

Databázová aplikace pro vyhodnocení přijímacího řízení

Database application for evaluation of admission exam

Lubomír Gebauer, DiS.



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Lubomír GEBAUER, DiS.**
Osobní číslo: **A08684**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**

Téma práce: **Databázová aplikace pro vyhodnocení přijímacího řízení**

Zásady pro vypracování:

1. Seznamte se s problematikou aplikací pro hodnocení výběrového řízení.
2. Prostudujte technologii PHP, HTML.
3. Realizujte funkcionalitu vstupu dat do aplikace – hromadný import uchazečů do systému, jednotlivé vkládání uchazečů.
4. Zpracujte funkcionalitu pro správu záznamů (třídění, filtry, mazání, úprava).
5. Vypracujte možnosti řešení funkcí pro hromadný export uchazečů podle kritérií.
6. Navrhněte možnosti automatizace hodnocení uživatelů o práci.
7. Realizovaný systém vyhodnoťte.

Rozsah bakalářské práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. M. SCHURMAN, Eric; J. PARDI, William . Dynamicé HTML v akci. Hornocholupická 22, Praha 4 : Computers Press, 2000. 421 s. ISBN 80-72226-401-X.
2. RESIG, John. JavaScript a Ajax : Moderní pogramování webových aplikací. Brno : Computer Press a.s., 2007. 360 s. ISBN 978-80-251-1824-5.
3. CROFT, Jeff; LLOYD, Lan; RUBIN, Dan. Mistrovství v CSS : Pokročilé techniky pro webové desingéry a vývojáře. Holandská 8, Brno : Computer Press, a.s., 2007. 409 s. ISBN 978-80-251-1705-7.
4. NARAMORE, Elizabeth, et al. Vytváříme webové aplikace v PHP, MySQL a Apache. Nám. 28. dubna 48, Brno : Computer Press, a.s., 2006. 812 s. ISBN 80-251-1073-7.
5. WELLING, Luke; THOMPSONOVÁ, Laura. PHP a MySQL – rozvoj webových aplikací. Druhé vydání. Pod Křížem 15, Praha 8 : SoftPress s.r.o., 2004. 910 s. ISBN 80-86497-60-7.
6. KOSEK, Jiří. Téměř vše o WWW [online]. 2009–2010 [cit. 2011-02-03]. Dostupné z WWW: www.kosek.cz.
7. JANOVSKEÝ, Dušan. Jak psát web – návod na html stránky [online]. –, 2011 [cit. 2011-02-03]. Dostupné z WWW: www.jakpsatweb.cz. ISBN 1801-0458.

Vedoucí bakalářské práce:

Ing. Petr Šilhavý, Ph.D.

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

25. února 2011

Termín odevzdání bakalářské práce:

7. června 2011

Ve Zlíně dne 25. února 2011



prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Tématem této bakalářské práce je tvorba programu pro střední školy a jejich přijímací řízení bez konání přijímacích zkoušek. Práce se z teoretického hlediska zaměřuje nejdříve na základní teorii kódu, který je při tvorbě programu použit, dále na principy a postupy přijímacího řízení na základě vyhodnocení uchazečů z nižšího stupně vzdělávání. V praktické části jsou tyto postupy zavedeny aplikačně do programu.

Klíčová slova: Webový server, databázový server, SQL dotaz, tagy, entita, přijímací řízení

ABSTRAKT

The goal of this thesis is to create application for intermediate schools and their admissions without entrance examinations. Theoretical part of the thesis first focuses on the basic theory of code that is used during application creation, then on the principles and procedures of entrance examination evaluation of candidates from primary schools. In the practical part of the thesis are these principles and procedures implemented in the application.

Keywords: Web server, database server, SQL query, tags, entity, entrance examination

Veliké poděkování patří ing. Petru Šilhavému za odborné rady a čas, který mi věnoval, a také rodině, která mě po celou dobu studia podporovala.

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 jsem seznámen 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.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH:

ÚVOD	10
I. TEORETICKÁ ČÁST	11
1 TECHNOLOGIE PHP, HTML, MYSQL	12
1.1 ZNAČKOVACÍ JAZYK HTML.....	12
1.2 JAVASCRIPT	14
1.3 SKRIPTOVACÍ JAZYK PHP	15
1.4 DATABÁZOVÝ SYSTÉM MYSQL	16
1.5 WEBOVÝ SERVER.....	18
2 NÁVRH APLIKACE	19
2.1 ZÁKLADNÍ ÚVAHA APLIKACE.....	19
2.2 POSLOUPNOST ÚKONŮ V PŘIJÍMACÍM ŘÍZENÍ.....	19
2.2.1 První fáze – vyhlášení přijímacího řízení.....	19
2.2.2 Druhá fáze – vkládání záznamů	20
2.2.3 Třetí fáze – vyhodnocení.....	20
2.3 POŽADAVKY APLIKACE.....	21
2.3.1 Víceuživatelský přístup	21
2.3.2 Přístup do aplikace z počítačové sítě	21
2.3.3 Zabezpečení dat o uchazeči.....	21
2.3.4 Tisk seznamů a export dat.....	22
2.4 DIAGRAM PŘÍPADU UŽITÍ	23
2.5 NÁVRH DATABÁZE	24
II. PRAKTICKÁ ČÁST	25
3 TVORBA APLIKACE	26
3.1 STRUKTURA APLIKACE	26
3.2 SPRÁVA ZÁZNAMŮ O PŘIJÍMACÍM ŘÍZENÍ	26
3.2.1 Kontrola dodržení správné posloupnosti.....	27
3.2.2 Zabezpečení proti přerušení relace mezi záznamy v tabulkách	27
3.3 VYHODNOCENÍ	30
3.4 IMPORT ZÁZNAMŮ	31
3.5 EXPORT ZÁZNAMŮ DO FORMÁTU CSV	33
3.6 EXPORT ZÁZNAMŮ DO FORMÁTU PDF.....	34

ZÁVĚR	36
ENDING	37
SEZNAM POUŽITÉ LITERATURY.....	38
SEZNAM OBRÁZKŮ	39
SEZNAM PŘÍKLADŮ PROGRAMOVÝCH KÓDŮ.....	40
SEZNAM PŘÍLOH.....	41

ÚVOD

V této práci se zabývám tvorbou aplikace vytvořenou pro střední školy, jež přijímají uchazeče bez konání přijímacích zkoušek, tedy jen podle prospěchu předchozího vzdělání, které se hodnotí na základě ředitelem stanovených kritérií. Aplikace by měla spravovat mnoho záznamů o uchazeči jako i výsledky přijímacího řízení. Proto je nutné, aby aplikace používala databázi, která bude potřebná data nabízet nebo uchovávat. Výsledná aplikace zahrne všechny možné úkony spojené s vyhodnocováním přijímacího řízení bez přijímacích zkoušek. Dále se zaměřím na zjednodušení vkládání jednotlivých uchazečů pomocí importu dat, a také na možnosti pro další jiné zpracování výsledných pomocí exportu. Velikým užitekem aplikace je pro uživatele odstranění možných početních chyb či jiných chyb při ručním nebo jiném hodnocení uchazečů a v nepodstatné řadě velká úspora času při vyhodnocování záznamů.

Přijímací řízení bez konání přijímacích zkoušek je dnes novým trendem pro mnoho středních škol, a to hlavně díky klesající populaci žáků odcházejících ze základních na střední a učňovské školy. Všechny střední školy se snaží naplno využít svou kapacitu tím, že do prvních ročníků přijímají maximální počet žáků. A protože je většina škol příspěvkovou organizací, která svůj chod udržuje pomocí dotací od zřizovatele (stát, kraj, obec) a výše této dotace je podle počtu žáků ve škole, je pro chod školy klíčovým ukazatelem právě počet žáků. To je právě důvodem, že mnoho středních škol přijímá žáky bez přijímacích zkoušek. Studijní předpoklady uchazečů si tedy škola třídí podle dosažených výsledků z osmé (závěrečné vysvědčení) a deváté třídy (pololetní vysvědčení) základní školy. Potom na základě zadaných „Kritérií přijímacího řízení“ ředitel školy rozhodne o pořadí přijatých či nepřijatých žáků.

I. TEORETICKÁ ČÁST

1 TECHNOLOGIE PHP, HTML, MYSQL

1.1 ZNAČKOVACÍ JAZYK HTML

Jazyk HTML (Hyper Text Markup Language) je určen pro tvorbu webových dokumentů, které zobrazuje internetový prohlížeč.

HTML jako značkovací jazyk formátuje text a další objekty pomocí značek, kterým se říká tagy. Tyto tagy se ohraničují ostrými závorky $\langle \rangle$ a vkládají se přímo do textu. Uvnitř se může nacházet atribut, jehož úkolem je více specifikovat formátování daného objektu. Výsledný kód je pak pouhý text, který je obalen značkami. Celý kód i s textem přijme klient (webový prohlížeč), ten tagy rozezná a daný text a další objekty naformátuje. Výsledkem je pak zobrazená stránka textově a graficky upravená podle programátora. Bohužel, protože se HTML kód zpracovává na straně klienta, nese tedy prohlížeč plnou zodpovědnost nad zobrazeným dokumentem a z tohoto důvodu se může stát, že každý prohlížeč může daný prvek na stránce zobrazit trochu jinak. Toto je jedna z nevýhod kódování HTML dokumentů. Aby se kód HTML zobrazoval ve všech klientech stejně, musí být optimalizovaný pro všechny běžně dostupné www prohlížeče.

Příklady některých tagů a jejich atributů:

Tag pro **odkaz** `<a>` ``

- href – atribut pro cíl odkazu
- name – jméno záložky
- target - cílový rám

Tag pro **obrázek** ``

- width – šířka
- height – výška
- src – adresa k obrázku
- alt – alternativní popis

CSS

V dnešní době se HTML kód doplňuje kaskádovými styly, které se používají pro stylování vlastností textů nebo jiných objektů v HTML dokumentu. Kaskádové styly označované

jako CSS jsou přeloženy z Cascading Style Sheet. Formátování pomocí kaskádových stylů nese mnoho výhod, pro které se stává daleko používanější než formátování pomocí přímých HTML tagů. Mezi tyto výhody se řadí tvorba tzv. třídy stylu, vrstvení definice stylu, jednoduché a rychlé formátování většího množství objektů.

