konference, kdy je registrační formulář vyplněný dříve zadanými daty.

Z důvodu velké obsáhlosti přihlášky byl registrační proces rozdělen do několika samostatných formulářů, které na sebe navazují a tvoří tak jeden celek. Dílčí kroky jsou interaktivní a vzájemně mezi sebou provázané. V případě, že uživatel zvolí specifické tlačítko v prvním formuláři, další strany s těmito údaji již počítají a přizpůsobí svůj obsah volbě uživatele. Např. pokud si uživatel zvolí, že se přihlašuje jako rodinný příslušník, nebude mu nabídnut formulář pro vyplnění příspěvku na konferenci.

### Popis průběhu registrace:

Před samotnou registrací je zobrazen formulář s uživatelskými údaji. (Obrázek 4) Pokud účastník vyplnil své kontaktní údaje dříve, může zde zkontrolovat, zda jsou korektně vyplněné a případně provést modifikaci.

Jméno	Tomáš		
Příjmení	Marný		
Titul(y) před jménem	doc. Ing.	za jménem	CSc.
Organizace, univerzita, fakulta	Univerzita Pardubice		
Pracoviště, ústav, katedra	Fakulta elektrotechniky a informatiky		
Ulice	Legie	Číslo popisné	666
Město	Pardubice	PSČ	12345
Stát	Česká republika ▼		
Telefon	+420 112 567 007		

Obrázek 4 - Kontrola osobních údajů

V prvním kroku registrace uživatel vybere, jestli se plánuje konference přímo zúčastnit nebo si pouze přeje inzerovat ve sborníku. (Obrázek 5) V případě, že se konference zúčastní, vybere, jestli má zájem o přednes prezentace či ukázkou exponátu. Příspěvek na konferenci přihlašuje vždy pouze jeden z autorů, ostatní vyberou možnost účasti jako spoluautoři. Poslední přípustnou volbou je registrace jako doprovod či rodinný příslušník.

<b>Mám zájem zúčastnit se konference</b>	
<input checked="" type="radio"/>	s prezentací (přednáškou, vývěskou, exponátem)
<input type="radio"/>	bez prezentace nebo jako spoluautor
<input type="radio"/>	jako doprovod (rodinný příslušník)
<input type="radio"/> publikovat za úhradu ve sborníku bez účasti na konferenci	
<input checked="" type="checkbox"/> inzerovat za úhradu ve sborníku	

Obrázek 5 - Registrační formulář pro zvolení účasti

Druhým krokem je přihlášení přednášky, vývěsky nebo exponátu s účastí na konferenci, anebo publikace ve sborníku bez účasti. (Obrázek 6) Tento formulář je nabídnut pouze autorům příspěvku nebo uživatelům, kteří si přejí inzerci ve sborníku. Každý účastník má možnost přihlásit až tři příspěvky. Tento počet byl zvolen z poznatků z předchozích ročníků. Uživatel vyplní název svého příspěvku anglicky, všechny jeho autory a vybere, do kterého tématického celku bude zařazen. K dispozici je volba pro uložení souboru na webový server, kterou uživatel nemusí využít a může uložení souboru odložit na později.

<b>Přihlašuje vždy jen jeden z autorů!</b>	
<b>1. Přihlašuji</b>	<input checked="" type="radio"/> přednášku <input type="radio"/> vývěsku <input type="radio"/> exponát <input type="radio"/> publikaci bez účasti
<b>Název anglicky (max. 80 znaků)</b>	
Taxonomie výpočetních modelů neuronových sítí	
<b>Autor, autoři (ve tvaru Příjmení, J.)</b>	
Marný, T.	Vomočil, K.
<b>Prosím zařadit do tématického celku</b>	
<input type="radio"/> A <input type="radio"/> B <input checked="" type="radio"/> C <input type="radio"/> D <input type="radio"/> E <input type="radio"/> F	
<b>Příspěvek mohu přednést v angličtině</b>	
<input checked="" type="radio"/> Ano <input type="radio"/> Ne <input type="text" value="Umělá inteligence"/>	

Obrázek 6 - Registrační formulář pro přihlášení příspěvku

Další formulář je určen účastníkům, kteří požadují ubytování. (Obrázek 7) V horní části uživatel zvolí, jestli má zájem o ubytovací služby. V případě kladného výběru vyplní formulář se svojí adresou trvalého bydliště, v opačném případě pokračuje v registraci dále.

