



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

VÝUKOVÁ APLIKACE PRO INTERNETOVÉ PROSTŘEDÍ

EDUCATION APPLICATION FOR INTERNET ENVIRONMENT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

BC. PETER KRATOCHVÍL

VEDOUCÍ PRÁCE
SUPERVISOR

DOC. ING. VÁCLAV ZEMAN, PH.D

BRNO 2009

ANOTÁCIA

Diplomová práca sa zaoberá zhodnotením využitia internetových aplikácií vo vyučovacom procese. Takéto aplikácie sú v dnešnej dobe nazývané E-learning.

Teoretická časť sa zaoberá vysvetlením základných pojmov, vysvetľuje možnosti rôznych foriem vzdelávania s ohľadom na možnosti účastnenia sa kurzov, prípravy na vyučovanie a vypracovávanie domácich úloh, vyhodnocuje dôvody používania e-learningu v praxi. Sú tu rozpísané ich výhody a nevýhody ako aj výhody a nevýhody klasickej formy vyučovania. Popisuje najčastejšie prípady využitia e-learningu.

Praktická časť sa zaoberá opísaním vyhotovenia konkrétneho systému podľa dopredu zadaných požiadaviek. Systém bol rozdelený do dvoch hlavných logických celkov, ktorých funkčnosť je popísaná v príslušných častiach. Ďalej sú vymenované všetky použité nástroje ako programovacie jazyky – ich výhody a nevýhody, operačné systémy serverov a podobne. Praktická časť takisto obsahuje informácie o použítom databázovom systéme ako aj podrobný rozpis konkrétnej databázy.

V závere sú zhodnotené teoretické ako aj praktické výsledky práce na projekte.

KLÚČOVÉ SLOVÁ

e-learning, internetová aplikácia, php, javascript, výuka

ABSTRACT

This Master's thesis is dealing with the usage possibility of education application for internet environment in learning process. Nowadays, this kind of applications is called the E-learning system.

Theoretical part is focused on basic principles, describes the reasons why are these systems so popular and used world wide. Negatives and positives of both, the e-learning system as well as the classic form of studying are described. Content of this part also shows the usual usage cases.

Practical part deals with construction of the particular system, based on requested features. The main system was divided into the two main logical parts. Their functionality is well described. In this part also all used programming languages as well as tools and server operation systems are mentioned. Practical part is also used to show information about database system and description of the particular database which was created.

In the conclusion can reader find the summary of all practical and theoretical results achieved during the work on the project.

KEYWORDS

e-learning, internet application, php, javascript, learning

BIBLIOGRAFICKÁ CITÁCIA

KRATOCHVÍL, P. *Výuková aplikace pro internetové prostředí*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 63 s. Vedoucí diplomové práce doc. Ing. Václav Zeman, Ph.D.

PREHLÁSENIE

Prehlasujem, že svoju diplomovú prácu na téma „Výuková aplikace pro internetové prostředí“ som vypracoval samostatne pod vedením vedúceho diplomovej práce s použitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky uvedené v zozname literatúry na konci práce.

Ako autor uvedenej diplomovej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto diplomovej práce som neporušil autorské práce tretích osôb, hlavne som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a som si plne vedomí následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia § 152 trestného zákona č. 140/1961 Sb.

V Brne dňa

.....

podpis autora

POĎAKOVANIE

Ďakujem vedúcemu diplomovej práce doc. Ing. Václavovi Zemanovi, Ph.D a externému vedúcemu Ing. Ondřejovi Pavelkovi, za veľmi užitočnú metodickú pomoc a cenné rady pri spracovaní diplomovej práce.

V Brne dňa

.....
podpis autora

ZOZNAM SKRATIEK

AJAX	Asynchronous JavaScript and XML
ANSI	American National Standards Institute
CD	Compact Disc
DBMS	DataBase Management System
DDL	Data Definition Language
DML	Data Manipulation Language
DOM	Document Object Model
FI	Form Interpreter
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
ID	Identification (number)
IE	Internet Explorer
J2EE	Java 2 Enterprise Edition
LAMP	Linux, Apache, MySQL, PHP/Python/Perl
LMS	Learning Management System
MySQL	My Structured Query Language
ODBC	Open Data Base Connectivity
PHP	Pre Hypertext Processor, Personal Home Pages
SQL	Structured Query Language.
SRBD	Systém Riadenia Bázy Dát
TD	Table Data
TR	Table Row
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USB	Universal Serial Bus
XML	Extensible Markup Language

OBSAH

1.	Úvod a rozbor zadania	9
2.	Úvod do problematiky e-learningových systémov.....	10
2.1.	Pojem E-Learning	10
2.2.	Využitie e-learningu	10
2.3.	Kurzy a ich typy.....	11
2.3.1.	Vyučovanie prezenčnou formou	11
2.3.2.	Dištančná forma vzdelávania	11
2.3.3.	Kombinovaná forma vzdelávania	12
2.4.	Porovnávanie klasických učebníc s elektronickými.....	13
2.5.	On-line, off-line vyučovanie	14
2.6.	LMS	15
2.6.1.	Výhody štúdia s pomocou e-learningu	16
2.6.2.	Účasť na vyučovaní.....	16
2.6.3.	Čas prístupu k e-learningu	17
2.6.4.	Spolupráca študentov.....	17
2.6.5.	Vonkajšie zdroje	18
2.6.6.	Komplexnosť štúdia	18
2.6.7.	Nevýhody e-Learningového prostredia pre štúdium	18
3.	Vlastné riešenie.....	19
3.1.	Popis praktického vyhotovenia vlastného programu	19
3.2.	Popis použitých technológií.....	20
3.2.1.	DBMS (DataBase management system)	20
3.2.2.	SQL.....	21
3.2.3.	Princíp PHP.....	23
3.2.4.	Súvislosť SQL a PHP.....	26
3.2.5.	Princíp JavaScript.....	27
3.2.6.	Princíp AJAX	28
3.2.7.	Ďalšie použité pomocné programy a nástroje.....	29
3.2.8.	Databázová štruktúra	30
3.2.9.	Popis grafického rozhrania aplikácie:	38
3.2.10.	Vlastná funkcionálna aplikácia.....	40
3.3.	Prihlásenie do systému ako učiteľ	43
3.3.1.	Správa kníh	44
3.3.2.	Správa lekcii	46
3.3.3.	Správa užívateľov	51
3.4.	Prihlásenie do systému ako študent	54
3.4.1.	Práca s článkom.....	54
3.4.2.	Pexeso	55
3.4.3.	Preklad viet	56
3.4.4.	Doplňovanie viet	57
3.4.5.	Domáca úloha.....	58
4.	Záver.....	59

1. Úvod a rozbor zadania

Zadaním tejto diplomovej práce bolo vytvorenie výukovej aplikácie pre internetové prostredie. Projekt mal byť realizovaný vytvorením sady skriptov v programovacom jazyku PHP, s použitím databázového systému MySQL ako aj iných nástrojov, ktoré budú spomenuté a bližšie opísané v hlavnej časti tejto práce. Špecifikácia funkcionality je zameraná na vyučovanie jazykov. Aplikácia pozostáva z dvoch hlavných logických častí a to z časti administrátorskej a klientskej. Tieto bloky komunikujú s databázou, čím tvoria v dnešnej dobe už zaužívaný model pre malé a stredne veľké internetové aplikácie. Bolo potrebné vytvoriť internetovú aplikáciu, ktorá bude čo najviac potlačovať ich typické negatívne vlastnosti ako napríklad problém „Click and wait“, čo znamená, že po zvolení nejakej akcie v internetovej aplikácii užívateľ čaká dokedy sa operácia vykoná a následne zobrazí pri opätovnom načítaní celej stránky so všetkými komponentmi. Toto bol jeden z hlavných problémov klasických internetových aplikácií v minulých rokoch. Riešením je použitie skriptovacích jazykov využívaných na strane klienta – internetový prehliadač užívateľa. Najpoužívanejší v dnešnej dobe a takisto zvolený ako vhodný pre túto prácu bol programovací jazyk JavaScript. V nasledujúcom texte bude čitateľ oboznámený s teoretickými znalosťami z oblasti internetových výukových aplikácií potrebnými k porozumeniu opisu praktickej časti tejto práce.

2. Úvod do problematiky e-learningových systémov

2.1. Pojem E-Learning

Podľa [5] by sa dal e-learning vo voľnom preklade definovať ako spôsob vyučovania ktorý využíva prostriedky moderných informačných a komunikačných technológií. Ako príklad by sa dali spomenúť komunikácia učiteľa so žiakom pomocou elektronickej pošty, tzv. real-time komunikácie, videokonferencie, vyučovania pomocou virtuálnych tried ako aj všetky reálne dostupné prostriedky ktoré by na tento typ služieb boli vhodné [6]. Všeobecne by sa dal e-learning definovať ako spôsob vyučovania s využitím moderných informačných a komunikačných technológií, ktorý ma za úlohu zabezpečiť čo najefektívnejšiu a najkvalitnejšiu formu vzdelávania pre čo najširšiu skupinu osôb. E-learning sa považuje za moderný spôsob vyučovania s čo najväčším možným využitím moderných informačných a komunikačných prostriedkov za účelom zefektívniť a skvalitniť vzdelávanie pre čo najširší okruh osôb. Základnou vzdelávacou jednotkou e-learningu je kurz. Je to ucelený samostatný prvok s ľubovoľnou dĺžkou trvania, rôznou formou realizácia, ale s rovnakým cieľom, a to poskytnúť študujúcemu čo najlepšie služby v oblasti vzdelávania tak, aby zvládol určitú problematiku. Najlepšie uplatnenie nachádza e-learning v dištančnom type štúdia. Hlavnou súčasťou e-learningu sú kvalitne spracované výukové materiály v rôznych podobách ako napr. texty, obrázky, grafy, animácie, video sekvencie, demonštračné úlohy, atď. Zabezpečuje taktiež služby pre komunikáciu a spätnú väzbu medzi učiteľom a študentom a to buď v reálnom čase alebo inou formou.

2.2. Využitie e-learningu

E-learningové systémy je možné využiť v mnohých úrovniach vzdelávacieho procesu. V súčasnej dobe sú najviac využívané pri vysokoškolskom a tzv. celoživotnom vzdelávaní ale e-learning si svoje silné postavenie buduje aj napríklad vo firemných vzdelávacích procesoch. Vysokoškolské programy sú obvykle 3 až 6 ročné a skladajú sa z predmetov

ktorých dĺžka je jeden semester, zatiaľ čo programy celoživotného vzdelávania sú obvykle kratšie a skladajú sa z kurzov rôznej dĺžky.

2.3. Kurzy a ich typy

Kurz môžeme definovať ako ucelenú, samostatnú vzdelávaciu jednotku vzdelávacieho programu, s rôznou dĺžkou trvania, formou realizácie, ale s rovnakým cieľom - poskytnúť študujúcemu maximálne využitie tak, aby zvládol určitú problematiku, ktorú sa má naučiť a to ako aj čo sa týka teoretických znalostí, tak i praktických skúseností. Rozoznávame tri základné formy realizácie kurzov:

- Prezenčná
- Dištančná
- Kombinovaná

2.3.1. Vyučovanie prezenčnou formou

Prebieha za prítomnosti študenta a vyučujúceho v rovnakom čase. Táto forma je najpoužívanejšia pretože je vo väčšine prípadov najvhodnejšia. Pri prezenčnej forme vyučovania sa odporúča používať prostriedky informačných a komunikačných technológií ako doplnok klasickej formy napríklad používaním multimedialných učebníc, výukových programov, internetových zdrojov a pod.