Pomocí CSS je také možná tvorba či formátování objektů takovým způsobem, který běžný HTML kód nedokáže (pozicování, nastavení průhlednosti,...).

viz příklad třídy stylu hlCast

```
#hlCast{  
    float: left;  
    width: 100%;  
    min-height: 450px;  
    background-color:rgb(0,112,80);  
    color: rgb(255,255,255)  
}
```

Zdrojový kód 1 - Použití kaskádových stylů

SEO Optimalizace

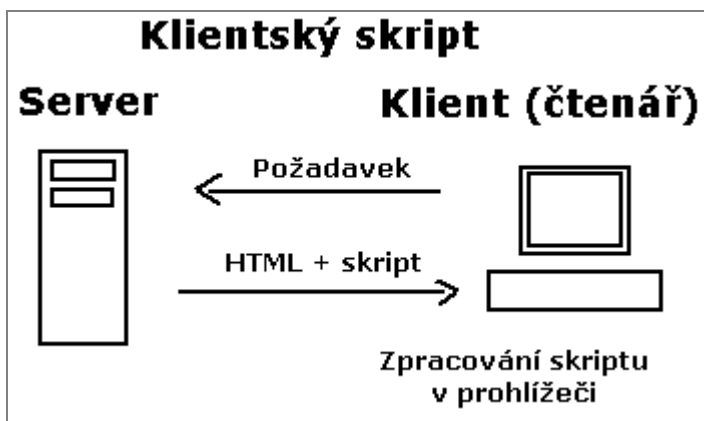
SEO optimalizace se provádí pro účel dobrého hodnocení stránek pro vyhledávače webových stránek. Každý majitel komerčních stránek si přeje, aby vyhledávač na určitá klíčová slova nabídl uživatelům právě jeho stránku jako první. Vyhledávače si proto každou navštívenou webovou stránku zkontrolují a podle vlastních kritérií stránku ohodnotí. Na základě svého hodnocení pak uživatelům na klíčová slova nabízejí odkazy na různé webové stránky v různém pořadí. Stránka, která dosáhne nejlepšího hodnocení, je ideálně optimalizovaná. Mezi kontrolované prvky na webové stránce patří: správná struktura použitých tagů, počet odkazů, klíčová slova, bohatost textu, atd.

Historie HTML

První verze jazyka 0.9 byla vydána v roce 1991. Ta však nepodporovala grafický režim. Mezi několik posledních verzí patří verze 4.0 z roku 1997, která už uměla práci s prvky, jako jsou tabulky, formuláře a rámy. Verze 4.01 pak opravuje chyby předešlé z verze. Poslední 5.0, která se začala vyvíjet v roce 2007 se specifikací HTML 5, by měla být plně podporována v období 2010 až 2012.

1.2 JAVASCRIPT

Javascript je programovací jazyk, který se vykonává na straně klienta a provádí ho přímo internetový prohlížeč (Obr. 1 - Vykonávání skriptu u klienta). Kód, jenž se má vykonat, je obsažen přímo HTML dokumentu. Nejčastěji se používá pro kontrolu formulářů před jeho odesláním na server, pro tvorbu menu webové stránky, vytváření nového okna prohlížeče a jeho nastavení a další.



Obr. 1 - Vykonávání skriptu u klienta¹

Umístění kódu Javascriptu

Kód javascriptu vykonává Internetový prohlížeč, který na něj může narazit třemi způsoby:

1. V HTML dokumentu mezi značky <SCRIPT> kód</SCRIPT>
2. Hodnota v attributech HTML značky (onClick, onMouseMove, ...)
3. Jako pseudoprotokol, který se píše místo URL adresy

¹ viz zdroj: <http://www.jakpsatweb.cz/javascript/javascript-uvod.html>

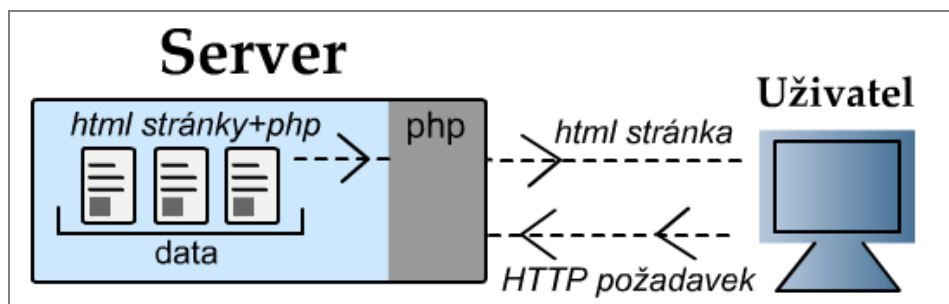
Příklad použití Javascriptu, který kontroluje, zda je vyplněné pole formuláře před jeho odesláním.

```
<html>
<head>
<title>Moje stránka</title>
<script type="text/javascript" language="JavaScript">
<!--
function kontrola()
{
    var jmeno = self.document.forms. formular.jmeno.value;
    var kontrola = jmeno != "false";
    if (kontrola == false) alert('Promiňte, ale jméno musí být
zadáno!');
    return kontrola; }
// -->
</script>
</head>
<body>
<form name="formular" onSubmit="return kontrola();">
Jméno: <input name="jmeno" type="text">
<input type="submit" value="Odeslat">
</form>
</body>
</html>
```

Zdrojový kód 2 – Příklad Javascriptu

1.3 SKRIPTOVACÍ JAZYK PHP

PHP je skriptovací jazyk pro tvorbu webových aplikací, který se vykonává na straně webového serveru např. Apache. Kód, který uložen na webovém serveru a který se volá, je vykonán a výsledkem je většinou vygenerovaný kód HTML dokumentu, ale může to být i soubor PDF, CSV a jiné další typy(Obr. 2 - Vykonávání skriptu na straně serveru).

Obr. 2 - Vykonávání skriptu na straně serveru²

První PHP kód byl vytvořen v roce 1994 a vytvořil jej Rasmus Lerdorf, který jej označoval jako Personal Home Page, odtud PHP. O jeho projekt se pak začalo zajímat mnoho firem a do dnešního dne prošlo PHP mnoho změnami. Aktuální poslední verze nese označení 5.3.6 (vydaná 17. března 2011).

PHP není jediný webový skriptovací jazyk na straně serveru, mezi jeho konkurenty patří například Perl, Java Server Pages, Active Server Pages (ASP), ale i přes tyto konkurenty patří PHP mezi nejrozšířenější pro tvorbu webových aplikací, a to díky mnoha jeho přednostem, jako je např. cena, multiplatforma, jednoduchost pro učení, podpora pro komunikaci s různými databázovými systémy, atd.

PHP je volně šířitelný, takže si může aktuální verzi zdarma stáhnout kdokoli z webu www.php.net, a je možné jej nainstalovat na více operačních systémů jako např. Linux nebo Windows server. Syntaxe PHP je velice podobná známým programovacím jazykům, jako je C, Pascal, JAVA a není těžké pro programátory, kteří již nějaké zkušenosti mají, se php jazyk naučit. PHP skripty se mohou připojovat a komunikovat s databázovým serverem, např. MySQL, Oracle, PostgreSQL či MSSQL a také podporují mnoho knihoven, které mohou sloužit třeba pro komunikaci přes síť, takže je možné odesílat emaily, využít komunikaci přes FTP nebo knihovny pro tvorbu souborů např. ve formátu PDF či CSV.

1.4 DATABÁZOVÝ SYSTÉM MYSQL

MySQL je databázový relační systém vytvořený společností MySQL AB, který vychází z deklarativního jazyka SQL. Zkratka MySQL je přeložena z anglického My Structured

² viz zdroj: <http://programujte.com/?akce=clanek&cl=2005101701-web-ve-flashi-1-dil>

Query Language. Databázový systém se používá pro ukládání, třídění, vyhledávání či editování dat, které se uchovávají v tabulkách, jenž obsahuje sloupce a řádky. Řádek obsahující data z různých sloupců z tabulky se nazývá záznam. Server MySQL zajišťuje správu dat hned pro více uživatelů současně, přitom kontroluje, zda k datům má uživatel oprávněný přístup. Díky těmto vlastnostem se MySQL server stává multi-uživatelským a multi-threadovým systémem. Také díky tomu, že jeho syntaxe je implementována v mnoha programových jazycích (php, C, JAVA, ...) a multiplatformě, tj. že jej lze instalovat jak na serverech Linux, Windows a dalších, a také díky licenci Open Source se stává jedním z nejpoužívanějších databázových relačních systémů.

Pro správu databáze MySQL jako např. vytvoření databáze, tabulek, správa dat či zálohování slouží nejčastěji PhpMyAdmin zpřístupněný přes webové rozhraní napsané v php, taktéž licencované jako Open Source. Aplikace pak přistupují k databázi pomocí dotazů jako SELECT, INSERT, DELETE, ...

Uvedu zde pár příkladů SQL dotazů.

Dotaz pro výběr dat z tabulky Uchazec

```
$dotaz = mysql_query("SELECT * FROM Uchazec WHERE id_obor='2'")
```

Zdrojový kód 3 - SQL dotaz SELECT

Dotaz pro update záznamu v tabulce Predmety hodnotami z formuláře

```
$dotaz = mysql_query("UPDATE Predmety SET
    zkratka= '". $_POST['zkratka'] ."',
    nazev = '". $_POST['nazev'] ."',
    prumerAN = '". $_prumer ."'
    WHERE id = '". $_POST['id'] .'"")
```

Zdrojový kód 4 - SQL dotaz UPDATE

Dotaz pro smazání záznamu z tabulky Predmety

```
$smazat=mysql_query("DELETE FROM Predmety WHERE
id='". $_GET['id'] ."' LIMIT 1")
```

Zdrojový kód 5 - SQL dotaz DELETE

1.5 WEBOVÝ SERVER

Webový server je počítač, který vyřizuje požadavky http (hyper text transfer protokol) od klienta, například internetového prohlížeče, a vyřízené požadavky posílá zpět klientovi. Nejčastěji se jedná o požadavky pro tvorbu webové stránky ve formě HTML. Součástí odpovědi je také stavový kód, který definuje, zda byl požadavek vyřízen v pořádku. Dokumenty, které webový server sestavuje, mohou být dvojího druhu, a to statické či dynamické.