Údaje pro ubytování			
<input checked="" type="radio"/> Mám zájem o ubytování <input type="radio"/> Nemám zájem o ubytování			
Adresa bydliště			
Ulice	<input type="text" value="J. K. Tyla"/>	Číslo popisné	<input type="text" value="23/6b"/>
Město	<input type="text" value="Hradec Králové"/>	PSČ	<input type="text" value="234 56"/>
Datum narození	<input type="text" value="1.7.1973"/>	Místo narození	<input type="text" value="Hradec Králové"/>
Stát	<input type="text" value="Česká republika"/> ▼		
Číslo OP	<input type="text" value="12345678"/>		

Obrázek 7 - Registrační formulář pro zadání ubytovacích údajů

Předposledním registračním krokem je potvrzení výše hrazené částky. (Obrázek 8) Ve formuláři jsou přístupná pouze některá políčka, ostatní jsou vyplněná dle předchozích voleb. Uživatel aplikace si na tomto místě může zvolit, jestli má zájem o více výtisků sborníku nebo kterou exkurzi si přeje navštívit. Dále si může vybrat, ve kterých dnech si přeje být ubytován a jestli souhlasí s ubytováním dle dispozic organizátora, nebo si zvolit samostatné ubytování s příplatkem. Ubytovací dny jsou zobrazeny dle systémového nastavení začátku a konce aktuální konference.

Výsledná částka je ihned po zvolení požadované možnosti přepočítána. Uživatel si může zvolit přepočet v českých korunách nebo eurech. Pro korektní funkci dynamického přepočítávání formuláře je nutné mít zapnutý JavaScript ve webovém prohlížeči.

Platby a požadované služby		
<input checked="" type="radio"/> Plné vložné (včetně sborníku)		
<input type="radio"/> Zlevněné vložné (studenti bez sborníku)		3700
<input type="radio"/> Doprovod - rodinný příslušník		
Publikace příspěvku ve sborníku bez účasti	<input type="text" value="0"/>	0
Samostatný sborník	<input type="text" value="1"/>	100
<input checked="" type="checkbox"/> Inzerce ve sborníku		2500
<input type="checkbox"/> Výstava exponátu, programového produktu		0
Ubytování s plnou penzí <input checked="" type="checkbox"/> 03/04 <input checked="" type="checkbox"/> 04/05 <input checked="" type="checkbox"/> 05/06		
<input type="radio"/> Souhlasím s ubytováním podle dispozic organizátora		
<input checked="" type="radio"/> Přeji si samostatné ubytování s příplatkem		450
Přeji si být, po vzájemné dohodě, ubytován s	1. <input type="text"/>	2. <input type="text"/>
Mám zájem o exkurzi		
<input checked="" type="checkbox"/> na EVP		
<input type="checkbox"/> do pivovaru Hanušovice		
<input type="checkbox"/> do papírny Velké Losiny		
<input checked="" type="checkbox"/> o koncert žáků ZUŠ		
o prodloužený víkend - ubytování		
<input type="checkbox"/> 31/01	<input type="checkbox"/> 01/02	<input type="checkbox"/> 02/03
<input checked="" type="checkbox"/> 06/07	<input checked="" type="checkbox"/> 07/08	<input type="checkbox"/> 08/09
<b>Celkem</b>		<b>8880</b>

Obrázek 8 - Registrační formulář pro vyplnění platby a požadovaných služeb

Závěr registrace tvoří vyplnění formuláře údaji pro vystavení faktury. (Obrázek 9) V případě, že je stránka zobrazena v českém jazyce, je uživateli nabídnuta možnost vyplnění IČO a DIČ. Pro zahraniční klienty je nabídka změněna pouze na číslo VAT<sup>23</sup>, což je obdobou českého daňového identifikačního čísla.[18] Identifikační číslo organizace je pouze českou záležitostí, tento údaj není v zahraničí používán, jedinou výjimku tvoří Slovensko.[19]

<sup>23</sup> Value Added Tax Identification Number je obdoba českého DIČ

Údaje pro vystavení faktury		
Organizace, univerzita, fyzická osoba	Univerzita Pardubice	
Útvar, fakulta	Fakulta elektrotechniky a informatiky	
Oddělení, ústav, katedra	Katedra softwarových technologií	
Adresa plátce		
Ulice	Legie	Číslo popisné 666
Město	Pardubice	PSČ 12345
Stát	Česká republika ▼	
IČO	1234567891	
DIČ	1234567892	
Poznámky, vzkazy, dotazy	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Jsem rád, že jsem se mohl registrovat na konferenci.</p> </div>	