Pri tejto forme vyučovania je dôležité, že vyučujúci je v osobnom kontakte so študentom, má možnosť priamo riadiť vyučovanie a získava priamu spätnú väzbu od študentov.

2.3.2. Dištančná forma vzdelávania

Je taká forma, pri ktorej sú vyučujúci a študent v priebehu vyučovania oddelení. Medzi jej hlavné prínosy patrí možnosť pre študenta nebyť

obmedzovaný časovými a vzdialenostnými problémami a takisto dostupnosť vzdelania pre ľudí s telesným hendikepom, ktorí by neboli schopní za vyučováním cestovať. Dištančné štúdium je teda vhodné najmä pre ľudí, ktorí by si prezenčné štúdium nemohli dovoliť či už z dôvodov zdravotných, sociálnych, alebo finančných a sú schopní absenciu intenzity vyučovania doplniť samo štúdiom.

Dištančná forma vzdelávania nadobúda na dôležitosti s rýchlou rozvojom informačných a komunikačných technológií. Z týchto je to hlavne internet, kde sa otvára nespočetné množstvo foriem komunikácie, ako napr.: elektronická pošta, video konferencie, instant messaging, atď. ktoré sa pre potreby vyučovania veľmi hodia. Takisto je vhodné použitie prostriedkov pre tvorbu a publikáciu študijných materiálov. Pri tomto type nie je použitie e-learningu nevyhnutné, ale odporúčané.

2.3.3. Kombinovaná forma vzdelávania

Pri vzdelávaní je veľmi dôležitý kvalitný výukový materiál. Do tohto pojmu patrí ako klasická učebnica tak aj ďalšie materiály pre vysvetlenie problematiky. Obsah výukového materiálu je rôzny. Môže obsahovať text, video sekvencie, zvukové stopy, grafiku, hry, atď. Tento typ materiálu je obvykle študentom dodávaný v elektronickej forme.

Zvláštny typ študijného materiálu je tzv. multimedialny materiál, ktorého súčasťou môže byť LIVE materiál, ktorý je priebežne vytváraný a aktualizovaný vyučujúcim ako aj študentom (študent vypracováva domáce úlohy a pod.). Akákoľvek aplikácia uložená v prostredí systému vyučovania musí byť spustiteľná webovým prehliadačom na počítačoch jednotlivých študentov. Aplikácia môže byť výukový program, príklad vyriešeného projektu alebo úlohy, atď.

Veľmi vhodná je kombinácia štandardnej papierovej učebnice s elektronickými výukovými materiálmi. Tento spôsob umožňuje kombinovať

výhody oboch foriem tvorby a distribúcie výukových materiálov a potláčať ich nevýhody.

2.4. Porovnávanie klasických učebníc s elektronickými

Pozitíva klasických učebníc [6]

Text vytlačený na papieri je pre štúdium zo zdravotného hľadiska najvhodnejší

- Učebnica je prenášateľná
- Do vytlačeného textu je možné si zapisovať poznámky, podčiarkovať a podobne.
- Pre štúdium z učebnice väčšinou študent nepotrebuje žiadne ďalšie nástroje.

Negatíva klasických učebníc

- Nové vydania, nákladné na zaobstarávanie.
- Nákladná distribúcia.
- Materiál v knihe je obmedzený na text, ilustrácie a grafy.
- Problematické, ba až nemožné aktualizovanie študijného materiálu, napríklad internetové odkazy
- Nie je možný interaktívny prístup.
- Nulová možnosť spätnej väzby.

Pozitíva elektronických výukových materiálov

- Môžu obsahovať prvky neuskutočiteľné v papierovom prevedení (aplikácie, výukové programy, interaktívne časti,...)
- Bezproblémová distribúcia bez veľkých nákladov
- Jednoduchá aktualizácia.

- Možnosť zdieľania informácií.
- Možnosť interakcie a spätnej väzby
- Spolupráca študentov a vyučujúceho.

Negatíva elektronických výukových materiálov

- Potreba technického vybavenia: PC so základnou multimediálnou výbavou
- Pri použití LIVE materiálu je potrebné internetové pripojenie
- Študent musí byť oboznámený s technickou stránkou používania systému
- Nepohodlné, možno až nezdravé čítanie textu z obrazovky monitoru

2.5. On-line, off-line vyučovanie

Ako už bolo spomenuté, základným prvkom e-learningových systémov je výukový materiál. Podľa tvrdenia v [8], e-learning obsahuje aj mnoho funkcií, ako napríklad riadenie vyučovania, hodnotenie študentov a podporu komunikácie vďaka ktorým sa stáva veľmi silným nástrojom. Väčšina používaných výukových materiálov sa radí do kategórie off-line, sú to materiály sprostredkované na pamäťových médiách ako CD nosiče, USB flash disky alebo dáta uložené na serveroch. Študent postupne (svojim tempom, alebo podľa učiteľom stanoveného rozvrhu) prechádza študijným materiálom, vykonáva rôzne cvičenia, vypracováva domáce úlohy. Výsledky tohto konania ukladá – posiela vhodnou formou vyučujúcemu, (pomocou e-mailu, alebo pomocou rozhrania e-learningového systému). Počas tejto doby sa môže vyučujúci venovať svojim povinnostiam a týmto dňami sa nezaoberať. K tejto činnosti učiteľ pristupuje v čase vyhradenom pre správu študijného materiálu, opravovanie domácich úloh a komunikáciu so študentmi, kedy sa vyučujúci zaoberá správnym priebehom e-learningového vyučovania,

zabezpečovaním aktuálnych materiálov a prípadným riešením problémov so študentmi. Dá sa povedať, že väčšina vyučovania prebieha v off-line režime.

Existujú ale situácie kedy je vhodné do vyučovacieho procesu zaradiť tzv. on-line prvky ako napríklad kontakt vyučujúceho a študenta v reálnom čase (instant messaging, video chat,...), zdieľanie študijných materiálov. Ideálna je kombinácia off-line a on-line materiálov v takom pomere aby sa čo najviac prispôbila požiadavkám vyučujúceho a študentom v snahe zaistiť plynulý priebeh e-learningového kurzu.

2.6. LMS

Bezproblémovú funkčnosť všetkých doteraz zmienených funkcií e-learningu má na starosti tzv. Learning Management Systém (LMS)[9]. LMS je súbor aplikácií, ktoré zabezpečujú administrátorské úlohy, ktoré je potrebné vykonávať a to od vytvárania a správy výukových materiálov, cez ich správu a aktualizáciu, komunikáciu so študentmi až po hodnotenie ich domácich úloh a ich spravovanie. Pre študenta znamená LMS virtuálny študijný priestor, nájde tu aktuality z vyučovania, nadchádzajúce domáce úlohy, môže sa tu zúčastniť on-line vyučovania, diskutovať s učiteľom / spolužiakmi. Praktická časť tejto diplomovej práce je samostatný LMS, ktorý bol prispôbený na mieru zadaným požiadavkám.

Existujú rôzne voľne dostupné riešenia LMS systémov, napr. systém Atutor, Docebo, Moodle, ktorý používa VUT v Brne a mnoho ďalších. Tieto riešenia som nepoužil predovšetkým pretože nevyhovovali konkrétnym prvkom zadania ako napríklad skutočnosti, že aplikácia má byť realizovaná v značnej časti pomocou Javascript a AJAX technológií, vďaka ktorým sa má správať viac-menej ako aplikácia spustená na lokálnom počítači, tzn. má eliminovať nežiaduce vlastnosti internetových aplikácií, ako neustála potreba znovu načítania celej stránky po každej operácii a s tým spojené nepohodlné dlhé časové medzery v práci s aplikáciou. Ďalším dôvodom bola nedostupnosť konkrétnej funkcionality študentskej časti aplikácie, napríklad zvyrazňovanie

cudzích slov v texte aj s prekladom, pexeso hra na cvičenie so slovičkami, alebo dopĺňanie správnych slov do hotových viet.

2.6.1. Výhody štúdia s pomocou e-learningu

S veľkými výhodami ktoré e-learning ponúka prichádzajú pre užívateľa aj určité komplikácie. [7] Tvrdí, že tou základnou je podmienka, že študent má prístup k počítaču pripojeného na internet. Takisto je dôležité aby študent ukázal určitú pružnosť v prístupí na úplne nový systém štúdia, tento prechod nemusí byť pre každého ľahký. Takisto ako pri prezenčnej forme štúdia, aj pri e-learningu je dôležité aby mal študent určitú disciplínu a pravidelne a zodpovedne si plnil svoje povinnosti, komunikoval s okolím (učiteľ, ostatní študenti). Je potrebná znalosť informačných technológií aspoň na minimálnej – užívateľskej úrovni tak, aby študent bol schopný pracovať s jednoduchým grafickým užívateľským rozhraním GUI.

- Výhody kvalitne pripraveného vyučovania v e-learningu
- Flexibilita v čase a mieste prístupu
- Vlastné zvolené tempo štúdia.
- Možnosť využitia spolupráce kolegov – študentov na omnoho vyššej úrovni ako pri klasickom vyučovaní.
- Výhody samo štúdia.

2.6.2. Účasť na vyučovaní

Jedna z najväčších výhod e-learningových systémov je podľa [6] možnosť účasti na kurzoch z akéhokoľvek miesta na svete, jedinou podmienkou je prítomnosť počítača a pripojenie na internet. V minulosti sa mohol vyskytovať problém s rýchlosťou pripojenia pri úlohách ktoré boli náročné na dátový tok (video, zvukové súbory,...) V dnešnej dobe tento

problém odpadá keďže internetové pripojenie je vo veľkej väčšine prípadov dostatočne rýchle – minimálne ISDN (128 Kbps) .

2.6.3. Čas prístupu k e-learningu

Výukové materiály e-learningu bývajú uložené na výukovom serveri, ktorý je prístupný prakticky nepretržite. Študenti môžu k týmto materiálom pristupovať kedykoľvek. Existuje tzv. Asynchrónna diskusia, pri ktorej je študent schopný sledovať komentáre svojich kolegov a reagovať na ne. V tomto prípade budú odpovede oveľa premyslenejšie, ako v prípade osobného rozhovoru – kedy musí účastník rozhovoru operatívne odpovedať na otázky bez väčšieho priestoru na rozmyslenie. Študenti tým pádom môžu pokračovať v kurzoch každý podľa svojho vlastného tempa. Toto zabezpečuje študentom individuálny prístup, vďaka ktorému sa stávajú e-learningové systémy čím ďalej populárnejšie. Študent má určitý pevne daný časový harmonogram ako dĺžku štúdia jednotlivých kurzov, ale konkrétne časové nároky naňho nie sú kladené. Problémy sa môžu vyskytnúť ak študent prestane dodržiavať časový harmonogram. Vtedy sa do riešenia musí zapojiť aj učiteľ a individuálne riešiť konkrétne problémy so študentom.