U statických je HTML dokument na serveru kompletní, není třeba jej nějakým skriptem skládat, takže server dokument pouze načte a odešle klientovi.

U dynamických dokumentů musí po přijetí požadavku od klienta webový server HTML stránku nejdříve sestavit, a to například pomocí dotazů v php pro získání potřebných dat z databáze. Tento postup je nejčastější a vyžaduje náročnější softwarové vybavení serveru.

Nejčastější softwarové vybavení webového serveru:

- Operační systém
 - Windows, Linux (zdarma)
- webový správce
 - IIS Microsoft, Apache (zdarma)
- Databázový systém
 - Oracle, MSSQL, MySQL (zdarma)
- Další aplikace
 - PhpMyAdmin, podpora php, ...

Dnes je možné setkat se programovými balíčky, které při jedné instalaci na operačním systému obsahují všechny zbývající výše uvedené části webového serveru. Výhodné je, že tyto aplikace jsou velmi jednoduché na zprovoznění komplexního webového serveru, a některé jsou zdarma. Zmíním se například o EasyPHP nebo Komplex Web Server, který, jak jsem zmínil obsahuje ony všechny základní balíky (Apache, MySQL, PHP a PhpMyAdmina).

2 NÁVRH APLIKACE

V této kapitole je popsána neformální specifikace a zahajovací část tvorby aplikace, zejména částí aplikace, návrhu databáze a diagramu případu užití pro vytvoření představy a principu funkčnosti aplikace. Tato fáze byla spojena s osobní schůzkou na obchodní akademii v Šumperku s osobami, které se přijímacího řízení přímo aktivně účastní. Na základě získaných informací vznikla níže popsaná specifikace.

2.1 ZÁKLADNÍ ÚVAHA APLIKACE

Celé přijímací řízení trvá několik měsíců a doprovází ho posloupnost několika důležitých kroků, které jsou na sobě závislé. Proto je potřeba si nejdříve tuto posloupnost a jejich vazby uvědomit a teprve potom řešit návrh celého systému.

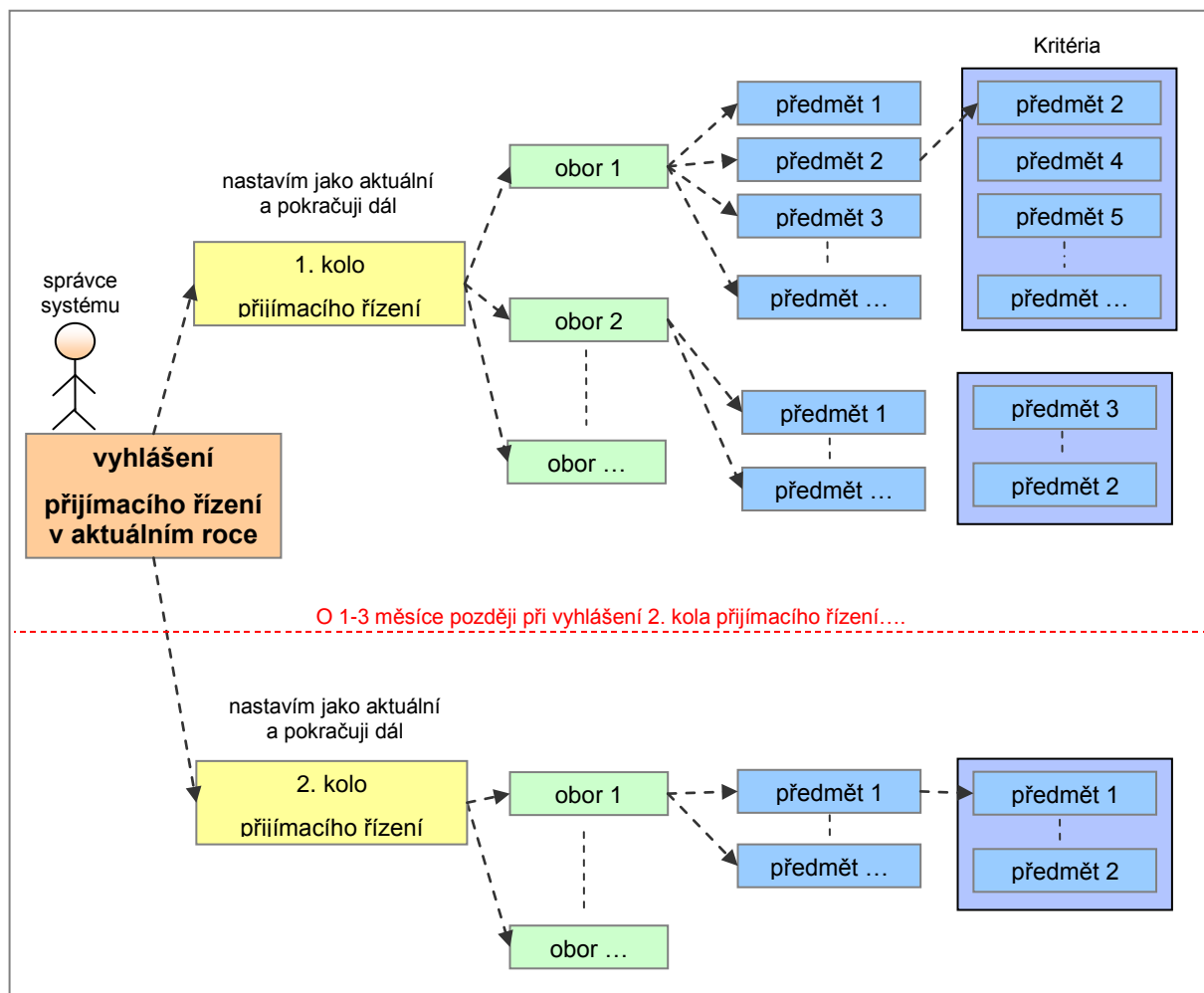
V první fázi to jsou posloupnosti úkonů při vyhlášení přijímacího řízení. Teprve až proběhne vyhlášení přijímacího řízení, je možné zahájit druhou fázi a to přijímat uchazeče, tzn. vkládat do systému jeho záznamy. Po zadání všech záznamů typů uchazeč se může přistoupit k třetí a poslední fázi, která zahrnuje výpočet pořadí těchto záznamů, podle nastavených kritérií. V závěru je možné výsledky tisknout nebo jinak data exportovat.

2.2 POSLOUPNOST ÚKONŮ V PŘIJÍMACÍM ŘÍZENÍ

2.2.1 První fáze – vyhlášení přijímacího řízení

Zahájení přijímacího řízení vyplívá ze školského zákona, který stanoví, že ředitel školy vyhlásí do 31. 1. kritéria přijímacího řízení.

V první části správce systému vyhlásí číslo kola přijímacího řízení a zavede toto kolo jako aktuální. V aktuálním kole přijímacího řízení se musí určit obory, pro které je kolo vypsáno, a u každého oboru se musí určit předměty, ke kterým se bude přiřazovat hodnocení jednotlivých uchazečů. Tato posloupnost je graficky znázorněna na obrázku (Obr. 3 - Posloupnost kroků při vyhlášení přijímacího řízení). Pokud jsou vloženy předměty v daném oboru, pak je potřeba v posledním kroku první fáze zvolit tzv. kritéria. To znamená, že se určí předměty a jejich pořadí podle jeho váhy při hodnocení.



Obr. 3 - Posloupnost kroků při vyhlášení přijímacího řízení

2.2.2 Druhá fáze – vkládání záznamů

V této části se vkládají všechny povinné záznamy uchazeče do systému, tím se shromáždí data před jejím samotným hodnocením.

2.2.3 Třetí fáze – vyhodnocení

V třetí poslední fázi se vyhodnotí záznamy uchazeče a podle kritérií se seřadí. Pak je možné výsledky dále zpracovávat (např. export do souboru pro tisk na nástěnku nebo export do souboru CSV pro další správu v jiných programech).

Práce, která je v průběhu celého přijímacího řízení, je vždy vázána na více osob. Ředitel vyhlásí přijímací řízení a jejich kritéria, administrativní pracovnice přijímají přihlášky, takže vytváří seznam uchazečů s potřebnými údaji, další pověřená osoba vyhodnotí uchazeče podle kritérií, ostatní pověřené osoby vytváří korespondenci atd. Je tedy zřejmé,

že do aplikace bude vstupovat více uživatelů, kteří budou mít odlišná práva a možnosti. Z těchto informací vyplývají na systém určité požadavky, které zde popisují.

2.3 POŽADAVKY APLIKACE

- Víceuživatelský přístup s odlišnými právy
- Přístup do aplikace z počítačové sítě (pokud je k dispozici)
- Zabezpečení dat o uchazeči proti neoprávněnému zneužití
- Tisk seznamů
- Export dat do různých standardních formátů

2.3.1 Víceuživatelský přístup

Pomocí dvou typů uživatelských účtů lze zajistit rozdělení hlavních úkolů aplikace. Uživatel s nejvyšším oprávněním má přístup ke všem funkcím aplikace (správa uživatelů, vyhlášení přijímacího řízení, správa uchazečů, tisk, ...), kdežto uživatel s nižším oprávněním má omezené funkce, jako je správa uchazečů, vyhodnocení uchazečů, tisk výsledků a export dat pro další korespondenci.