Obrázek 9 - Registrační formulář pro zadání údajů za účelem vystavení faktury

### 6.4.3 Práce s měnou

Protože je konference otevřená pro účastníky nejen z České republiky, ale také z ostatních zemí Evropské unie, je uživatelům nabídnuta možnost volby měny k úhradě registračního poplatku. Na výběr je k dispozici česká koruna nebo euro.

Ceny jsou v databázi uloženy v českých korunách. Pokud je zvolen přepočítání na eura, systém se připojí na server České národní banky a zjistí aktuální kurz. Zjištěný kurz je zaokrouhlen na celé číslo směrem nahoru a poté přepočítá výši eur, potřebných k zaplacení daného poplatku. V případě, že se nepodaří připojit na server ČNB, je kurz nahrazený konstantou 25,- Kč, která lze jednoduše změnit při velké odchylce kurzu eura oproti české koruně.

**Funkce pro zjištění aktuálního kurzu eura:**

```
function zjistKurz() {
    Define("KURZ",
    "http://www.cnb.cz/cs/financni_trhy/devizovy_trh/kurzy_devizove
    ho_trhu/denni_kurz.txt");
    if (httpFileExists(KURZ)) {
        $kurzy = @file(KURZ);
        foreach ($kurzy as $radek) {
```

```

        $sloupec = explode("|", $radek);
        if ($sloupec[3] == "EUR") {
            return ceil(floatval(str_replace(",", ".",
$sloupec[4])));
        }
    } else {
        return 25;
    }
}

```

#### 6.4.4 Aktuality

Pravou stranu systému tvoří rámec s aktualitami, ve kterém jsou přehledně zobrazeny všechny zásadní informace o termínech registrace a další důležité zprávy. Každá zpráva obsahuje datum a čas přidání do systému, zvýrazněný nadpis a krátký text o maximální délce 1000 znaků.

Pokud je přihlášenému uživateli přiřazena administrátorská role, zobrazí se na spodní straně rámce nabídka s možností úpravy aktualit. Při této volbě je přesměrován na přehledný seznam novinek, kde je k dispozici přidání nové aktuality a editace nebo smazání stávajících aktualit.

#### 6.4.5 Lokalizace

Pojmem lokalizace rozumíme proces překladu systému a celkové přizpůsobení obsahu nové jazykové verzi. V této práci tvoří velmi důležitou roli, protože se jedná o aplikaci určenou pro konferenci se zájmem i v zahraničí. Jedním z důvodů je možnost přednášet přihlášené prezentace a příspěvky v anglickém jazyce.

Při tvorbě webové aplikace má autor k dispozici několik možností, kterými lze řešit překlad. K práci bylo využito řešení pomocí SQL databáze. Vzhledem k velikosti aplikace a předpokládané návštěvnosti stránek není nutné řešit optimalizaci rychlosti přístupu k jednotlivým překládaným textům, databázové řešení je v tomto případě zcela vyhovující. V případě, že by se od systému očekávalo několikanásobně větší zatížení, muselo by být provedeno důkladnější testování a případně zvoleno odlišné řešení.

Veškeré texty na stránce včetně menu a chybových hlášek jsou nahrávány z databáze. Jedinou výjimku tvoří chybové upozornění o neúspěšném připojení k databázi. Zvolené řešení umožňuje snadné přidání libovolného jazyka. Uživatel si může vybrat mezi českým a anglickým jazykem. Pokud systém z použitého URL nerozezná platný typ zvoleného jazyka, je uživateli nabídnut český jazyk, který je nastavený jako výchozí.

## 6.4.6 Fakturace

Registrace na konferenci pořadatelé nezaručí, že se přihlášený uživatel skutečně zúčastní. Proto bylo od systému požadováno, aby umožnil výstup s vygenerovanou proforma fakturou. Zaplacením účastnického poplatku uživatel potvrdí svůj zájem o účast na konferenci. Plnohodnotná faktura je účastníkům vystavena na místě konání.