2.6.4. Spolupráca študentov

E-learningové systémy umožňujú okrem iného množstva funkcií aj komunikáciu nielen medzi študentom a učiteľom ale aj medzi jednotlivými študentmi navzájom. Vzniká tak určitá spolupráca kedy môžu študenti pracovať na zadaných úlohách napríklad v skupinách. Tento druh spolupráce je pozitívnym prínosom na skvalitnenie vyučovania.

2.6.5. Vonkajšie zdroje

Niekedy je užitočné k študijným materiálom pridávať doplňujúce informácie napríklad vo forme externých odkazov na odborné články a ďalšie veci súvisiace s náplňou kurzu tak, aby bol študentom umožnený hlbší rozbor obsahu. Do výukového programu dobré zapojiť napríklad aj odborníkov – osoby z praxe, ktorých prítomnosť veľmi pozitívne prispieva na celkový efekt vyučovania. Je treba dobre zvážiť výber externých zdrojov použitých v texte, aby obsahovali to čo je pre študentov podľa názoru učiteľa výhodné vedieť a na druhú stranu ich zbytočne nezahltiť informáciami, ktoré pre nich nie sú dôležité.

2.6.6. Komplexnosť štúdia

Jednotlivcovi musia E-learningové systémy poskytovať pedagogickú aj technickú podporu typické pre samostatné štúdium. Toto zahŕňa špeciálne vypracované výukové materiály, ktoré musia byť napísané účelne a zrozumiteľne, multimedialne materiály a simulácie. Informačné materiály by vždy mali obsahovať základné informácie o kurze, kontakty na pedagógov, ako aj na technickú podporu systému a návod ako zaobchádzať so študijným materiálom.

2.6.7. Nevýhody e-Learningového prostredia pre štúdium

V [5] sú definované obmedzenia v prístupe k informačným technológiám a technické problémy. Ani v dnešnej dobe nie je pripojenie každej domácnosti k internetu samozrejmosťou. Dôvodom môže byť cena za internetové pripojenie, nedostupnosť prípojky (zriedkavo). Kvôli tomuto dôvodu nie je e-learning dostupný pre všetkých ľudí rovnako. Takisto z technického hľadiska sa môžu vyskytnúť rôzne komplikácie pri prevádzke systému. Z tohto dôvodu je potrebné mať zabezpečenú užívateľsky dostupnú

technickú podporu, kam sa môže študent v prípade akýchkoľvek problémov obrátiť a nájsť pomoc [7].

3. Vlastné riešenie

3.1. Popis praktického vyhotovenia vlastného programu

Aplikáciu je možné rozdeliť do dvoch logických častí a to na aplikáciu klienta a databázový server, z hľadiska práv sa klientska aplikácia delí na časť Učiteľ a časť Študent, databázový systém bol vytvorený z tabuliek ktorých vzťahy boli definované na základe zadania pre zaistenie požadovanej funkčnosti systému. Programy boli napísané v skriptovacom jazyku PHP[2], s pomocou podpory v jazyku Javascript[13] a technológie Ajax[4], databáza naprogramovaná v jazyku MySQL[3]. Vývoj bol realizovaný pod operačným systémom Ubuntu (Debian Linux), kvôli jeho stabilnosti a predošlým skúsenostiam. Linuxový server bol spúšťaný ako virtuálny stroj cez program Vmware, v prostredí Windows bol spustený Xming daemon, ktorý slúži na tzv. preposielanie dát X-serveru z linuxového prostredia do Windows tak, že je možné spúšťať programy s grafickým rozhraním v Linuxe a pracovať s nimi vo Windows. Ako vývojové prostredie bol zvolený textový editor Geany, o ktorom je možné nájsť podrobnejšie informácie na [10], podporujúcom zvýrazňovanie kľúčových slov používaných vyššie zmienenými jazykmi. Tento editor tiež podporuje pokročilé funkcie ako napr. spustenie linuxového terminálu v samostatnom paneli priamo v hlavnom okne programu, čo je veľmi užitočné najmä pri práci s databázou z príkazového riadku. Na počítači bol nainštalovaný kompletný tzv. LAMP systém (Linux, APACHE, MySQL, PHP), ktorý je odporúčaný pre vývoj malých a stredne veľkých webových aplikácií používajúcich databázové systémy z dôvodu relatívne vysokej spoľahlivosti a voľnej dostupnosti. Systém bol od začiatku testovaný za behu ako na lokálnom, tak aj na webhostingovom serveri a virtuálnom prenajatom serveri z dôvodu zabezpečenia čo najväčšej nožnej dosiahnutej spoľahlivosti. Aplikácia bola testovaná na rôznych internetových prehliadačoch (Mozilla Firefox, Opera, IE),

v operačných systémoch Linux a Windows. Optimalizovaná je ale pre prehliadač Firefox, z dôvodu značného množstva odlišností a problémov v internetovom prehliadači Internet Explorer verzií nižších ako 7.0 najmä použitím technológie AJAX, ktoré by potrebovali veľké množstvo rozdielnych prístupov k programovaniu v celom systéme.

3.2. Popis použitých technológií

3.2.1. DBMS (DataBase management system)

Podľa [6] si je možné databázu predstaviť ako priestor, do ktorého sa ukladajú všetky potrebné údaje. Spracovaním a prístupom údajov v databázach býva poverený program, ktorý sa nazýva DBMS (DataBase Management System), - SRBD (Systém Riadenia Bázy Dát). Celé je to zariadené tak, že všetko spracovanie, ukladanie a získavanie dát je vykonávané DBMS programom. Akékoľvek aplikácie, ktoré databázu používajú vždy pristupuje cez tento DBMS program. Medzi DBMS patrí mnoho programov z najrôznejších cenových kategórií. Väčšina profesionálnych a komerčných DBMS nie sú lacné a ich cena sa pohybuje vo veľmi vysokých čiastkach. Našťastie existujú v kategórii DBMS aj programy, ktoré sú v nízkej cenovej relácii, alebo celkom zadarmo. Predstaviteľom týchto DBMS systémov sú napríklad MySQL[1], PostgreSQL, alebo Firebird. Porovnaním týchto systémov z hľadiska rýchlosti vedie databázový systém MySQL, na druhej strane v možnostiach nastavenia je prepracovanejšia PostgreSQL. Väčšina dnešných DBMS stavia na tzv. relačnom modeli dát. V tomto modeli sú všetky dáta usporiadané do databázových entít. Každá tabuľka spravidla usporadúva údaje o určitom objekte. Databázová entita je usporiadaná do riadkov a stĺpcov. V práci som použil databázový systém MySql z nasledujúcich dôvodov. MySQL ktorý patrí k jednoduchým databázovým systémom, ktorý vďaka orezaniu pre bežné aplikácie nedôležitých funkcií poskytuje zvýšenie rýchlosti bežných a najčastejšie používaných operácií. Jednoducho zhrnuté, MySQL je malý, rýchly, jednoduchý a nenáročný databázový systém. MySQL je projektované pre jednoduché databázy, aj keď s veľkým množstvom

údajov. Každopádne MySQL sa presadilo predovšetkým do oblasti malých webových databáz.

3.2.2. SQL

Už bolo spomenuté, že prístup k údajom uloženým v databáze obstaráva DBMS program. Aby mohli iné aplikácie používať tieto údaje z databázového servera, musí DBMS program ponúkať rozhranie, pomocou ktorého budú môcť ostatné aplikácie s DBMS spolupracovať. Pri práci s DBMS programom sa používa osvedčený model klient/server. V role serveru vystupuje DBMS program ponúkajúci svoje služby. DBMS program je tak nepretržite spustený a čaká na požiadavky od klientov - iných aplikácií. Na tieto požiadavky potom odpovedá a reaguje na ne. Pretože DBMS program vystupuje v role servera, nazýva sa tiež databázový server. Pre zápis požiadaviek na databázový server sa dnes najčastejšie používa SQL jazyk. SQL je anglická skratka znamenajúca Structured Query Language. Tento jazyk dnes používa drvivá väčšina databázových serverov. Preto sa také servery nazývajú zjednodušene SQL servery. Samé SQL obsahuje všetko potrebné pre prácu s databázami – ako už príkazy pre prácu na vytvorenie, rušenie, modifikovanie tabuľky, tak aj pre prácu so samotnými údajmi - pridávanie, zmenu, rušenie a vyhľadávanie údajov. Jazyk SQL má za sebou pomerne dlhý vývoj. Jeho prototyp bol prvýkrát implementovaný v roku 1974 spoločnosťou IBM. Od tej doby má za sebou SQL dynamický vývoj, a dá sa povedať, že jazyk SQL sa dodnes stále vyvíja, hlavne čo sa týka pridávania nových technológií. Skutočným faktickým štandardom sa stal SQL až v roku 1986, kedy sa dočkal štandardizácie vďaka organizácii ANSI. Od tej doby sa SQL stal univerzálnym nástrojom pre programovanie databáz. Každý, kto sa dnes dostáva do styku s vývojom, alebo ladením databáz, sa musí zoznámiť s jazykom SQL, ktorý je u databáz prakticky všadeprítomný. Pomocou jazyka SQL teda vyjadrujeme, čo chceme v databáze vykonať. Jazyk SQL sa skladá z dvoch základných častí a to DDL, a z časti DML. Príkazy DDL sú skratkou Data Definition Language, a slúžia na definíciu a modifikáciu tabuliek a ďalších

databázových objektov. Príkazy DML sú zas skratkou Data Manipulation Language a slúžia pre manipuláciu s databázovými údajmi. Štruktúra SQL príkazu je znázornená na obrázku Obr. 1

Základná terminológia

Podľa [3] SQL pracuje s tromi objektmi: **tabuľka** (table), **riadok** (row), **stĺpec** (column)



Obr.1 Štruktúra SQL príkazu [3]

kľúčové sloveso: označuje činnosť, ktorú daný príkaz vykonáva (napr. SELECT)

klauzula: bližšie špecifikuje činnosť alebo dáta, s ktorými príkaz pracuje (napr. DISTINCT)

kľúčové slovo: slovo, ktorým klauzula začína (napr. FROM)

meno objektu: meno tabuľky alebo stĺpca

kvalifikátor: jednoznačná špecifikácia objektu (spravidla menom tabuľky – napr. KNIHY.Autor)

výraz: hodnotu výrazu musí SQL spočítať, potom ju umiestni do tabuľky výsledkov

operand: časť výrazu obsahujúca dáta, s ktorými sa má pracovať

- konštanta: jej hodnota sa objaví v každom riadku tabuľky výsledkov
- premenná: symbol, ktorý môže nadobúdať rôznych hodnôt

mená stĺpcov (polí, položiek, atribútov) – *meno tabuľky.meno stĺpca*

- vstavané funkcie (agregáty)

operátor: je uvedený symbol označujúci operácie, ktoré je nutné vykonať s danými operandami

- aritmetické
- logické
- relačné

3.2.3. Princíp PHP

V tejto kapitole bude priblížený systém, akým PHP pracuje a v skratke aj históriou jeho vývoja

Základná charakteristika

V dobe internetových začiatkov boli všetky internetové stránky statické. To znamená, že tak, ako bola stránka napísaná, tak bola aj odoslaná do prehliadača a tak bola tiež zobrazená. To pochopiteľne časom prestávalo stačiť, a preto bola vyvinutá celá rada technológií, ktoré mali stránky rozpohybovať. Jednoducho povedané, sa dajú tieto technológie rozdeliť do dvoch skupín, na "klientske" a "serverové". Podľa [14] "Klientske" technológie spoliehajú na skutočnosť, že spolu s HTML stránkou je prehliadaču odoslaná aj časť programového kódu a ten je vo vhodnej chvíli na "cieľovom" počítači spustený. Vhodná chvíľa môže nastať napríklad pri kliknutí na tlačidlo, pri ukázaní myšou na odkaz, pri otvorení okna prehliadača a podobne. O spustenie klientskeho kódu sa stará prehliadač - a to môže byť nevýhoda. Prehliadač totižto musí poznať programovací jazyk, v ktorom je kód napísaný. Príkladom technológií bežiacich na strane klienta je podľa [3] napríklad Java

script. Na odlišnom princípe sú založené "Serverové" technológie. Keď prehliadač požaduje webovú stránku zo servera, server túto stránku najprv zostaví a potom odošle. Servery môžu zostavovať zakaždým inú stránku v závislosti na tom, čo presne prehliadač požaduje.