2.3.2 Přístup do aplikace z počítačové sítě

Pokud je potřeba přistupovat k aplikaci ze sítě, musí se všechny podpůrné systémy a data umístit na server v lokální síti. Podpůrnými systémy jsou webový a databázový sever, mezi data se řadí aplikace.

2.3.3 Zabezpečení dat o uchazeči

Každé přijímací řízení ať je realizováno na papíře nebo pomocí nějakého softwaru obsahuje citlivé údaje o uchazeči. Tyto informace by neměly přijít do styku s neoprávněnými osobami. Jelikož aplikace, kterou vytvářím jako součást této práce, vyžaduje ke své činnosti webový a databázový server, rozhodně nedoporučuji využívat hostingových služeb, nikdy nevíme, kde data fyzicky leží a kdo k nim má přístup. Ideálním řešením je využití lokální počítačové sítě, kde už nějaký server běží. Možné je i řešení bez počítačové sítě, tzn. na počítači administrativního pracovníka bude nainstalovaná jak aplikace, tak i databázový a webový server. Přístup k databázovému serveru musí být pod heslem, kde toto heslo k databázi zná většinou pouze administrátor a samotná aplikace, která s daty pracuje. Dalším krokem k ochraně citlivých údajů bude vytvořit aplikaci tak,

aby s ní mohla manipulovat pouze oprávněná osoba, tudíž software vyžaduje autorizaci pomocí unikátního jména a hesla.

A screenshot of a web application's login interface. It features two input fields: the first is labeled 'jméno:' and contains the text 'sekretariat'; the second is labeled 'heslo:' and contains four black dots. To the right of these fields is a blue button with the text 'přihlásit'.

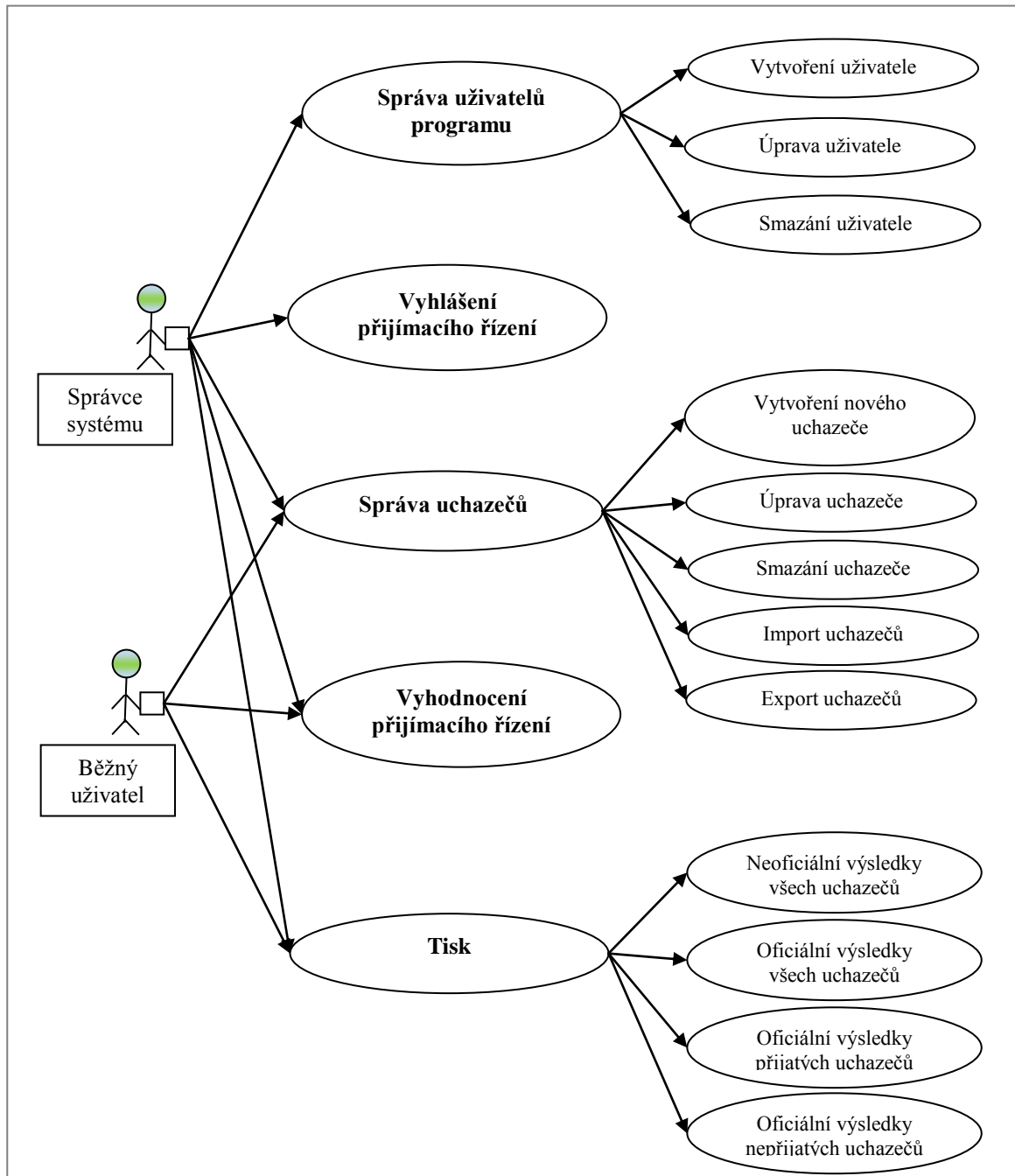
Obr. 4 - Identifikace oprávněné osoby

2.3.4 Tisk seznamů a export dat

Po vyhodnocení přijímacího řízení musí každá škola výsledky umístit na nástěnky v budově školy, na své webové stránky a odeslat vyrozumění o výsledku uchazeče poštou jeho zákonnému zástupci. K účelu tisku výsledků v aplikaci vytvářím přednastavený export do formátu PDF, který lze jednoduše vytisknout nebo vložit na webové stránky školy. Pro korespondenci lze využít funkci pro export ve formátu CSV, se kterým si v hromadné korespondenci poradí každý kancelářský textový editor.

2.4 DIAGRAM PŘÍPADU UŽITÍ

„Diagram použití se používá pro zachycení způsobu, jakým jsou jednotlivé prvky diagramu používány navzájem (pomocí případu použití, aktérů a vztahů mezi nimi).“³



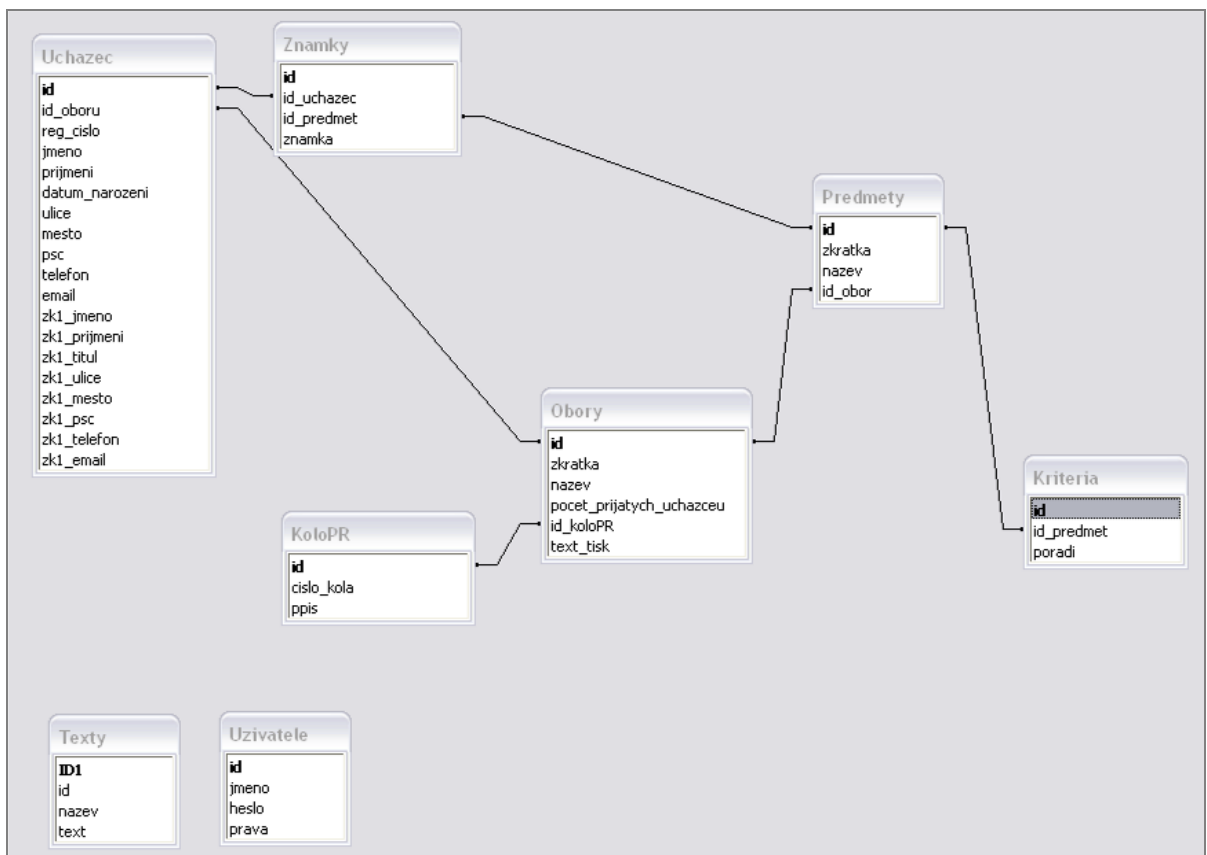
Obr. 5 - Diagram užití

³ <http://www.sgpsys.com/doc/simpleuml/cs-CZ/UML.Foundation.Core.html>

Tento diagram popisuje chování systému z pohledu uživatele. Ukazuje typy uživatelů, které do systému vstupují, a jaké činnosti mohou vykonávat. Umožňuje znázornit funkční požadavky na systém tím, že popisuje interakci mezi ním a uživateli (Obr. 5 - Diagram užití).