Po úspěšném dokončení registrace je uživateli v přehledu registračních údajů nabídnuta volba generování faktury. Každé faktuře je přiřazeno unikátní číslo, které je také obsaženo v posledním trojčíslí variabilního symbolu, na základě kterého je identifikován registrovaný účastník. Výsledek je zobrazený v HTML formátu na samostatné stránce, aby byl umožněn případný tisk v přehledné formě.

Řízení procesů 2012		Proforma faktura - daňový doklad č. 2011 / 001	
<b>Dodavatel:</b> Česká společnost průmyslové chemie pobočka Univerzita Pardubice nám. Čs. legií 565 532 10 Pardubice  IČO: 60160322 DIČ: CZ60160322 Společnost registrována u MVČR č. VSP/1-1089/90R Pobočka r.č. 07-06-12		<b>Odběratel:</b> Univerzita Pardubice Legie 666 123 45 Pardubice Česká republika  IČO: 1234567891 DIČ: 1234567892	
<b>Peněžní ústav:</b> Česká spořitelna pobočka Pardubice Palackého tř. 1908, 530 00 Pardubice  SWIFT CODE: GIBA CZ PX IBAN code: CZ43 0800 0000 0012 0317 1349  č.ú. 1203171349/0800 variabilní symbol: 31361001 konstantní symbol: 0308		Datum vystavení: 11.08.2011 Datum zdan. plnění: 11.08.2011 Datum splatnosti: 02.04.2012 Forma úhrady: převodem	
Účtujeme Vám účastnický poplatek na vědecko-technickou konferenci s mezinárodní účastí Řízení procesů 2012, která se koná v Koutech nad Desnou v termínu 03.06.2012 - 06.06.2012 za  <b>doc. Ing. Tomáš Marný, CSc.</b> Fakulta elektrotechniky a informatiky Katedra softwarových technologií			
Označení položky	Množství	Jedn. cena	Celkem
Plně vložné (včetně sborníku)	1	3700 Kč	3700 Kč
Inzerce ve sborníku	1	2500 Kč	2500 Kč
Samostatný sborník	1 ks	100 Kč	100 Kč
Ubytování s plnou penzí	3 dny	710 Kč	2130 Kč
Příplatek za samostatné ubytování	3 dny	150 Kč	450 Kč
Částka k úhradě			<b>8880 Kč</b>
Prosíme, nekumulujte platby za několik účastníků. U každé platby nebo položky uvádějte variabilní symbol, abychom mohli jednoznačně identifikovat za koho byla platba uhrazena. Děkujeme.			
Dodavatel není plátcem DPH.		_____ Razítko a podpis	

Obrázek 10 - Ukázka vygenerované proforma faktury



## 6.5 Zabezpečení

### 6.5.1 Formuláře

Všechna zadávaná uživatelská data jsou na vstupu kontrolována, aby byl zajištěn správný chod celého systému. Zadané emailové adresy musí splňovat standardizovaný tvar, u číselných údajů je kontrolováno, jestli jsou obsaženy v požadovaném číselném intervalu nebo jestli bylo zadáno skutečně číslo a u přihlašovacích údajů je ověřována minimální délka uživatelského jména a hesla. Kontrola vstupních dat je na místech, kde byla možná, provedena pomocí Validation Engine (viz kapitola 3.4.2) a ostatní vstupy jsou ošetřeny pomocí PHP.

Pokud by byly vstupní údaje sbírány pouze na území České republiky, bylo by možné nasadit mnohem striktnější filtry, než byly nasazeny. V případě mezinárodních systému musí být autor obezřetnější a ponechat uživatelům jistou volnost, aby pokryl odchylky od struktury dat, zadávaných na domácím území.

### 6.5.2 SQL Injection

Jedná se o metodu útoku na databázi vsunutím vlastního SQL kódu s podvrženými daty do standardního vstupu formuláře. Tento kód je pak zpracován a změní chování původního příkazu. Pokud by nebyla aplikace ošetřena proti SQL Injection, hrozilo by riziko ztráty citlivých uživatelských dat do nepovolaných rukou útočníka a případné odstavení celého systému.

K ošetření byla použita funkce `oci_bind_by_name` z PHP knihovny OCI8.