PHP je technológia spustená na serveri. Typický PHP skript obsahuje ako časti normálneho HTML kódu, tak aj časti programového kódu. Keď webový server obdrží požiadavku na spracovanie takého skriptu, vykoná nasledujúce tri kroky:

- Časti HTML bez akýchkoľvek úprav
- Časti PHP programového kódu vykoná
- výsledok skombinuje a odošle prehliadaču

Táto filozofia fungovania je nesmierne mocná. Server totiž môže vykonať jednu alebo dokonca niekoľko operácií a výsledok poslať do prehliadača ako obyčajnú HTML stránku. Prehliadač nemá žiadne informácie o tom, čo všetko sa na serveri dialo, vidí len samotný výsledok. Je potrebné si uvedomiť, že na serveri sa mohlo diať veľké množstvo vecí – matematické výpočty, prístupy k databázam, formátovanie, operácie s reťazcami a podobne. Z toho vyplýva niekoľko dôležitých vecí:

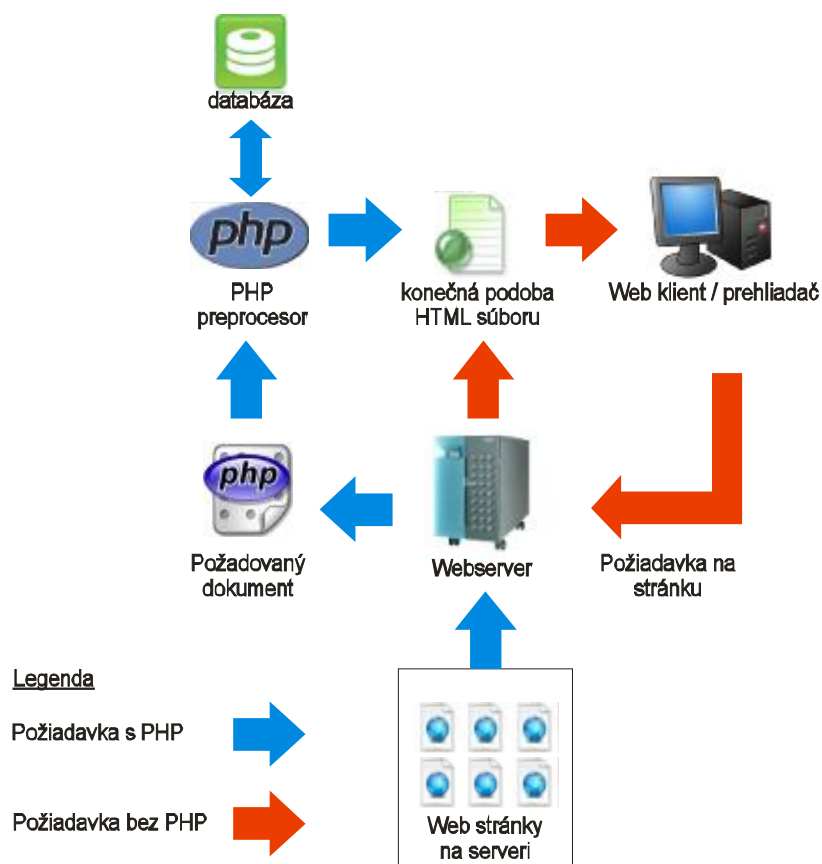
- PHP skripty nie sú vôbec závislé na použitom prehliadači, pretože ten s nimi nepríde do styku. Všetku prácu si vykoná server a do prehliadača dorazí len prosté HTML, ako je ilustrované na obrázku Obr.2
- Časti PHP kódu sa do skriptov vkladajú pomocou špeciálnych značiek, ktorých podrobnejší popis môžeme nájsť v [3]. Do úvahy prichádzajú nasledujúce možnosti kombinácie HTML a PHP kódu v jednom súbore:

1. Celý súbor môže byť jeden dlhý PHP kód. Potom začína a končí znakmi „<?“ a „?>“
2. V súbore môže byť kód PHP len na jednom určitom mieste.
3. PHP kód môže byť v súbore rozdelený na viacero miest.
4. Súbor sa síce môže tváriť ako php skript, ale v skutočnosti je to normálny HTML súbor a PHP kód v ňom vôbec nie je.

História PHP

PHP vzniklo a stále sa najviac používa ako serverový skriptovací jazyk pre HTML stránky. Ako tvrdí [15], PHP, pôvodne známe ako Personal Home Pages, predstavil na jeseň 1994 Rasmus Lerdorf. Napísal ho ako prostriedok pre záznamy o čitateľoch svojho online životopisu. Prvá verzia bola uvoľnená počiatkom roku 1995, kedy Rasmus zistil, že pokiaľ projekt uvoľní ako open-source, ľudia opravia chyby, ktoré narobil. Prvá verzia bola veľmi prostá a mala jednoduchý parser, ktorý rozoznal niekoľko špeciálnych makier a poskytoval niekoľko vtedy bežne užívaných funkcií. V polovici roku 1995 bol parser prepísaný a premenovaný na PHP/FI verzie 2., ktorý Rasmus pridal, aby sa PHP vysporiadalo s narastajúcimi požiadavkami webových stránok. Bola taktiež pridaná podpora MySQL. PHP/FI prešlo mohutným rozvojom a začali prispievať ďalší ľudia. V polovici roku 1997 prepísali Zeev Suraski a Andi Gutmans hlavný parser a PHP sa zmenilo na projekt celej skupiny. Tým boli položené základy PHP3, pomenovaného PHP: Hypertext Preprocessor. Posledná verzia, PHP4, je ďalší kompletný prepis od Suraskiho a Gutmansa a je postavená na Zend engine (viac v [2]). PHP má teraz cez dve stovky pravidelných prispievateľov, ktorí pracujú na rôznych častiach projektu.

Obsahuje veľké množstvo rozširujúcich modulov, podporu všetkých obľúbených serverov a štandardnú podporu MySQL a ODBC. Princíp činnosti PHP je zjednodušene zobrazený na Obr.2.



Obr.2 Schéma činnosti PHP[2]

3.2.4. Súvislosť SQL a PHP

Podľa [16] môže v roli klienta pre databázový server ako aplikácia vystupovať aj PHP. Teda PHP sa môže spojiť s databázovým serverom a využívať jeho služby, požadovať zápis, alebo získanie určitých údajov z databázy. To znamená, že vytvorené PHP skripty môžu obsahovať príkazy zapísané v jazyku SQL a PHP môže spracovávať výsledky SQL po vykonaní na databázovom serveri.

V praxi má každý SQL server svoj vlastný protokol, a preto PHP musí vedieť s každým SQL serverom komunikovať mierne rozdielne. PHP dokáže komunikovať s mnohými SQL servermi, mimo iné napríklad s Oracle, Sybase, Informix, MySQL, PostgreSQL.

3.2.5. Princíp JavaScript

Podľa [13] je JavaScript interpretovaný, multiplatformový programovací jazyk so základnými objektovo orientovanými schopnosťami. Univerzálne jadro jazyka bolo vložené do webových prehliadačov a rozšírené pridaním objektov reprezentujúcich okno prehliadača a jeho obsah. Táto *klientska* verzia JavaScriptu umožňuje vložiť do webových stránok vykonateľný obsah. Stránky sa tak stávajú dynamickými – môžu obsahovať najrôznejšie programy, ktoré komunikujú s užívateľom, riadia prehliadač, alebo dynamicky vytvárajú obsah HTML. Pri práci skriptu nie je treba kontaktovať server, všetku prácu skriptu zaistúje sám prehliadač.

Jadro jazyka syntakticky pripomína C++ a Javu (podrobnejšie v [4]). Avšak syntaxou podobnosť končí. JavaScript je jazyk bez typovej kontroly, čo znamená, že premenné nemusia mať špecifikovaný typ. A na viac JavaScript je čisto interpretovaný jazyk, na rozdiel napríklad od kompilovaných C a C++ a na rozdiel od Javy, ktorá je pred interpretáciou kompilovaná do bajtového kódu.

Čo sa týka bezpečnosti JavaScriptu. JavaScript na strane klienta neumožňuje čítanie a zapisovanie súborov, z dôvodov ktoré sú zrejmé. Nebol by potom problém jednoduchým programom úplne znehodnotiť obsah pevného disku a pod. Takisto nepodporuje prácu so sieťou, až na jednu dôležitú výnimku: môže donútiť webový prehliadač k načítaní ľubovoľného URL.

3.2.6. Princíp AJAX

Ako je popísané v [20], technológia **AJAX (Asynchronous JavaScript and XML)** je v poslednej dobe veľmi diskutovanou témou. Nejde o názov produktu, ale o časom ustálené označenie architektúry webových aplikácií, ktorá využíva **JavaScript** a **XML** pre načítanie dát zo servera bez nutnosti klasického tzv. reloadu HTML stránky.

Interakcia webovej stránky s užívateľom prebieha v podstate v reálnom čase, čo približuje webové rozhranie tomu desktopovému, podrobnosti v [4].

Princíp komunikácie webovej stránky so serverom je nasledujúci.

Pri spustení udalosti JavaScriptu (napr. onkeyup) na webovej stránke dôjde na spustenie nejakej návrhárom vytvorenej funkcie. Tá obsahuje vytvorenie objektu XMLHttpRequest, ktorý vyšle požiadavku serveru, z ktorého bola daná stránka načítaná. V prípade použitia J2EE na strane serveru je zavolaný servlet, ktorý požiadavku spracuje a do stránky vráti HttpServletResponse vo formáte XML s potrebnými dátami. Javascript túto XML správu dekoduje a na jej základe môže vykonať úpravu modelu DOM webovej stránky.

Tento prístup k tvorbe interaktívnejších webových aplikácií je známy už pomerne dlho, ale k jeho výraznejšiemu presadeniu dochádza až neskôr. Príčinou tohto rozšírenia je hlavne fakt, že vlastnosti JavaScriptu, kľúčové pre použitie AJAXu, sú dnes podporované všetkými hlavnými webovými prehliadačmi. Týka sa to najmä objektu XMLHttpRequest, ktorý mimochodom ani nie je súčasťou špecifikácie JavaScriptu.