2.5 NÁVRH DATABÁZE

Návrh ukazuje strukturu informací, které do databáze vstupují nebo se z ní čtou. Způsobů pro zobrazení takové struktury je více, mezi nejčastější patří návrh databáze pomocí entit, atributů a jejich vztahy. Návrh databáze je vidět obrázku (Obr. 6 - Návrh databáze).



Obr. 6 - Návrh databáze

II. PRAKTICKÁ ČÁST

3 TVORBA APLIKACE

3.1 STRUKTURA APLIKACE

Strukturu této aplikace tvoří v prakticky PHP soubory, které svůj kód vykonávají na webovém serveru. Ten podle potřeby komunikuje s databázovým serverem a nakonec vytvoří soubor s HTML kódem, který se odešle prohlížeči. Ten stránku naformátuje a ve finále ji zobrazí. V PHP souboru se tak dohromady prolínají jazyky HTML, Javascript, SQL a PHP. Kódy HTML plní funkci jak výstupní pro formátování vzhledu dokumentu, tak vstupní pro předávání dat z formuláře. PHP kódy zpracovávají data, tedy jejich funkcí je výpočet a SQL dotazy komunikují vstupně/výstupně s databází. Všechny tyto soubory jsou umístěny a rozděleny dle struktury do složek umístěných na webovém serveru.

Tyto složky obsahují kromě kódu PHP také soubory ve formátech PDF a CSV pro import či export dat z databáze.

3.2 SPRÁVA ZÁZNAMŮ O PŘIJÍMACÍM ŘÍZENÍ

Záznamy o přijímacím řízení se rozdělují na sobě závislé posloupnosti, které je nutné provést ještě před vložením prvního uchazeče, viz kapitola 2.2 Posloupnost úkonů v v přijímacím řízení.

Tato posloupnost je v pořadí:

1. kolo přijímacího řízení,
2. obory v daném kole,
3. předměty v daném oboru
4. a poslední kritéria v oboru z daných předmětů.

Tato posloupnost je znázorněna na obrázku Obr. 3 - Posloupnost kroků při vyhlášení přijímacího řízení.

V této části je potřeba zajistit, aby se při vkládání záznamů opravdu dodržela správná posloupnost a hlavně, aby nebylo možné smazat jen dílčí část záznamu tabulky, protože by byla přerušena relace mezi záznamy v tabulkách.

3.2.1 Kontrola dodržení správné posloupnosti

Ta je při vkládání záznamu zajištěna kontrolním dotazem do tabulky předešlého kroku. Pokud dotaz vrátí nulový počet záznamů, naznačuje to, že uživatel přeskočil některý z kroků a aplikace vypíše varovnou informaci, který z kroků přeskočil.

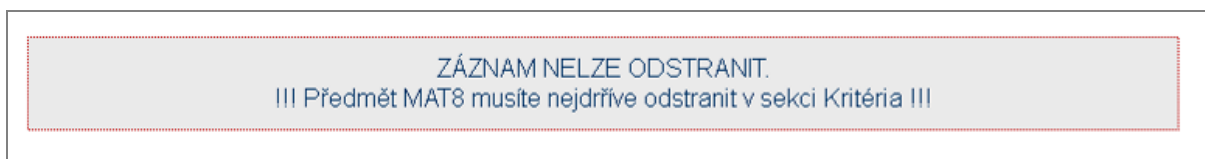
3.2.2 Zabezpečení proti přerušení relace mezi záznamy v tabulkách

Při mazání záznamu v tabulce Kolo PŘ, Obory nebo Předměty se kontroluje dotazem, zda je v tabulce o krok dál nějaký záznam s ID záznamem, který chce uživatel smazat. Pokud dotaz vrátí nenulový počet řádků, smazání se neprovede a vypíše se informace, o tzv. osamocených záznamech.

Příklad kódu, který zabrání přerušení relace mezi tabulkami Předměty a Kritéria:

```
//-----SMAZAT ZAZNAM-----  
  
if ($_GET['i']=="del"){  
  
    $sql_kriterium = mysql_query("SELECT * FROM Kriteria WHERE  
id_predmet= '$_GET['id'].'" LIMIT 1");  
  
    IF(mysql_num_rows($sql_kriterium)>0)  
  
    {  
  
        zprava("ZÁZNAM NELZE ODSTRANIT. <BR />!!! Předmět  
".DejDataZtabulky($_GET['id'], 'Predmety', 'zkratka')." musíte  
nejdříve odstranit v sekci Kritéria !!! ");  
  
    }else  
  
    {  
  
        if($smazat=mysql_query("DELETE FROM Predmety WHERE id=  
'$_GET['id'].'" LIMIT 1"))  
  
        {  
  
            zprava ("Předmět byl smazán");  
  
        }else zprava ("příkaz se neprovedl");  
  
    }  
  
}
```

Výpis odchyčení takové události v aplikaci:



Obr. 7 - Odchyčení chyby

Formulář pro vložení nového předmětu spolu se seznamem předmětů: Správa záznamů o uchazeči

přidej nový předmět

Obor:

zkratka:

název:

Zahrnout známku do průměru

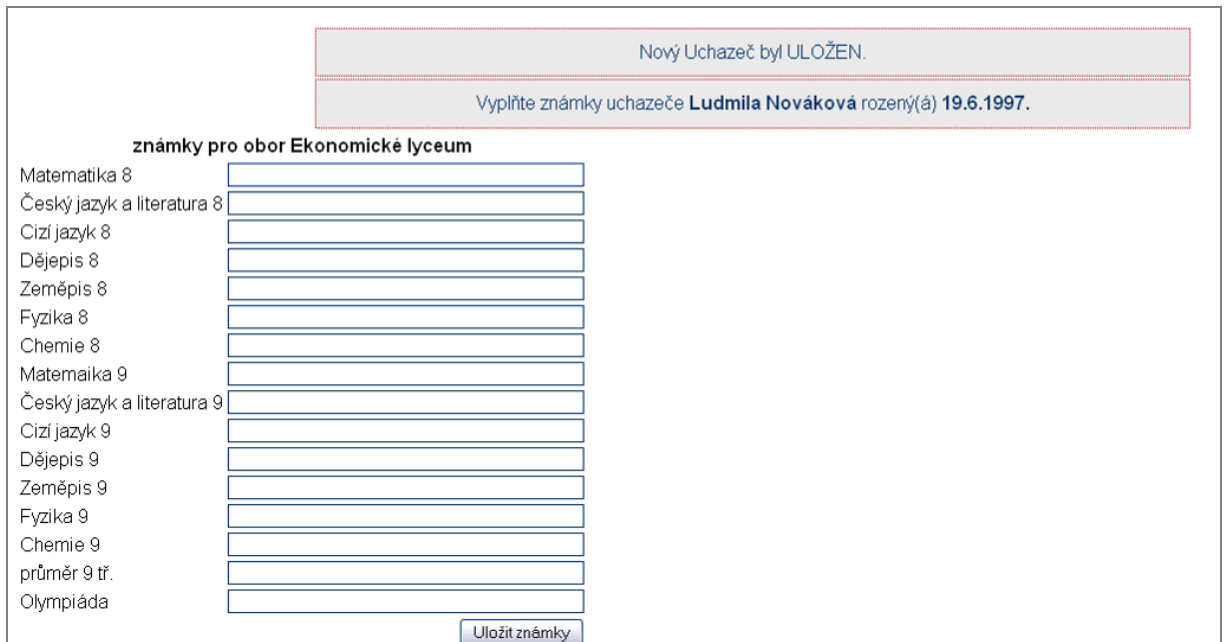
Ekonomické lyceum				
č.	zkratka	název	Zahrnut v průměru	další funkce
1	MAT8	Matematika 8	ano	[upravit] [smazat]
2	CJL8	Český jazyk a literatura 8	ano	[upravit] [smazat]
3	CJ8	Cizí jazyk 8	ano	[upravit] [smazat]
4	DEJ8	Dějepis 8	ano	[upravit] [smazat]
5	ZEM8	Zeměpis 8	ano	[upravit] [smazat]
6	FYZ8	Fyzika 8	ano	[upravit] [smazat]
7	CHE8	Chemie 8	ano	[upravit] [smazat]
8	MAT9	Matematika 9	ano	[upravit] [smazat]
9	CJL9	Český jazyk a literatura 9	ano	[upravit] [smazat]
10	CJ9	Cizí jazyk 9	ano	[upravit] [smazat]
11	DEJ9	Dějepis 9	ano	[upravit] [smazat]
12	ZEM9	Zeměpis 9	ano	[upravit] [smazat]
13	FYZ9	Fyzika 9	ano	[upravit] [smazat]
14	CHE9	Chemie 9	ano	[upravit] [smazat]

Obr. 8 - Správa předmětů

Vkládání záznamu typu uchazeč je v celém řízení časově nejnáročnější částí, protože počet uchazečů se může počítat ve stovkách a u každého uchazeče je potřeba vyplnit mnoho údajů. V této části je poměrně složité vytvoření formuláře pro zadávání hodnocení. Tento formulář musí být plně dynamický, protože názvy předmětů a jejich počet, ze kterých se uchazeč hodnotí, se může každý rok měnit a je to předmětem kritérií v daném kole a oboru. Dynamický HTML formulář pro zadávání hodnocení je vytvořen webovým serverem pomocí SQL dotazu, který zjistí počet předmětů v daném oboru v pořadí, ve kterém se do systému zadávali (podle id). Zápis hodnocení o uchazeči z formuláře do databáze se děje

pomocí SQL dotazu INSERT, který se používá v cyklu podle počtu předmětů v daném oboru.

V případě, že škola má již seznam údajů o některých uchazečů, třeba díky elektronické přihlášce, je možné tyto údaje hromadně do systému importovat. Princip popisují v kapitole 3.5 Import záznamů.