#### Příklad ošetření SQL Injection:

```
function vlozAktualitu($conn, $title, $content) {
    if (pripojeno($conn)) {
        $insert = 'INSERT INTO aktuality (id_aktuality,
datum, nadpis, text) VALUES (news_seq.nextval, sysdate,
:title_bind, :content_bind)';
        $stid = oci_parse($conn, $insert);

        oci_bind_by_name($stid, ":title_bind", $title);
        oci_bind_by_name($stid, ":content_bind", $content);

        if(oci_execute($stid)) {
            echo "Novinka úspěšně odeslána!";
        } else {
            echo 'Odeslání novinky se nezdařilo!';
        }
    }
}
```

```

        unset ($insert);
    }
}

```

### 6.5.3 Transakce

Transakce je skupina databázových příkazů, které spolu tvoří atomický celek. V případě neúspěšného provedení jakékoliv operace uvnitř transakce je proveden tzv. rollback a všechny provedené změny v rámci dané transakce jsou vráceny do původního stavu. Toho využíváme především v případech, kdy se jedná o rizikové operace. Všechna místa, na kterých jsou provedeny minimálně dva po sobě jdoucí SQL dotazy, jsou ošetřena pomocí transakcí.

#### Příklad ošetření SQL dotazů pomocí transakce:

```

        $stid = oci_parse($conn, "INSERT INTO platba
(id_platby, id_registrace, id_vlozne, pocet_sborniku)
VALUES(".$id.", ".$id.", ".$vlozne[0]."', ".$textbook.'")");
        $res = oci_execute($stid, OCI_NO_AUTO_COMMIT);

        $stid = oci_parse($conn, "INSERT INTO ubytovani
(id_ubytovani, id_platby, samostatne, spolubydlici1,
spolubydlici2, den1, den2, den3, den4, den5, den6, den7, den8,
den9) VALUES(".$id.", ".$id.", ".$samostatne.",
'".$roommate1."', '".$roommate2."', '".$acc1."', '".$acc2."',
'".$acc3."', '".$first."', '".$second."', '".$third."',
'".$acc4."', '".$acc5."', '".$acc6.'")");
        $res = oci_execute($stid, OCI_NO_AUTO_COMMIT);
        if (!$res) {
            oci_rollback($conn);
        }

        $res = oci_commit($conn);
        if (!$res) {
            $e = oci_error($conn);
            trigger_error(htmlentities($e['message']),
E_USER_ERROR);
        }

```

Na příkladu je znázorněna část kódu, ve které jsou ukládány registrační údaje do tabulek platba a ubytování. V případě, že se dotaz nezdaří, je zavolána funkce `oci_rollback`, databáze je vrácena do původního stavu a na obrazovku je vypisována chybová hláška.

#### 6.5.4 Šifrování

Při registraci do systému je uživatel povinen zadat své uživatelské jméno a heslo. Pro zvýšení bezpečnosti jsou hesla zašifrována.

K převodu je použita hashovací funkce Secure Hash Algorithm, zkráceně SHA, která zašifruje vstupní řetězec na tzv. hash. Výsledný řetězec je také znám pod názvem otisk a je tvořen 40 hexadecimálními znaky. Uživatelem zadané heslo při přihlášení je opětovně zašifrováno, v tomto formátu odesláno do databáze a pokud je potvrzena shoda nově generovaného otisku s otiskem původním, je uživatel úspěšně přihlášen.[20]

## 7 Závěr

Cílem práce byla realizace webové aplikace, která usnadní práci nejen účastníkům odborné konference, ale také jejím organizátorům. Informační systém poskytuje základní služby, které jsou potřebné k pořádání konferencí a splňuje veškeré požadavky, které byly na systém kladeny. Pořadatelům konference nabízí nástroje k přehlednému zobrazení souhrnných dat o počtu účastníků a množství příspěvků, které budou na konferenci k dispozici. Dále umožňuje předběžně zajistit potřebnou ubytovací kapacitu na základě zjištěných dat a také souhrnnou cenu všech účastnických poplatků.

Aplikace je určena pro nasazení do provozu, takže plánuji s jejím vývojem pokračovat i v budoucnosti. Je mnoho oblastí, ve kterých by bylo vhodné systém rozšířit. Jako příklad bych uvedl export přihlášených účastníků do XML formátu, automatické generování sborníku z uložených příspěvků, přidání RSS kanálu pro možnost odběru aktualit či zavedení fotogalerie. Při testování systému navíc není možné odhalit sto procent chyb, takže předpokládám, že bude potřeba dodělat drobné úpravy a záplaty i za provozu.