AJAX je možné použiť napríklad k dosiahnutiu nasledujúcich efektov:

- automatické dopĺňovanie textu behom písania,
- real-time validácie formulárov s využitím dát z databázy,
- refresh len tých dát na stránke, ktoré je nutné udržiavať aktuálne,
- Pomalá inicializácia dát zo serveru v momente, kedy sú potrebné

Príkladom scenára posledného bodu je napríklad úvodná stránka nejakého elektronického obchodu, kde by sa nachádzal rozsiahly zoznam

výrobkov v akcií. Pri ukázaní kurzorom myši na obrázok každého produktu by sa mohol nejakým spôsobom objaviť jeho podrobný popis. Pri použití klasického riešenia by sa stránka musela načítať už so všetkými popismi výrobkov. V prípade nasadenia AJAXu by bolo možné dosiahnuť načítanie popisu zo serveru až pri ukázaní kurzorom myšou na obrázok, čím by sa mohlo načítanie úvodnej stránky e-shopu výrazne zrýchliť.

Výsledkom použitia AJAXu môžu byť skutočne zaujímavé vlastnosti webového rozhrania, ktoré sú zatiaľ iným spôsobom nedosiahnuteľné, AJAX však prináša vývojárom i radu problémov. Môžeme spomenúť napríklad možnosť zobrazenia kódu JavaScriptu v prehliadači, čo predstavuje zvýšené nebezpečenstvo napadnutia aplikácie. Tiež je nutné vziať do úvahy zníženie čitateľnosti kódu a sťaženie debugovateľnosti a testovateľnosti aplikácie. AJAX je možné vidieť napríklad na projektoch Google suggest a Google Maps.

3.2.7. Ďalšie použité pomocné programy a nástroje

- **Selenium**

Open Source nástroj pre testovanie webovej aplikácie. Pracuje na základe zaznamenávania vykonaných aktivít v aplikácií a následnom reprodukovani zaznamenaného scenára. Veľmi užitočný a silný nástroj ktorý dokáže zaznamenaný obsah (vyplňovanie formulárov, klikanie...) uložiť do súboru, prenášať, ba dokonca exportovať do formátu skriptovacích jazykov ako napr. Python – kde sa dajú ďalej presnejšie upravovať parametre, dopĺňať funkcie atď. Podrobnejšie informácie o projekte Selenium sú k dispozícii na internetových stránkach projektu [11].

- **GIT**

Vývojársky nástroj pre Linuxové systémy, slúžiaci na uchovávanie predošlých verzií vyvíjanej aplikácie. Zjednodušene povedané - existuje repozitár, kam sa po zmenách v zdrojovom kóde odosielajú vždy nové verzie aplikácie s tým, že staré sú stále dostupné. Napr. pri vývoji sa zistí že

Rozpis a funkcie jednotlivých tabuliek

kniha:

Slúži na ukladanie základného stavebného prvku databázy, ktorým je kniha. Pomocou dodatočnej funkcie *auto_increment* (táto funkcia pri každom uskutočnení záznamu do tabuľky vygeneruje nové číslo, ktoré bude použité ako *id* v kombinácii s primárnym kľúčom použitej na stĺpec *id*, je zaručená unikátnosť každého prvku z tabuľky *kniha*. K vytváraniu a odstraňovaniu kníh má prístup len užívateľ s učiteľskými právami. Študent sa s knihou stretne po prihlásení do systému pod svojim menom a heslom so študentskými právami, kedy si bude môcť vybrať prácu s knihou, ktorá mu bola učiteľom sprístupnená. Tabuľka obsahuje dva stĺpce:

- *id* - generuje sa automaticky pomocou už zmienenej mySql funkcie *auto_increment*
- *meno* - zadáva učiteľ pri vytváraní novej knihy.

lekcia:

Lekcia je po knihe druhou základnou jednotkou databázy. Zhromažďuje všetky cvičenia ako aj zadania domácich úloh, ktoré má študent k dispozícii. Tabuľka *lekcia* slúži na uchovávanie zoznamu existujúcich lekcí, ich názvov a identifikačných čísel. Podbloky tabuľky *lekcia* sú pripájané, podobne ako *kniha* -> *lekcia*, pomocou relačných vzťahov 1:x, kedy každá lekcia môže obsahovať ľubovoľný počet napr. cudzích slovíčok, a pod. Tabuľka obsahuje dva stĺpce:

- *id* - generuje sa automaticky pomocou funkcie *auto_increment*
- *meno* - zadáva učiteľ pri vytváraní novej lekcie.

kniha_ma_lekciu:

Táto tabuľka zabezpečuje priradenie ľubovoľného počtu lekcí ku konkrétnej knihe, zároveň obsahuje stĺpec *poradie*, ktorým učiteľ mení poradie jednotlivých lekcí v knihe tak, ako ich chce sprístupniť študentom. Je referenčná, čo znamená že obsahuje záznamy o konkrétnom identifikačnom

čísle knihy, ktoré patrí k identifikačnému číslu lekcie. Na tabuľku *kniha* je reláciou 1:x, pomocou tejto tabuľky naviazaná tabuľka *lekcia*, čo v praxi znamená, že jedna kniha môže mať obsahovať viacero lekcii. Obsahuje tri stĺpce:

- *id_kniha* – priradenie lekcie ku knihe
- *poradie* – určuje poradie lekcie v knihe

lekcia_ma_clanok:

Pomocou tejto tabuľky sa priraduje do lekcie článok. Za článok je pokladaný súvislý text, obsahujúci niekoľko viet, obmedzený na dĺžku 2000 znakov - (2000 znakov je dostačujúca a nezahlcujúca dĺžka článku). Článok obsahuje cudzie slová, ktoré sú študentovi vhodnou metódou prekladané. Každá lekcia môže obsahovať maximálne jeden článok. Cudzie slová sú ukladané do tabuľky *lekcia_ma_slovo*. Obsahuje dva stĺpce:

- *id_lekcia* – príslušnosť ku konkrétnej lekcii
- *text_clanok* – text článku v lekcii

lekcia_ma_doplnovacku_veta:

Administrátor (učiteľ) túto dvojicu tabuliek využíva pri vytváraní viet s jedným slovíčkom ktoré je umiestnené do výberového menu (tzv. *select box*) , kde je možné si vybrať medzi viacerými slovíčkami a zložiť tak vetu správneho významu. Tabuľka obsahuje tri stĺpce:

- *id_lekcia* - slúži na priradenie vety ku konkrétnej lekcii,
- *id_dopln_veta* - je automaticky generované identifikačné číslo vety, každá veta má svoje ID preto, aby sa k nej dali jednoducho priradiť slovíčka na výber.
- *dopln_veta* - uchováva samotný text vety.

lekcia_ma_doplnovacku_slovo:

Obsahuje štyri stĺpce:

- *id_lekcia* – definuje lekciiu ku ktorej je slovo priradené
- *id_dopln_veta* – označuje vetu v ktorej budú slovíčka použité na výber

- *dopln_slovo* – text samotného slova ktoré bude na výber,
- *spravne* – pre slovo ktoré je správne sa nastaví na hodnotu *true*.

lekcia_ma_pexeso:

Táto tabuľka obsahuje dvojice slov, v českom a cudzom jazyku, ktoré sa používajú pri skladaní hry – pexesa. Plní ju učiteľ pri vytváraní lekcie.

Obsahuje stĺpce:

- *id_lekcia* – priradenie dvojice slovíčok konkrétnej lekcii ,
- *id_slovo* – identifikačné číslo dvojice,
- *eng_slovo* – slovo v cudzom jazyku,
- *svk_slovo* – slovo v češtine,
- *kopirovane* – označuje slovíčka, ktoré boli skopirované z tabuľky *lekcia_ma_slovo*, kvôli vnútornej implementácii kopírovacej funkcie, kde slúžia ako zvýrazňované slová s prekladom.

lekcia_ma_slovo:

Slúži na ukladanie dvojíc slovíčok (české / cudzie). Slovíčka sú používané na zvýrazňovanie cudzích slov v článku pri cvičení. Tabuľka má rovnakú štruktúru ako tab. *lekcia_ma_pexeso*.

lekcia_ma_ulohu_veta:

Do tejto tabuľky vkladá učiteľ vety, ktoré budú použité v domácej úlohe pre danú lekcii. Vkladá sa iba veta v českom jazyku, na študentovi ďalej bude, aby vetu preložil, tá sa následne uloží do tabuľky *student_ma_ulohu*, kde sa uchová ako česká veta, ktorú študent prekladal, tak aj študentom preložená anglická veta, tento výsledok si môže pozrieť učiteľ a následne ho aj ohodnotiť. Tabuľka obsahuje tri stĺpce:

- *id_lekcia* – príslušnosť ku konkrétnej lekcii
- *id_veta* – generované identifikačné číslo každej vety v úlohe
- *svk_veta* – text vety ktorá má byť preložená, dĺžka vety je obmedzená na 300 znakov.

lekcia_ma_vetu:

Tabuľka slúži na ukladanie dvojíc česko-cudzojazyčných viet, určených na cvičenie pre študentov. Študentovi bude ukázaná česká veta, ktorú má v rámci cvičenia preložiť. Po ukončení prekladu uvidí správny preklad konkrétnej vety, vložený do databázy učiteľom. Skladá sa zo stĺpcov

- *id_lekcia* – príslušnosť ku konkrétnej lekcii
- *id_veta* – vygenerované identifikačné číslo vety
- *eng_veta* – cudzojazyčný text vety, vložený učiteľom, ktorý slúži študentovi na kontrolu správnosti jeho prekladu pri cvičení
- *svk_veta* – zdrojová veta v českom jazyku, ktorú má študent za úlohu preložiť

prava_skupin_kniha:

Pomocou tejto tabuľky sa povoľuje prístup pre jednotlivé skupiny ku konkrétnym knihám. Tieto práva pridáva učiteľ. Povoľovanie kníh je riešené tak, že každý záznam v tabuľke zodpovedá jednej knihe povolenej pre jednu skupinu. To znamená, že pokiaľ je potreba povoliť viacero kníh jednej skupine, pridávajú sa ďalšie záznamy s číslom skupiny a číslami nových kníh.

Tabuľka obsahuje stĺpce:

- *id_skupina* – identifikačné číslo skupiny
- *id_kniha* – identifikačné číslo knihy

prava_skupin_lekcia:

Principiálne funguje rovnako ako tabuľka *prava_skupin_kniha*, s rozdielom, že sa v nej povoľuje prístup k jednotlivým lekciam.

Tabuľka obsahuje tri stĺpce:

- *id_skupina* - identifikačné číslo skupiny
- *id_lekcia* - identifikačné číslo lekcie
- *id_kniha* - identifikačné číslo knihy, v ktorej sa lekcia nachádza

skupina:

Je určená na uchovávanie informácií o existujúcich skupinách užívateľov, potrebné pre povoľovanie prístupových práv k jednotlivým knihám a lekciam bez nutnosti povoľovať prístup pre každého užívateľa zvlášť.