Nový Uchazeč byl ULOŽEN.

Vyplňte známky uchazeče **Ludmila Nováková** rozený(á) 19.6.1997.

známky pro obor Ekonomické lyceum

Matematika 8	<input type="text"/>
Český jazyk a literatura 8	<input type="text"/>
Cizí jazyk 8	<input type="text"/>
Dějepis 8	<input type="text"/>
Zeměpis 8	<input type="text"/>
Fyzika 8	<input type="text"/>
Chemie 8	<input type="text"/>
Matematika 9	<input type="text"/>
Český jazyk a literatura 9	<input type="text"/>
Cizí jazyk 9	<input type="text"/>
Dějepis 9	<input type="text"/>
Zeměpis 9	<input type="text"/>
Fyzika 9	<input type="text"/>
Chemie 9	<input type="text"/>
průměr 9 tř.	<input type="text"/>
Olympiáda	<input type="text"/>

Uložit známky

Obr. 9 - Formulář pro vkládání hodnocení uchazeče

Vkládání hodnocení uchazeče do systému

```

$predmety = mysql_query("SELECT * FROM Predmety WHERE id_obor =
'".$_POST['obor']."' ORDER BY id");

while($predmety_row = mysql_fetch_array($predmety)){

    if ($vloz_znamku = mysql_query("INSERT INTO Znamky (

        id_uchazec,

        id_predmet,

        znamka )

VALUES (

    '".$_POST['id_uchazec'].'',

    '$predmety_row[id]','',

    '".$_POST['predmetID'].$predmety_row[id]'.")

```

```

) " )
}

```

Zdrojový kód 7 - Vkládání hodnocení uchazeče

Samozřejmostí je také výpis uchazečů v každém oboru aktuálního kola. V tomto seznamu je možné uchazeče řadit podle vlastního výběru. Tato možnost řazení je pomocí parametru ORDER BY v dotazu typu SELECT.

Vyberte obor pro zobrazení seznamu uchazečů:

Obor:

řadit podle:

Ekonomické lyceum																						
č.	řad. č.	jméno	příjmení	dat. narození	město, ulice	MAT8	C.JL8	C.J8	DE.J8	ZEM8	FYZ8	CHE8	MAT9	C.JL9	C.J9	DE.J9	ZEM9	FYZ9	CHE9	PRUM9tr	OLY	další funkce
1.	51	Lucie	Andrilová	14.2.1995	Olšany, čp. 3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	[upravit] [smazat]
2.	1	Adéla	Babištová	22.2.1995	Štítý, Na Pilníku 247	2	2	3	1	1	2	1	2	3	3	1	1	2	2	1.957142857	1.53	[upravit] [smazat]
3.	75	Petra	Banková	27.2.1995	Zábřeh, Žerotínov 10	1	2	1	1	1	1	1	2	2	2	1	1	1	2	1.357142857	1.35	[upravit] [smazat]
4.	108	Lucie	Bartlová	2.3.1995	Šumperk, K. H. Máchy 15	2	2	1	1	1	1	1	1	2	1	1	1	1	1	1.214285714	1.12	[upravit] [smazat]
5.	132	Jana	Bartošová	26.4.1995	Šumperk, Šumavská 18	2	2	1	2	2	1	1	1	2	1	2	2	2	2	1.642857143	1.35	[upravit] [smazat]
6.	54	Barbora	Bartošová	17.5.1995	Štítý, Nákladní 236	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1.071428571	1.07	[upravit] [smazat]
7.	39	Michaela	Bilková	7.6.1995	Šumperk, Zahradní 21	3	2	2	1	2	3	2	3	2	2	2	2	2	2	2.142857143	1.59	[upravit] [smazat]
8.	81	Jarmila	Boldilová	16.6.1995	Postřelmov, Novodvorská 149	2	2	2	2	1	2	2	2	3	2	2	1	1	2	1.857142857	1.4	[upravit] [smazat]
9.	49	Pavína	Bröcklová	23.6.1995	Postřelmov, V Kopci 40	4	2	3	3	2	2	2	3	2	2	2	3	2	3	2.5	1.8	[upravit] [smazat]
10.	22	Barbora	Bubeníková	25.6.1995	Jeseník, Zeyerova 656/10	2	2	2	2	2	2	2	2	3	2	2	1	2	3	2.071428571	1.69	[upravit] [smazat]
11.	121	Tomáš	Bundil	29.6.1995	Zábřeh, Na Výsluní 4	2	2	2	2	1	1	1	3	2	2	2	2	2	2	1.857142857	1.56	[upravit] [smazat]
12.	36	Michaela	Čiupná	3.7.1995	Leština, 7. května 97	2	2	1	2	2	2	1	2	2	1	2	1	1	1	1.571428571	1.25	[upravit] [smazat]
13.	15	Michaela	Damiančíková	4.7.1995	Šumperk, Bratrušovská 4	3	3	3	2	3	3	3	3	3	3	2	3	3	3	2.957142857	2.06	[upravit] [smazat]
14.	47	Thi Nhoac	Doan	5.7.1995	Šumperk, J. z Poděbrad	2	2	2	2	2	1	2	2	2	1	1	1	2	2	1.714285714	1.27	[upravit] [smazat]

Obr. 10 - Výpis uchazečů

3.3 VYHODNOCENÍ

Výsledkem vyhodnocení je seřazený seznam uchazečů podle předem zadaných kritérií. Například dle přílohy č. 1 je hlavním kritériem vypočítaný průměr na 5 desetinných míst z určitých předmětů a pokud má tento průměr s některým z dalších uchazečů stejný, pak se hodnotí hodnota zadaných předmětů v daném pořadí.

Při výpočtu vyhodnocení je tedy potřeba získat data z více tabulek, provést výpočet průměru, data dle kritérií seřadit a následně vypsát. Při této operaci se použijí data z tabulek, Kritéria, Předměty, Znamky a Uchazeč, kde si na databázovém serveru vytvořím dočasnou tabulku, do níž jsou uloženy další potřebné údaje. Z této dočasné tabulky lze pak pomocí SQL dotazu SELECT a parametru ORDER BY získat vyhodnocené pořadí záznamů uchazečů se všemi potřebnými údaji viz obrázek (Obr. 11 - Seznam vyhodnocených uchazečů).

Pro archivaci vyhodnocených dat byla vytvořena funkce pro možnost exportu kompletního záznamu každého uchazeče, který se účastnil vyhodnocení, včetně vypočítaných průměrů

a veškerých informací v systému. Všechny tato záznamy jsou seřazeny a číslovány podle pořadí. Více tuto funkci rozvádím v kapitole. 3.6 Export záznamů.

pořadí	jméno	příjmení	dat. narození	průměr	pocet znamkek	DEJ9	CJ9	CJL9	MAT9	CHE8	FYZ8	ZEM8	DEJ8	CJ8	CJL8	MAT8	PRUM9tr	OLY
1.	Lucie	Andrlová	14.2.1995	1.00000	11	1	1	1	1	1	1	1	1	1	1	1	1	1
2.	Pavčina	Kohoutková	19.10.1995	1.00000	11	1	1	1	1	1	1	1	1	1	1	1	1	1
3.	Adéla	Smékalová	15.5.1996	1.00000	11	1	1	1	1	1	1	1	1	1	1	1	1.071428571	1.12
4.	Martina	Pavličiková	1.3.1996	1.09091	11	1	1	1	1	1	1	1	1	1	2	1	1.071428571	1
5.	Lucie	Piknová	1.3.1996	1.18182	11	1	1	1	1	1	1	2	1	1	2	1	1.142857143	1
6.	Veronika	Hrabáčková	11.9.1995	1.18182	11	1	1	1	1	2	1	1	1	1	1	2	1.142857143	1
7.	Nikola	Kobzová	15.10.1995	1.27273	11	1	1	1	1	2	2	1	1	1	1	2	1.357142857	1.13
8.	Martina	Příborská	24.3.1996	1.09091	11	1	1	1	2	1	1	1	1	1	1	1	1.071428571	1.06
9.	Barbora	Bartošová	17.5.1995	1.09091	11	1	1	1	2	1	1	1	1	1	1	1	1.071428571	1.07
10.	Adam	Křenek	21.11.1995	1.18182	11	1	1	1	2	1	1	1	1	1	1	2	1.142857143	1.06
11.	Denisa	Moravcová	4.1.1996	1.18182	11	1	1	1	2	1	1	1	1	1	2	1	1.142857143	1.06
12.	Eliška	Kupková	27.11.1995	1.18182	11	1	1	1	2	1	1	1	1	1	2	1	1.142857143	1.13
13.	Daniel	Skoumal	6.5.1996	1.36364	11	1	1	1	2	1	1	2	2	1	1	2	1.357142857	1.12
14.	Bára	Třetinová	22.6.1996	1.18182	11	1	1	1	2	2	1	1	1	1	1	1	1.142857143	1.07
15.	Martina	Matoušková	19.12.1995	1.45455	11	1	1	1	2	2	2	2	1	1	1	2	1.428571429	1.13
16.	Martina	Příkopová	29.3.1996	1.81818	11	1	1	1	2	2	3	2	2	2	2	2	1.857142857	1.47
27.	Petra	Janků	29.9.1995	1.63636	11	1	1	2	2	2	2	1	1	1	2	3	1.571428571	1.2
28.	Veronika	Linhartová	11.12.1995	1.63636	11	1	1	2	2	2	2	2	2	1	1	2	1.571428571	1.31
29.	Libuše	Rýznarová	7.4.1996	1.63636	11	1	1	2	3	1	2	2	2	1	1	2	1.571428571	1.4
30.	Šárka	Římská	11.4.1996	2.18182	11	1	1	3	3	2	3	2	2	1	3	3	2.142857143	1.5
31.	Barbora	Weingartová	2.8.1996	2.18182	11	1	1	3	3	3	2	2	1	2	3	3	2.214285714	1.71
32.	Aneta	Poláchová	8.3.1996	1.63636	11	1	2	1	2	1	2	2	1	2	2	2	1.5	1.12
33.	Marek	Hudec	21.9.1995	1.63636	11	1	2	2	1	1	2	1	1	2	2	3	1.642857143	1.24
34.	Petra	Banková	27.2.1995	1.36364	11	1	2	2	2	1	1	1	1	1	2	1	1.357142857	1.35
35.	Jan	Kamler	14.10.1995	1.63636	11	1	2	2	2	1	1	2	1	2	2	2	1.571428571	1.33
36.	Marie	Krejčí	27.10.1995	1.63636	11	1	2	2	2	1	1	2	1	2	2	2	1.571428571	1.4
37.	Pavla	Minářová	31.12.1995	1.63636	11	1	2	2	2	2	1	1	1	2	2	2	1.5	1.2
38.	Kristýna	Krňávková	1.11.1995	1.63636	11	1	2	2	3	1	2	1	1	1	2	2	1.5	1.32