Po kompletní implementaci celého systému jsem si vědom, kolik času zabrala tvorba těch nejzákladnějších částí aplikace, a proto obdivuji komplexní webové systémy, které nabízejí několikanásobně větší počet funkcí.

Největší problém jsem zaznamenal při grafickém návrhu. Víím, že se nejedná o skvost mezi designem webových stránek, a pokud bych měl někdy tvořit podobný projekt, zvolil bych raději spolupráci s grafikem.

Práce pro mě byla přínosem, vyzkoušel jsem si vytvoření informačního systému od samotného základu až po kompletní dokončení. Jsem zvědav, jakým způsobem proběhne nasazení do provozu a na případný ohlas uživatelů.

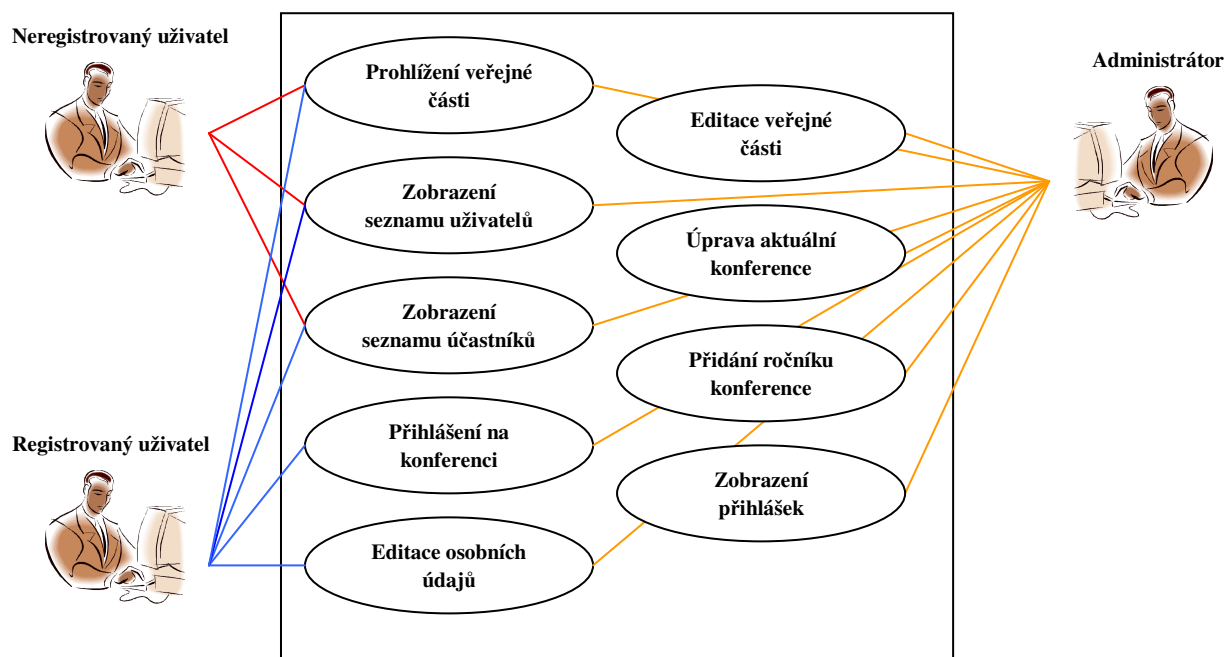
## Použitá literatura a zdroje

1. Extensible HyperText Markup Language. Wikipedie, otevřená encyklopedie. [Online] 27. 6. 2011. [Citace: 20. 7. 2011.] Dostupný z WWW: <[http://cs.wikipedia.org/wiki/Extensible\\_HyperText\\_Markup\\_Language](http://cs.wikipedia.org/wiki/Extensible_HyperText_Markup_Language)>.
2. Kaskádové styly. Wikipedie, otevřená encyklopedie. [Online] 21. 6. 2011. [Citace: 20. 7. 2011.] Dostupný z WWW: <[http://cs.wikipedia.org/wiki/Kask%C3%A1dov%C3%A9\\_styly](http://cs.wikipedia.org/wiki/Kask%C3%A1dov%C3%A9_styly)>.
3. PHP. Wikipedie, otevřená encyklopedie. [Online] 20. 6. 2011. [Citace: 21. 7. 2011.] Dostupný z WWW: <<http://cs.wikipedia.org/wiki/PHP>>.
4. JavaScript. Wikipedie, otevřená encyklopedie. [Online] 18. 7. 2011. [Citace: 21. 7. 2011.] Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Javascript>>.
5. World Wide Web. Wikipedie, otevřená encyklopedie. [Online] 1. 6. 2011. [Citace: 21. 7. 2011.] Dostupný z WWW: <[http://cs.wikipedia.org/wiki/World\\_Wide\\_Web](http://cs.wikipedia.org/wiki/World_Wide_Web)>.
6. jQuery. Wikipedie, otevřená encyklopedie. [Online] 29. 6. 2011. [Citace: 21. 7. 2011.] Dostupný z WWW: <<http://cs.wikipedia.org/wiki/JQuery>>.
7. AJAX. Wikipedie, otevřená encyklopedie. [Online] 22. 3. 2011. [Citace: 22. 7. 2011.] Dostupný z WWW: <<http://cs.wikipedia.org/wiki/AJAX>>.
8. DUGAS, Cédric. A jQuery inline form validation, because validation is a mess. Position Absolute. [Online] 20. 10. 2010. [Citace: 22. 7. 2011.] Dostupný z WWW: <<http://www.position-absolute.com/articles/jquery-form-validator-because-form-validation-is-a-mess/>>.