Obsahuje dva stĺpce:

- *id* – generované identifikačné číslo
- *meno* – meno skupiny, zadané učiteľom

uzivatel:

Tabuľka obsahuje údaje o konkrétnych užívateľoch. Dáta sa do tabuľky pridávajú pri registrácii nového užívateľa pri vstupe na stránku e-learningu.

Definuje užívateľov, ktorí sú oprávnení vstúpiť do systému, ako učitelia, tak aj ako študenti. Obsahuje stĺpce:

- *id* – vygenerované unikátne identifikačné číslo užívateľa
- *typ_uctu* – označuje typ užívateľského účtu. Môže byť ucitel / student
- *meno* – krstné meno užívateľa (obmedzenie na 30 znakov)
- *priezvisko* – priezvisko užívateľa (obmedzenie na 30 znakov)
- *heslo* – heslo ktoré si stanoví užívateľ. Je ukladané vo forme tzv. hashu kvôli zabezpečeniu komunikácie pri odosielaní hesla do / z databázy pomocou php funkcie MD5
- *email* – e-mailová adresa užívateľa, slúži ako kontakt na užívateľa, ale aj ako prihlasovacie meno, pri vstupe do systému
- *akceptovany* – je pomocná hodnota, ktorá dokazuje, či bol užívateľský účet študenta autorizovaný učiteľom. Ak je hodnota 0, užívateľ nie je pripustený do systému. Ak je hodnota 1, má povolený prístup.
- *skupina* – príslušnosť k študijnej skupine. Pri registrácii nového užívateľa si tento vyberá študijnú skupinu ku ktorej do ktorej sa chce prihlásiť.
- *body* – ukladá sa počet bodov dosiahnutý študentom
- *hodnotit* – ak študent odovzdal nejakú domácu úlohu a tá čaká na ohodnotenie, je táto položka nastavená na hodnotu 1. Ak neodovzdal úlohu, alebo ju už učiteľ ohodnotil, je nastavená na 0.

student_ma_cvicenie_veta:

V tejto tabuľke sa ukladajú záznamy o výsledkoch prekladov viet pri cvičení v lekciách. Preložené vety sú ponúknuté učiteľovi na kontrolu. Štruktúra tabuľky je rovnaká ako v tabuľke *student_ma_ulohu*.

student_ma_ulohu:

Do tejto tabuľky sa ukladá text študentom vypracovanej domácej úlohy, ktorá je súčasťou každej lekcie. Podľa tohto textu učiteľ pridáva študentovi body.

Stĺpce v tabuľke:

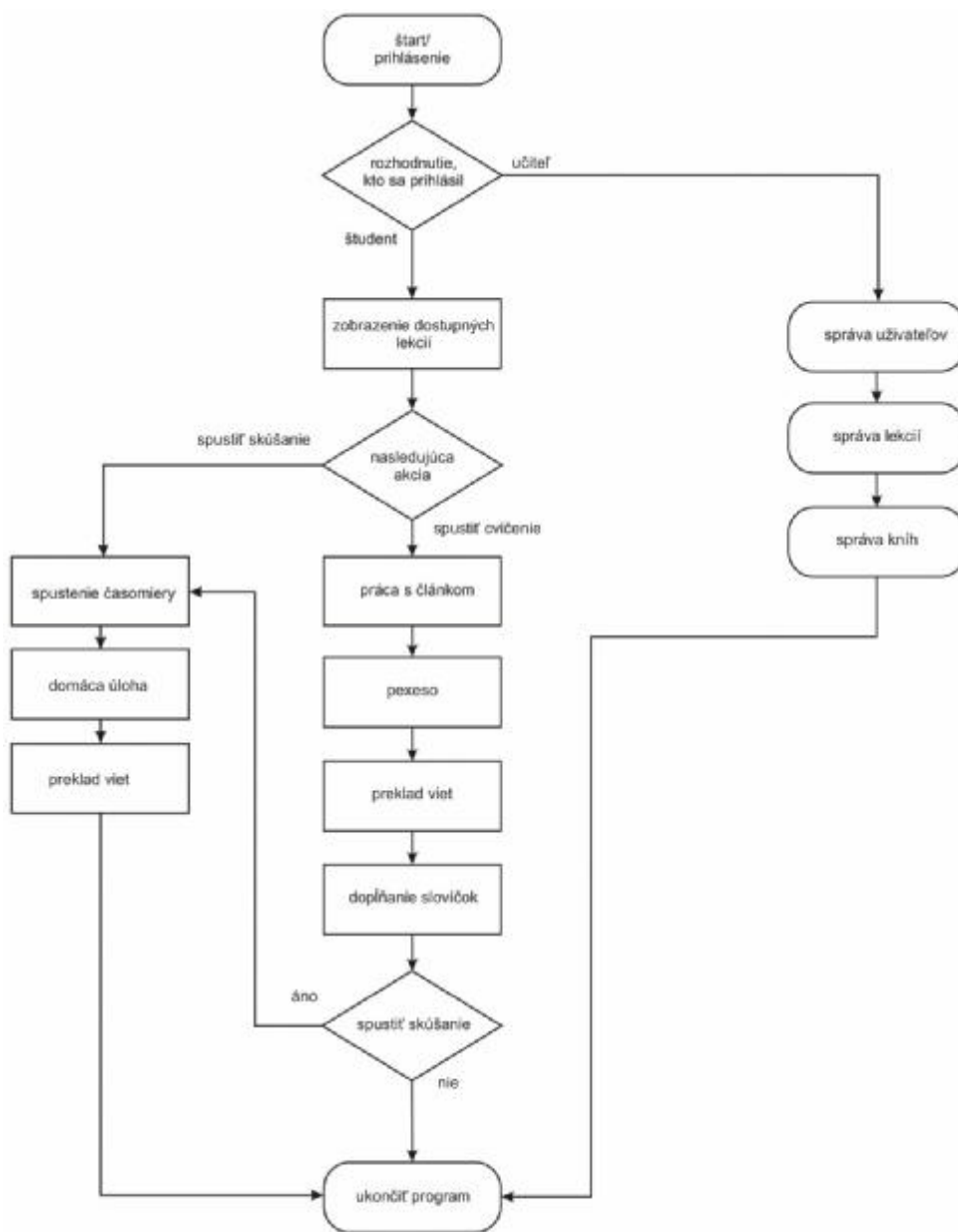
- *id_student* – identifikačné číslo študenta, ktorý vypracoval domácu úlohu
- *id_lekcia* – identifikačné číslo lekcie ktorá obsahovala danú odpoveď
- *id_odpoved* – identifikačné číslo konkrétnej odpovede
- *svk_veta* – znenie vety preloženej do českého jazyka študentom
- *eng_veta* – predloha vety, ktorá mala byť preložená, v cudzom jazyku
- *body* – počet bodov pridelených učiteľom za jednu vetu z domácej úlohy

uloha_hodnotenie:

Táto tabuľka je určená na uchovávanie informácie o tom, či bola odovzdaná domáca úloha z danej lekcie daným študentom. Ak áno, učiteľovi sa zobrazí možnosť konkrétnu domácu úlohu opraviť. Tabuľka obsahuje stĺpce:

- *id_student* – identifikačné číslo študenta
- *id_lekcia* – identifikačné číslo lekcie v ktorej bola / nebola odovzdaná domáca úloha
- *stav* – stav odovzdávania domácej úlohy: odovzdaná/neodovzdaná

Na nasledujúcom vývojovom diagrame je zobrazený predpokladaný pohyb užívateľa aplikáciou vid' Obr. 4



Obr.4 Vývojový diagram aplikácie

3.2.9. Popis grafického rozhrania aplikácie:

Úvodná stránka je zložená z komponentov jazyka pre tvorbu internetových dokumentov HTML, vzhľad a rozloženie objektov na stránke sú definované pomocou kaskádových štýlov – CSS v externom dokumente, ktorý je importovaný do hlavného súboru *index.php*.

Hlavná stránka je popísaná súborom *index.php*. Ako korektný HTML dokument sa skladá z hlavných častí dokumentu: hlavičky HEAD a tela BODY. Obsah hlavičky ako aj typ dokumentu majú podľa [19] určité presné pravidlá, ktoré je treba dodržať, konkrétne napríklad:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
```

Obr.5 Typ HTML dokumentu v súbore index.php

Časť kódu programu zobrazená na Obr.5 znamená, že dokument je typu tzv. HTML 4.01 Strict. Tento štandard býva definovaný ako obmedzená verzia HTML 4.01 ktorá sa viac ako na výzor zameriava na štruktúru. Použitím HTML 4 Strict môže autor dosiahnuť prístupný, bohato štruktúrovaný dokument, ktorý je ľahko prispôsobivý na použitie s kaskádovými štýlmi. Ďalej definuje primárny XML menný priestor - tzv. *Namespace* ktorý je použitý na celý dokument. *Namespace* je kolekcia kľúčových slov identifikovaných pomocou URI, ktoré sú použité v XML dokumente ako názvy objektov a atribútov.

Hlavička dokumentu

Pomocou TITLE je definované meno stránky ktoré sa zobrazí na hlavnej lište, alebo záložke internetového prehliadača. *Meta tag* je zvláštna značka, ktorá sa uvádza do hlavičky stránky. V každej stránke je potrebné použiť *meta tag* pre kódovanie znakov v dokumente. Tu (Vid' Obr.6) je použité štandardné kódovanie UTF-8. Pomocou značky GENERATOR definujeme názov textového editoru použitého na vývoj aplikácie. Ďalej sú sprístupnené externé javascriptové súbory: „jquery-1.2.6.js“ - *jquery* je javascriptova knižnica, ktorá umožňuje pohodlné a silné formátovanie dizajnu HTML stránky a jej komponentov, umožňuje množstvo špeciálnych efektov a vzhľadov pre HTML komponenty. Súbor "boxover.js" je tzv. plugin ktorým sa dá dosiahnuť zobrazenie podrobnejšieho popisu objektu v podobe informačného okna ktoré sa zobrazí pri nejakej udalosti komponentu (napr. pri prechode myšou nad objektom sa zobrazí bližší popis objektu). Súbor zxml.src.js je knižnica ktorá rozširuje možnosti práce s technológiou AJAX, je odporúčaná v [20]. Posledný vnorený je súbor "styles.css". Je to súbor v ktorom sú definované kaskádové štýly výzoru takmer všetkých HTML objektov v dokumente.