Obr. 11 - Seznam vyhodnocených uchazečů

3.4 IMPORT ZÁZNAMŮ

Pokud má uživatel k dispozici databázi uchazečů i se známkami z jiného zdroje, je zde vytvořena funkce pro importování celých záznamů. Jelikož pro každý obor v daném kole přijímacího řízení může být požadováno hodnocení z různých předmětů, musí importovaný soubor mít takovou strukturu, která odpovídá vyhlášení přijímacího řízení v aplikaci a to zejména seznamu předmětů. Jednotlivé záznamy o uchazeči, tak musí obsahovat atributy s daty v přesné posloupnosti dle každého oboru. Aby uživatel mohl importovaný soubor správně nastavit, je tak v aplikaci vytvořená funkce, která zjistí počet a názvy atributů o uchazeči, počet předmětů a jejich názvy v pořadí jejich vytvoření a vygeneruje tak

správnou posloupnost potřebných atributů pro soubor. Část kódu pro import zde předkládám spolu s obrázkem č. 12 (Obr. 12 - Předloha struktury souboru pro import), který ukazuje potřebnou strukturu souboru pro daný import.

```
//-----sloupce z tabulky uchazec-----  
  
    $sqlUchazec = mysql_query ("SHOW COLUMNS FROM Uchazec") or  
die ("SQL dotaz neslo provest");  
  
    $row_uchazec = mysql_fetch_row($sqlUchazec);  
  
    $cnt = 1;  
  
    while ($row_uchazec = mysql_fetch_row($sqlUchazec))  
    {  
        if($row_uchazec[0]=="id_obor") { continue; }  
        if($row_uchazec[0]=="id_rok") { continue; }  
        if ($row_uchazec[0]=="kolo_souteze") { continue; }  
        echo "<tr><td>" . $cnt. " - " . $row_uchazec[0]. "</td></tr>";  
        $cnt++;  
    }  
  
//-----vypis predmetu podle daneho oboru-----  
  
    $sqlPredmety = mysql_query("SELECT * FROM Predmety WHERE  
id_obor='".$IDobor.'");  
  
    echo "<tr><td align=\"center\" style=\"background-color:  
#C0FFF0; color:#black; padding:5px;\"><b>ZNÁMKY V  
POŘADÍ</b></td></tr>";  
  
    while($row_predmet = mysql_fetch_array($sqlPredmety))  
    {  
        echo "<tr><td>" . $cnt. " - " . $row_predmet['zkratka']. "  
</td></tr>";  
        $cnt++;  
    }
```

Zdrojový kód 8 – Část kódu pro import dat do tabulek

}

Pořadí dat v souboru pro obor Ekonomické lyceum	ZNÁMKY V POŘADÍ
1 - cislo_jednaci	20 - MAT8
2 - spis_znacka	21 - CJL8
3 - reg_cislo	22 - CJ8
4 - jmeno	23 - DEJ8
5 - prijmeni	24 - ZEM8
6 - rodne_cislo	25 - FYZ8
7 - ulice	26 - CHE8
8 - mesto	27 - MAT9
9 - PSC	28 - CJL9
10 - telefon	29 - CJ9
11 - email	30 - DEJ9
12 - zk1_jmeno	31 - ZEM9
13 - zk1_prijmeni	32 - FYZ9
14 - zk1_titul	33 - CHE9
15 - zk1_ulice	34 - PRUM9tr
16 - zk1_mesto	35 - OLY
17 - zk1_PSC	
18 - zk1_telefon	
19 - zk1_email	

Obr. 12 - Předloha struktury souboru pro import

3.5 EXPORT ZÁZNAMŮ DO FORMÁTU CSV

Export záznamů z databáze může sloužit pro další zpracování v jiných programech nebo jako archivace výsledků či záznamů o uchazeči. Funkce pro export je v aplikaci umožněna ve dvou variantách a to jako export vložených uchazečů a jejich hodnocení nebo export výsledků vyhodnocení přijímacího řízení. Soubory ve formátu CSV je možné v tabulkových editorech prohlížet nebo dále editovat.

Pro tyto exporty záznamů vytvářím vlastní funkce, které pomocí SQL dotazů, cyklů a podmínek v PHP kódu vyberou z databáze příslušná data v daném pořadí a ty vypisují a oddělují pomocí značek tak, aby struktura CSV souboru byla správná. Protože nejen ve funkci pro export vybírám data z více tabulek, je pro rychlost načítání dat z databáze velice efektivní používání spojování tabulek. V těchto funkcích se také objevují kódy využívající knihovny pro práci se soubory. Uvádím zde příklad získání zkratky a id předmětu z tabulek Předměty a Kritéria.

```
$hodnoceni_prvni_radek = "";
$seznam_id_pred_podle_kr = "";
```

```
$sql_kriteria3 = mysql_query("SELECT zkratka, id_predmet FROM
Predmety, Kriteria WHERE Kriteria.id_obor='".$_GET['obor']."' AND
Kriteria.id_predmet=Predmety.id ORDER BY Kriteria.poradi");

WHILE($kriteria_row3 = mysql_fetch_array($sql_kriteria3))
{
    $hodnoceni_prvni_radek.="; ".$kriteria_row3['zkratka']."";
    $seznam_id_pred_podle_kr.=", ".$kriteria_row3['id_predmet'];
}
}
```

Zdrojový kód 9 - Část kódu pro export záznamů

3.6 EXPORT ZÁZNAMŮ DO FORMÁTU PDF

Pro tisk seznamu vyhodnocených uchazečů nebo vyvěšení na webové stránky je optimální použít export potřebných záznamů do formátu PDF, který zaručí správné zobrazení a na různých softwarových prohlížečích. „*FPDF je PHP třída umožňující generování PDF souboru bez použití PDFLib knihovny. První písmeno "F" v názvu FPDF znamená Free a je tedy možné tuto třídu jakkoliv používat nebo i pozměnit.*“⁴

Tato třída podporuje pro tvorbu PDF souborů mnoho velice užitečných vlastností, mezi něž patří volba měřicí jednotky, vkládání záhlaví, podpora pro importování obrázků, barevné formátování objektů, odkazy, komprese, ...

Základní princip pro tvorbu těchto souborů je takový, že místo programování HTML dokumentu a jejich tabulek nebo jiných rámečků se vkládají texty načtené třeba z databáze do takzvaných buněk (cell), které přesně pozicují.

```
if ($nazev_skoly = mysql_result(mysql_query("SELECT text FROM
texty WHERE nazev = 'nazev_skoly'"), 0))
{
    #echo $nazev_skoly;
    $pdf->Cell(0, 11, $nazev_skoly, '');
}
```

⁴ zdroj: <http://www.fpdf.org/>

```
}  
else  
{  
    #echo " V sekci Další funkce není vyplněno pole nazev školy";  
    $pdf->Cell(0, 12, "V sekci Další funkce není vyplněno pole nazev  
školy", 'B');  
}
```

Zdrojový kód 10 - Vkládání záznamu do buňky PDF souboru

ZÁVĚR

Mým cílem bylo vytvořit program pro střední školy, které přijímají nové studenty bez přijímacích testů, tedy jen na základě výsledků hodnocení z nižšího stupně vzdělávání. To znamená, že program vyhodnotí jednotlivé hodnocení uchazečů, kteří jsou v databázi podle předem zadaných kritérií a vyhodnotí jejich pořadí popř. hodnocení. Cílem programu nebylo jen vyhodnocení výsledků uchazečů, ale i další formality související s přijímacím řízením. Například tisk vyhodnocení na nástěnku či prezentaci na webové stránky nebo export vyhodnocených dat, který se dá využít například pro hromadnou korespondenci nebo pro systém školní matriky.

V teoretické části jsem představil pojmy webový a databázový sever, jenž aplikace potřebuje pro svoji činnost a programovací jazyky (HTML, Javascript, PHP a SQL), ze kterých se kód aplikace skládá. Dále jsem v této části ukázal podrobný postup vyhlášení přijímacího řízení a jeho vyhodnocení pro správnou realizaci programu.

V praktické části této práce se věnuji strukturám souborů typu PHP, ze kterých se aplikace skládá a ve kterých se kombinují již v teorii řečené kódy PHP, SQL a Javascript. Asi největší a nejnáročnější částí při tvorbě programu byly dotazy SQL, kde ve spoustě případů bylo potřeba použití spojování tabulek. Ovšem základním kamenem pro správnou realizaci programu bylo vytvoření návrhu databáze, kterou jsem musel několikrát konzultovat s mým vedoucím bakalářské práce panem ing. Petrem Šilhavým, za což děkuji.

Program jsem se snažil vytvořit tak, aby jeho použití nebylo specifické pouze pro jednu školu.

I když program splňuje všechny cíle kladené v zadání, doporučil bych další vývoj nových funkcí či modulů, a to zejména v úkonech, které se provádí po přijímacím řízení. Může to být například modul pro tvorbu hromadné korespondence s uchazeči.