9. LACKO, Luboslav. *SQL : Kapesní přehled*. 1. vyd. Brno: Computer Press, 2005. 96 s. ISBN 80-251-0788-4.
10. Welcome to the NetBeans Community. Netbeans. [Online] 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <<http://netbeans.org/about/index.html>>.
11. Open source software. Wikipedie, otevřená encyklopedie. [Online] 14. 5. 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <[http://cs.wikipedia.org/wiki/Open\\_source\\_software](http://cs.wikipedia.org/wiki/Open_source_software)>.
12. Freeware. Wikipedie, otevřená encyklopedie. [Online] 28. 6. 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <<http://cs.wikipedia.org/wiki/Freeware>>.
13. Toad Data Modeler. Quest software. [Online] 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <<http://www.quest.com/toad-data-modeler/>>.

14. Oracle SQL Developer Overview. Oracle. [Online] 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <<http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index.html>>.
15. Apache HTTP Server. Wikipedia, the Free Encyclopedia. [Online] 19. 7. 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <[http://en.wikipedia.org/wiki/Apache\\_HTTP\\_Server](http://en.wikipedia.org/wiki/Apache_HTTP_Server)>.
16. Oracle Database. Wikipedia, the Free Encyclopedia. [Online] 21. 7. 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <[http://en.wikipedia.org/wiki/Oracle\\_Database#cite\\_ref-13](http://en.wikipedia.org/wiki/Oracle_Database#cite_ref-13)>.
17. Oracle Database 10g Express Edition Overview. Oracle. [Online] 2011. [Citace: 23. 7. 2011.] Dostupný z WWW: <<http://www.oracle.com/technetwork/database/express-edition/overview/index.html>>.
18. Value added tax identification number. Wikipedia, the Free Encyclopedia. [Online] 29. 7. 2011. [Citace: 31. 7. 2011.] Dostupný z WWW: <[http://en.wikipedia.org/wiki/Value\\_added\\_tax\\_identification\\_number](http://en.wikipedia.org/wiki/Value_added_tax_identification_number)>.
19. Identifikační číslo osoby. Wikipedie, otevřená encyklopedie. [Online] 3. 7. 2011. [Citace: 31. 7. 2011.] Dostupný z WWW: <<http://cs.wikipedia.org/wiki/I%C4%8CO>>.
20. Secure Hash Algorithm. Wikipedie, otevřená encyklopedie. [Online] 5. 4. 2011. [Citace: 2. 8. 2011.] Dostupný z WWW: <[http://cs.wikipedia.org/wiki/Secure\\_Hash\\_Algorithm](http://cs.wikipedia.org/wiki/Secure_Hash_Algorithm)>.