```
<head>
  <title>e-learning system</title>
  <meta http-equiv="content-type" content="text/html; charset=utf-8" />
  <meta name="generator" content="Geany 0.14" />
  <script type="text/javascript" src="./plugins/jquery-1.2.6.js"></script>
  <script type="text/javascript" src="./plugins/boxover.js"></script>
  <script type="text/javascript" src="./js_include/zxml.src.js"></script>
  <link rel="stylesheet" type="text/css" href="./css/styles.css" />
  <SCRIPT LANGUAGE="JavaScript">
</head>
```

Obr.6 Hlavička súboru index.php

Stránka je vizuálne aj funkčne rozdelená na štyri hlavné rámce. Toto rozdelenie bolo realizované pomocou HTML objektu TABLE, čo je tabuľka. Tabuľka je zložená z riadkov TR a stĺpcov TD. Pomocou vhodnej kombinácie týchto prvkov, ako aj nastavovania ich správnych vlastností bola vytvorená tabuľka, fungujúca ako navigačné a operačné prostredie spoločné pre obe časti programu: "študent" aj "učiteľ".

3.2.10. Vlastná funkcionálna aplikácie

Pri návšteve stránky je užívateľovi ponúknuté úvodné grafické rozhranie zobrazené na Obr. 7. V tejto úrovni programu má užívateľ dve možnosti postupu. Ako existujúci užívateľ sa môže prihlásiť do systému (Ako učiteľ, alebo ako študent). K tomu potrebuje registrovaný a povolený užívateľský účet (vysvetlené v ďalšom texte). Pre úspešné prihlásenie do systému je potrebná e-mailová adresa a heslo, stlačenie tlačidla „login“

Ako nový – neregistrovaný užívateľ sa má možnosť zaregistrovať do systému a následne mu môže byť účet aktivovaný. Registrácia prebieha pomocou stlačenia tlačidla „novy uzivatel“.



Obr.7 Úvodná obrazovka programu

Rámec „banner“

Prvý riadok tabuľky reprezentujúci *Banner rámec* (vid' Obr.7) slúži na zobrazenie hlavného loga (*banneru*) stránky. Toto bolo dosiahnuté umiestnením obrázku na pozadie bunky tabuľky, nastavenom pomocou HTML atribútu BACKGROUND-IMAGE v externom súbore s CSS štýlom. Takmer každý nasledujúci HTML objekt používa pri svojom zobrazení externý súbor s CSS štýlom, takže to v ďalšom texte nebudem zmieňovať.

Navigačný rámec

Pomocou tohto rámca (vid' Obr.7) je zabezpečená navigácia medzi jednotlivými oknami v oboch častiach aplikácie. Pre tento účel bol vytvorený súbor *navigacia.php* ktorý obsahuje jednoduchý smerovací mechanizmus, rozpoznávajúci obsah premennej "clanok". Táto je nastavovaná na konkrétne názvy stránok pri kliknutí myšou na odkazy zobrazené v ľavom rámci. Samotné nastavovanie hodnôt premennej článok je zabezpečené pomocou súboru "admin_adresy.php". Tu je zoznam všetkých dostupných odkazov obsahujúci nastavovanie premennej "clanok" pri ich zvolení. Táto hodnota je ďalej posunutá súboru "navigacia.php", ktorý obsahuje funkciu, smerujúcu vyžiadané stránky do hlavného rámca.

Hlavný rámec

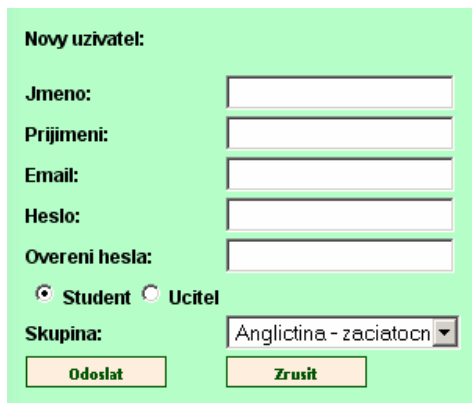
Rámec, do ktorého sú zobrazované všetky funkčné stránky aplikácie

Informačný rámec

Spodný rámec slúži na zobrazenie doplnkových informácií, ako napríklad autor aplikácie, dátum poslednej revízie a pod.

Registrácia nového užívateľa

Po stlačení tlačidla „**novy uzivatel**“ je užívateľ presmerovaný na stránku registrácie. (Vid' Obr.8)



Novy uzivatel:

Jmeno:

Prijmeni:

Email:

Heslo:

Overeni hesla:

Student Ucitel

Skupina:

Obr.8 Registrácia nového užívateľa

Užívateľ je požiadaný o vyplnenie jeho osobných a kontaktných informácií, e-mailová adresa bude použitá ako jeho užívateľské meno, potrebné pre prihlásenie. Zadané heslo musí byť z hľadiska bezpečnosti dlhé od šesť do pätnásť znakov a dva krát prepísané (heslo skopírované a vložené je neplatné). Užívateľ si ďalej vyberá typ účtu, ktorý si chce zaregistrovať (podľa toho či je študent alebo učiteľ). Pokiaľ si zvolí študentský účet, má na výber študijnú skupinu ku ktorej sa hlási (výber je zostrojený z už vytvorených študijných skupín – podrobné vysvetlenie vytvárania študijných skupín je vysvetlené v ďalších častiach) Ak je užívateľ nový učiteľ registrujúci sa do systému, prepne prepínač na hodnotu **Učiteľ**, tým pádom sa mu zablokuje možnosť výberu študijnej skupiny určeného pre študentov. Stlačením tlačidla **Odoslat**, sa zadané informácie odošlú na server a uložia do tabuľky užívateľ. Novo registrovaný užívateľský účet musí byť aktivovaný užívateľom s učiteľskými právami. Bez aktivácie účtu nie je možné sa prihlásiť do systému.

3.3. Prihlásenie do systému ako učiteľ

Je realizované štandardne zabezpečenou metódou: V tabuľke *uzivatel* je uložený tzv. *HASH* hesla, ktoré si užívateľ zvolil pri registrácii. Hash hesla je reťazec znakov, ktorý vznikol zašifrovaním pôvodného hesla pomocou jednosmernej hashovej funkcie (V PHP sa najčastejšie používa funkcia MD5). Pri každom pokuse o prihlásenie je pomocou tohto postupu vypočítaný hash zadaného hesla a porovnaný s hashom uloženým v databáze pre konkrétneho užívateľa podľa jeho prihlasovacieho mena, v našom prípade e-mailovej adresy. Ak sa hash z tabuľky zhoduje s hashom vypočítaným zo zadaného hesla, užívateľovi je povolený vstup do systému. V opačnom prípade ostáva na prihlasovacej stránke so zobrazeným varovaním vysvetľujúcim dôvod neúspešného prihlásenia.

Po prihlásení ako učiteľ sa uskutočnia nasledovné veci:

Do ľavého horného rohu navigačného rámca sa zobrazí text : „**Prihlasen:** <meno užívateľa>“, ďalej sa načítajú ikony slúžiace ako navigačné prostriedky pre pohyb v tejto časti aplikácie. Boli zvolené príjemne vyzerajúce ikony, kvôli priblíženiu sa tzv. *User friendly* charakteru aplikácie, čo v doslovnom preklade znamená že aplikácia je priateľská k užívateľovi, jej ovládanie je jednoduché a transparentné ako vo funkčnosti tak vo výzore. Ikony boli použité z projektu *IconEden* ktorý sa zaoberá vytváraním voľne dostupných ikon na základe GNU licencie pre webové ale aj iné typy aplikácií, kde sa vyžaduje použitie ikon. Bol zvolený súbor ikon, ktorý je voľne šíriteľný a vyhovujúci potrebám e-learningovej aplikácie [17]. Ikony sú opatrené tzv. *tooltip pluginom* - *Boxover* [18], vytvorenom v jazyku javascript, takisto poskytovaným ako voľne dostupné a šíriteľné riešenie. Tento plugin zabezpečuje nápovedu ku konkrétnemu objektu, v tomto prípade ikone. Pod pojmom nápoveda si predstavme pomocný text ktorý sa zobrazí pri prechode myšou ponad ikony, popisuje kam sa presmeruje aplikácia po kliknutí na ikonu.

Na obrázku Obr.9 je zobrazený obsah navigačného rámca aplikačnej časti „Učiteľ“ potom, ako sa užívateľ úspešne prihlási do systému.



Obr.9 Navigačný rámec aplikácie Učiteľ

3.3.1. Správa kníh

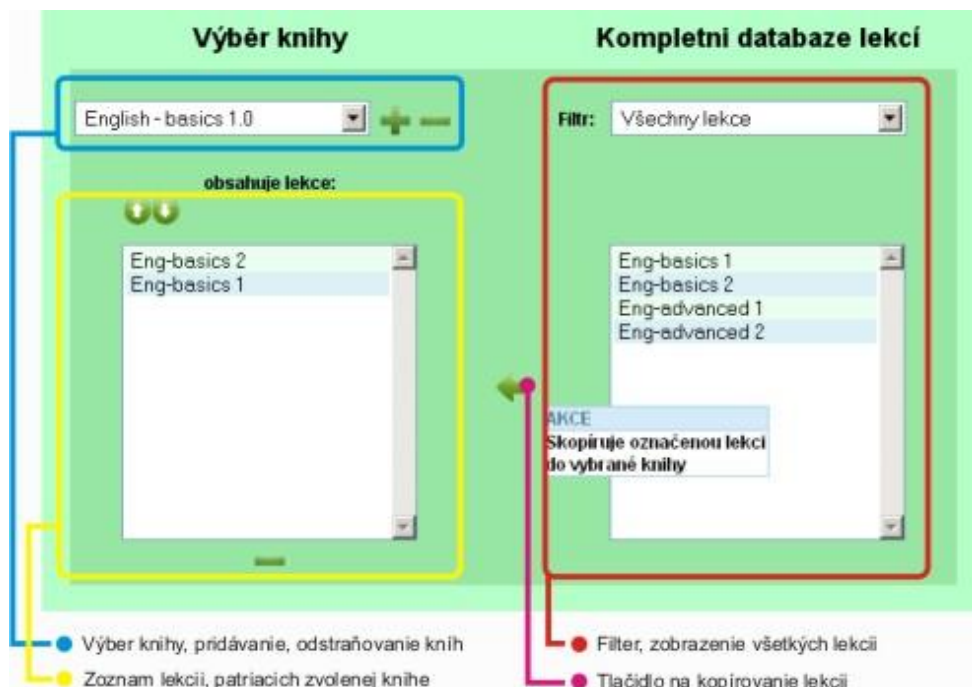