Závěrem chci říci, že tato práce mi pomohla rozvinout základní znalosti a chápání pro programování skriptů obsahující hlavně SQL jazyk, a myslím si, že tento typ vytvořené aplikace má pro školy v dnešní době veliký význam, a to zejména v odstranění některých početních či jiných chyb, a také z důvodu velké úspory času při vyhodnocování výsledků uchazečů.

ENDING

Goal of this thesis was creation of application for intermediate schools which accept new students without entrance examinations, just by evaluating results from primary schools. The application analyzes individual results of the competitors which are in the database by the requested criterions and evaluates the order of competitors. Goal of the application was not only evaluation of the primary school results but also other formalities related to the admissions. For example print of the admission results, presenting of the results on the school web pages or export of admission data which could be used for sending letters to the competitors or for school register.

In the theoretical part of the thesis I introduced web server and database server which are needed by the application and programming languages (HTML, Javascript, PHP and SQL) that are used in source code of my application. Then I described detailed procedure of admission creation and admission evaluation that is implemented in my application.

In the practical part of the thesis I described structure of PHP files that create skeleton of my application and which combine programming languages mentioned above (HTML, Javascript, PHP and SQL). The biggest and most complicated part of my work was creating of SQL queries because lot of these queries joins several database tables. For proper realization of my work was necessary to design database model. This database model was several times consulted with leader of my thesis Ing. Petr Šilhavý. I'd like to thank a lot for his effort.

I've tried to implement the application not only for one concrete intermediate school but for many of them.

The application fulfils all requested goals but I would recommend extending it by other functions and modules to support also situations after admissions. It could be for example module for sending letters to the competitors.

My work on this application helped me to evolve basic knowledge and understanding of scripts with SQL queries and I also think that this kind of tool could be really useful for intermediate schools because it minimize compute mistakes and also saves time during evaluation of competitors' results.

SEZNAM POUŽITÉ LITERATURY

- [1] M. SCHURMAN, Eric; J. PARDI, William . *Dynamicé HTML v akci*.
Hornocholupická 22, Praha 4 : Computers Press, 2000. 421 s. ISBN 80-72226-401-X.
- [2] RESIG, John. *JavaScript a Ajax : Moderní pogramování webových aplikací*.
Brno : Computer Press a.s., 2007. 360 s. ISBN 978-80-251-1824-5.
- [3] CROFT, Jeff; LLOYD, Lan; RUBIN, Dan. *Mistrovství v CSS : Pokročilé techniky pro webové desingéry a vývojáře*. Holandská 8, Brno : Computer Press, a.s., 2007. 409 s. ISBN 978-80-251-1705-7.
- [4] NARAMORE, Elizabeth, et al. *Vytváříme webové aplikace v PHP, MySQL a Apache*. Nám. 28. dubna 48, Brno : Computer Press, a.s., 2006. 812 s. ISBN 80-251-1073-7.
- [5] WELLING, Luke; THOMPSONOVÁ, Laura. *PHP a MySQL - rozvoj webových aplikací*. Druhé vydání. Pod Křížem 15, Praha 8 : SoftPress s.r.o., 2004. 910 s. ISBN 80-86497-60-7.
- [6] KOSEK, Jiří. *Téměř vše o WWW* [online]. 2009-2010 [cit. 2011-02-03].
Dostupné z WWW: <www.kosek.cz>.
- [7] JANOVSKEÝ, Dušan. *Jak psát web - návod na html stránky* [online]. -, 2011 [cit. 2011-02-03]. Dostupné z WWW: <www.jakpsatweb.cz>. ISBN 1801-0458.

SEZNAM OBRÁZKŮ

Obr. 1 - Vykonávání skriptu u klienta	14
Obr. 2 - Vykonávání skriptu na straně serveru	16
Obr. 3 - Posloupnost kroků při vyhlášení přijímacího řízení.....	20
Obr. 4 - Identifikace oprávněné osoby.....	22
Obr. 5 - Diagram užití.....	23
Obr. 6 - Návrh databáze.....	24
Obr. 8 - Správa předmětů.....	28
Obr. 7 - Odchycení chyby.....	28
Obr. 9 - Formulář pro vkládání hodnocení uchazeče.....	29
Obr. 10 - Výpis uchazečů	30
Obr. 11 - Seznam vyhodnocených uchazečů	31
Obr. 12 - Předloha struktury souboru pro import	33

SEZNAM PŘÍKLADŮ PROGRAMOVÝCH KÓDŮ

Zdrojový kód 1 - Použití kaskádových stylů	13
Zdrojový kód 2 – Příklad Javascriptu	15
Zdrojový kód 3 - SQL dotaz SELECT	17
Zdrojový kód 4 - SQL dotaz UPDATE	17
Zdrojový kód 5 - SQL dotaz DELETE	17
Zdrojový kód 6 - Kontrola při mazání záznamu	27
Zdrojový kód 7 - Vkládání hodnocení uchazeče	30
Zdrojový kód 8 – Část kódu pro import dat do tabulek.....	32
Zdrojový kód 9 - Část kódu pro export záznamů	34
Zdrojový kód 10 - Vkládání záznamu do buňky PDF souboru	35

SEZNAM PŘÍLOH

1. Kritéria přijímacího řízení pro Obchodní akademii v Šumperku
2. Výsledky přijímacího řízení

PŘÍLOHA 1: KRITÉRIA PŘIJÍMACÍHO ŘÍZENÍ PRO OBCHODNÍ AKADEMII V ŠUMPERKU

Do 1. ročníku budou přijati uchazeči, kteří dosáhli nejlepšího celkového průměrného prospěchu počítaného ze známek z vybraných předmětů v 1. pololetí posledního ročníku a ve 2. pololetí předposledního ročníku, ve kterém uchazeč splnil povinnou školní docházku nebo úspěšně ukončil základní vzdělávání před splněním povinné školní docházky. Jedná se o následující předměty: český jazyk a literatura, první cizí jazyk, dějepis, zeměpis, matematika, fyzika, chemie. Vše počítáno na 5 desetinných míst.

V případě dosažení shodného celkového průměru u uchazečů na posledním přijímaném místě rozhodne o pořadí těchto uchazečů následující:

Výsledky dosažené v okresním kole soutěží (olympiád), kterých se žák zúčastnil ve školním roce 2010/2011. Hodnotí se výsledky ze soutěží (olympiád) v profilových předmětech, viz výše. Uchazeč je povinen doložit doklady (diplom) prokazující účast a umístění v daných soutěžích nejpozději do 15. března 2011. V případě, že i po zohlednění tohoto kritéria bude na posledním přijímaném místě více uchazečů, rozhodne o jejich pořadí:

Lepší hodnocení, kterého uchazeč dosáhl na vysvědčení v 1. pololetí posledního ročníku, ve kterém uchazeč splnil povinnou školní docházku nebo úspěšně ukončil základní vzdělávání před splněním povinné školní docházky, z předmětu: 1. Matematika. Při shodě; 2. Český jazyk a literatura. Při shodě; 3. První cizí jazyk. Při shodě; 4. Dějepis. Při shodě; 5. Zeměpis; Při shodě; 6. Fyzika Při shodě; 7. Chemie.

V případě, že i po zohlednění tohoto kritéria bude na posledním přijímaném místě více uchazečů, rozhodne o jejich pořadí: Lepší průměrný prospěch, kterého uchazeč dosáhl na vysvědčení v 1. pololetí posledního ročníku, ve kterém uchazeč splnil povinnou školní docházku nebo úspěšně ukončil základní vzdělávání před splněním povinné školní docházky ze všech předmětů.

PŘÍLOHA 3: VÝSLEDKY PŘIJÍMACÍHO ŘÍZENÍ

Obchodní akademie a Jazyková škola s právem státní jazykové zkoušky, Šumperk, Hlavní třída 31
POŘADÍ UCHAZEČŮ PODLE VÝSLEDKŮ HODNOCENÍ PRVNÍHO KOLA PŘIJÍMACÍHO ŘÍZENÍ ZE DNE 22. DUBNA 2010

Obor vzdělání 78-42-M/02 **EKONOMICKÉ LYCEUM**

Denní forma vzdělávání

Pořadí	Reg. číslo	Celkový průměr	Olympiáda	MAT 8	ČJL 8	CJ 8	DEJ 8	ZEM 8	FYZ 8	CHEM 8	MAT 9	ČJL 9	CJ 9	DEJ 9	ZEM 9	FYZ 9	CHEM 9	Pr. 9. tř.	Výsledek
1	8	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
2	17	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
3	19	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
4	21	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
5	28	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
6	44	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
7	47	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
8	56	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
9	69	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
10	72	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
11	74	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
12	80	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
13	82	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
14	86	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
15	87	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
16	109	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
17	49	1,00000	Ne	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1,07	PŘIJAT
18	13	1,07143	Ne	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
19	22	1,07143	Ne	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
20	25	1,07143	Ne	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
21	50	1,07143	Ne	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
22	78	1,07143	Ne	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1,00	PŘIJAT
23	107	1,07143	Ne	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
24	83	1,07143	Ne	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1,06	PŘIJAT
25	76	1,14286	Ne	1	2	1	1	2	1	1	1	1	1	1	1	1	1	1,00	PŘIJAT
104	5	2,14286	Ne	2	2	2	2	3	3	2	2	2	2	2	2	2	2	1,50	NEPŘIJAT
105	62	2,14286	Ne	1	3	2	2	3	2	2	2	2	3	2	2	2	2	1,73	NEPŘIJAT
106	90	2,21429	Ne	3	2	2	2	2	4	2	3	2	2	2	1	2	2	1,56	NEPŘIJAT
107	93	2,28571	Ne	2	2	3	2	2	2	2	3	2	3	3	2	2	2	1,71	NEPŘIJAT
108	98	2,42857	Ne	3	4	3	3	3	2	2	2	3	2	2	2	1	2	1,56	NEPŘIJAT
109	64	2,64286	Ne	2	3	2	3	3	3	2	3	2	3	3	3	2	3	1,94	NEPŘIJAT

V Šumperku dne 22. dubna 2010

Ing. Tom Ristovský
ředitel školy