## Příloha A – Use Case Diagram

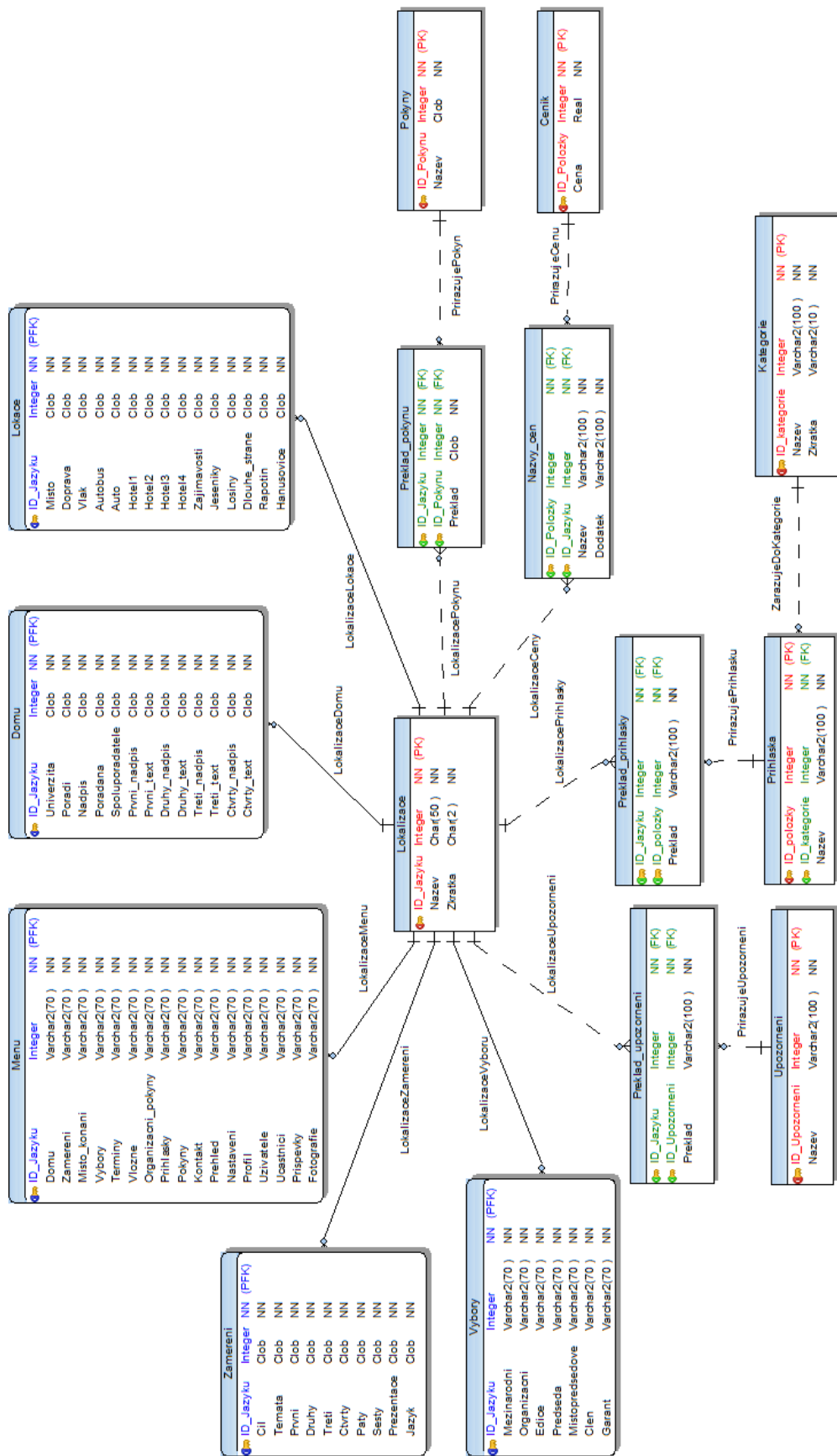
Diagram znázorňuje chování uvnitř modelovaného systému. Jeho cílem je znázornit funkčnost aplikace z pohledu jednotlivých uživatelů.



Obrázek 11 - Use Case diagram







Obrázek 13 - Entity-relationship diagram znázorňující strukturu lokalizace aplikace

## **Příloha C – Obsah přiloženého CD**

Obsah kompaktního disku tvoří:

- Kompletní implementaci aplikace včetně zdrojových kódů.
- E-R diagram znázorňující databázovou strukturu.
- SQL příkazy umožňující vytvoření databázové struktury.
- Text bakalářské práce v elektronické podobě.