Po zvolení možnosti pre správu kníh bude užívateľovi, v tomto prípade učiteľovi, ponúknuté grafické rozhranie pre správu základného a najdôležitejšieho stavebného prvku databázovej štruktúry – knihy. Knihu si je možné predstaviť ako klasickú vyučovaciu knihu (učebnicu), ktorá sa skladá z jednotlivých lekcí, ktoré zas ďalej pozostávajú z menších častí. Časť aplikácie – správa kníh učiteľovi umožňuje: Vytvorenie novej knihy: po stlačení tlačidla „+“ sa objaví tzv. *promptbox*, javascriptový komponent, ktorý sa používa v prípade že chceme od užívateľa textový vstup a zároveň nechceme vkladať komponent pre vstupný text priamo na stránku (kvôli vzhľadu / bezpečnosti). Komponent sa objaví podobne ako tzv. popup (vyskakovacie) okno. Pozadie (hlavné okno aplikácie) je zablokované pre pohyb užívateľa do tej doby dokedy užívateľ buď potvrdí zadaný názov knihy, alebo okno zruší. Po potvrdení uloženia novej knihy je tento údaj odoslaný na server a uložený do tabuľky *knihy*, kde je pre ňu vygenerované nové

identifikačné číslo. Nasleduje znovu načítanie stránky, vo výberovom komponente sa už objaví novo vytvorená lekcia. Odstránenie existujúcej knihy: stlačením tlačidla „-“ docielime vymazanie existujúcej označenej knihy z databázy. Vymazaním knihy užívateľ zároveň vymaže aj lekcie, ktoré kniha obsahuje. Po vykonaní tejto operácie sa stránka opäť znovu načíta a vymazaná lekcia už nie je na výber. Zmena poradia lekcie v knihe: pomocou šípok „↑“ a „↓“ je učiteľ schopný meniť poradie lekcií v konkrétnej knihe. Táto funkcia je dôležitá pri vytváraní nových kníh s použitím kopírovania lekcií, kedy je cieľom použiť už existujúce lekcie ale zoradiť ich v inom poradí oproti knihe z ktorej boli kopírované. Pomocou šípok sa označená lekcia pohybuje ako je požadované. Toto je zabezpečené pomocou stĺpca *poradie* do tabuľky *kniha_ma_lekciu*, kde sa pomocou radiaceho algoritmu v tomto stĺpci menia hodnoty a v závislosti na tom sú položky zobrazované.

Odstránenie lekcie z knihy je zabezpečené pomocou tlačidla „-“ umiestnenom pod zoznamom lekcií, ktorá zvolená kniha obsahuje. Po stlačení tlačidla sa na server odošle požiadavka na odstránenie väzby kniha : lekcia, vymaže sa jeden záznam z tabuľky *kniha_ma_lekciu*.

- Filter: je nástroj na filtrovanie lekcií podľa názvu knihy ktorá ich združuje. Ďalšia dôležitá funkcia filtrovania je možnosť tento filter vypnúť a tým pádom zobrazíť všetky lekcie, ktoré boli v systéme vytvorené. Z tejto časti sa dá nasledovne lekcia kopírovať do inej knihy.

- Kopírovanie lekcie (viď Obr.10): Realizované stlačením tlačidla (šípky) „←“. Po vyfiltrovaní lekcií podľa požiadaviek myšou označíme jednu lekciu a po stlačení šípky smerujúcej vľavo (do druhej knihy) sa lekcia skopíruje. Skopírujú sa takisto všetky jej časti, o ktoré sú bližšie rozobrané v sekcii „Správa lekcií“. Kopírovanie je ošetrené proti skopírovaniu lekcie do knihy ktorá už takúto lekciu obsahuje (vyhodnocuje sa podľa názvu lekcie).



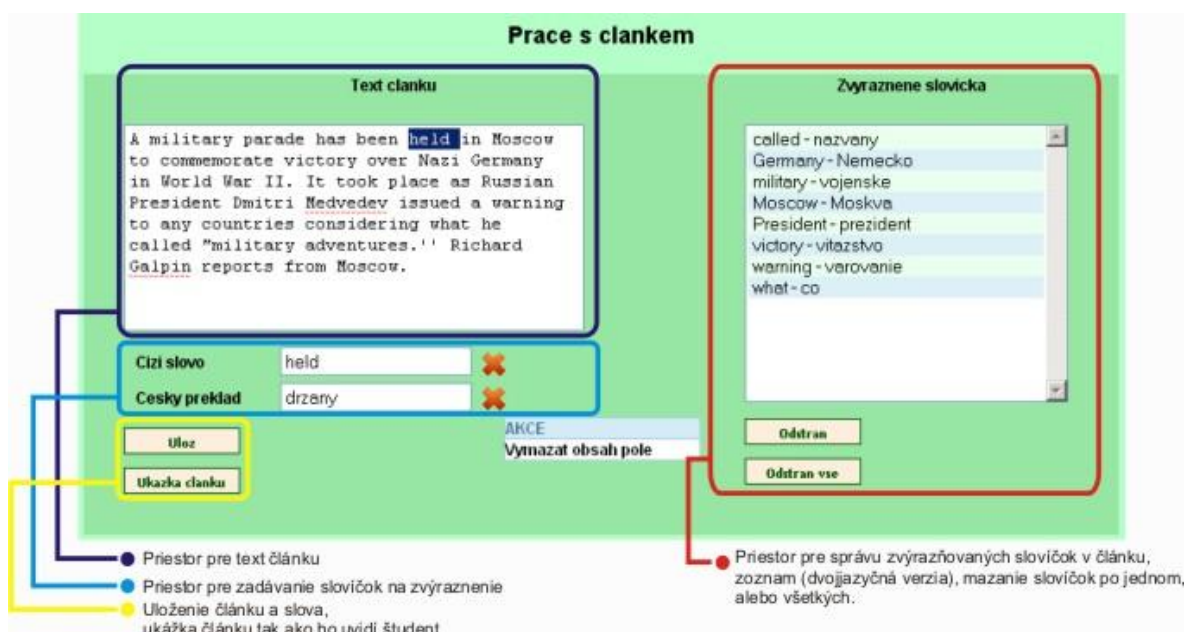
Obr.10 Rozloženie komponentov v časti pre správu kníh

3.3.2. Správa lekcií

Po zvolení cesty pre vstup do sekcie správy lekcií je učiteľov internetový prehliadač presmerovaný do časti, kde môže vytvárať, mazať alebo upravovať jednotlivé lekcie. Kvôli vysloveným požiadavkám je kompletná správa lekcie na jednej stránke, čo sa môže na prvý pohľad zdať neprehľadné, ale naučiť sa so systémom pracovať zaberie veľmi málo času, ovládanie je intuitívne. Jednotlivé časti pre správu lekcií zahrňujú:

Práca s článkom

V zadaní práce bola časť, ktorá hovorila o potrebe prítomnosti tzv. článku v cudzom jazyku, v ktorom bude učiteľ schopný označiť ľubovoľné slovo, alebo frázu (textový reťazec), priradiť mu jeho, do Českého jazyka preloženú hodnotu a následne umožniť jeho zvýraznenie v texte. Pri prechode myšou nad takto zvýraznené slovo sa v predtým zmieňovanom *tooltip* okne zjaví preložená hodnota slova. Ako prvú vec je treba vložiť text v cudzom jazyku do HTML objektu TEXTAREA. Nasleduje zvýrazňovanie cudzích slovíčok v článku. Toto sa realizuje tak, že označíme konkrétne slovo z článku (pomocou kurzoru myši – ťahaním a označovaním, alebo dvojklikom na dané slovo), označené slovo sa objaví v komponente *Cizi slovo*, ktorá je umiestnená hneď pod článkom. Ďalšia potrebná akcia je vloženie správneho prekladu pre označené slovo: Pomocou klávesnice sa zadá textový reťazec reprezentujúci preklad slova. Po stlačení tlačidla *Ulož* je uložený text článku ako aj označené a preložené slovíčko. Pri každom uložení sa kontroluje, či sa zmenil text článku, ak nie, neposiela sa ten istý znova, ak sa text zmenil, aktualizuje sa v databáze. Rozloženie komponentov v časti pre prácu s článkom je zobrazené na Obr.11.



Obr. 11 Správa lekcie – časť pre prácu s článkom

Časť cudzie slová – pexeso

V tejto časti stránky, vid' Obr.12, je učiteľovi umožnené pridávať a odstraňovať cudzie a preložené slovička, ktoré budú nasledovne použité v jazykovom cvičení pre študenta realizovaného na štýl hry Pexeso. Študentovi sa načíta zoznam vložených slovičok do grafického útvaru podobného rozdanej hre - pexeso. Je to mriežka v ktorej je do každej bunky načítaná dvojica slov (anglické - zobrazené a české - skryté). Pri načítaní stránky s pexesom sa vykreslí HTML objekt TABLE, tabuľka, do ktorej buniek sú vypísané anglické slovička v náhodnom poradí. Slovička sú opatrené hypertextovým odkazom, pomocou ktorého sa pri kliknutí myšou slovo zmení na preložené a naspäť. Existuje možnosť skopírovania slovičok z časti pre úpravu článku pomocou tlačidla "Skopiruj z clanku".

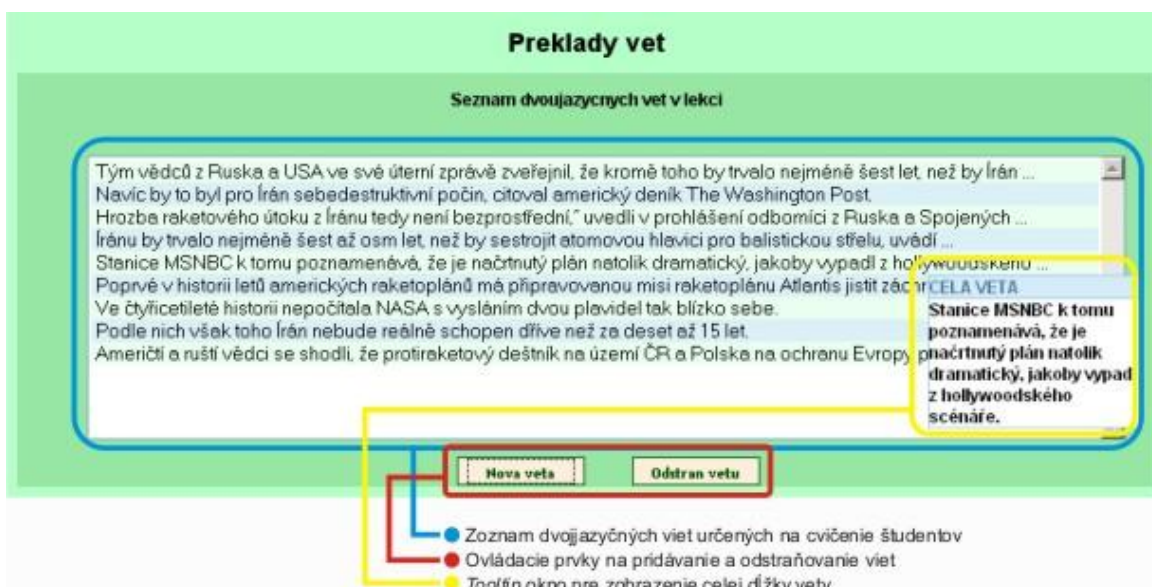


Obr. 12 Správa lekcií – časť pre prácu so slovičkami pre pexeso

Časť pre pridávanie dvojjazyčných viet

Na preklady pri cvičení študenta. V tejto časti stránky učiteľ pridáva do lekcie vety, ktoré budú študentovi ponúknuté na preklad v rámci cvičenia. Učiteľ stlačí tlačidlo na pridanie novej vety (Vid' Obr.13). Nasleduje zobrazenie tzv. *prompt boxu* – elementu jazyka javascript, ktorý slúži na získanie vstupu

od užívateľa. Tento element sa objaví dva krát, prvý krát pre umožnenie zadania vety v českom jazyku, druhý krát pre vetu v jazyku cudzom. Pri zadávaní vety v cudzom jazyku je česká veta stále zobrazovaná aby sa nemohlo stať, že užívateľ zabudol jej znenie. Po úspešnom pridaní vety do databázy sa zobrazí do kontrolného panelu ktorý zobrazuje existujúce vety v lekcii. Dlhšie vety by sa nevmestili do šírky tohto komponentu, preto bol opatrený opäť *tooltipom* ktorý pri prechode myšou nad jednotlivými vetami zobrazí znenie celej vety (aj tú časť ktorá sa nevmestila do štandardného zobrazenia). Maximálna povolená dĺžka vety ako v Českom tak aj v cudzom jazyku je 150 znakov. Po prekročení tohto limitu sa objaví informačná správa.

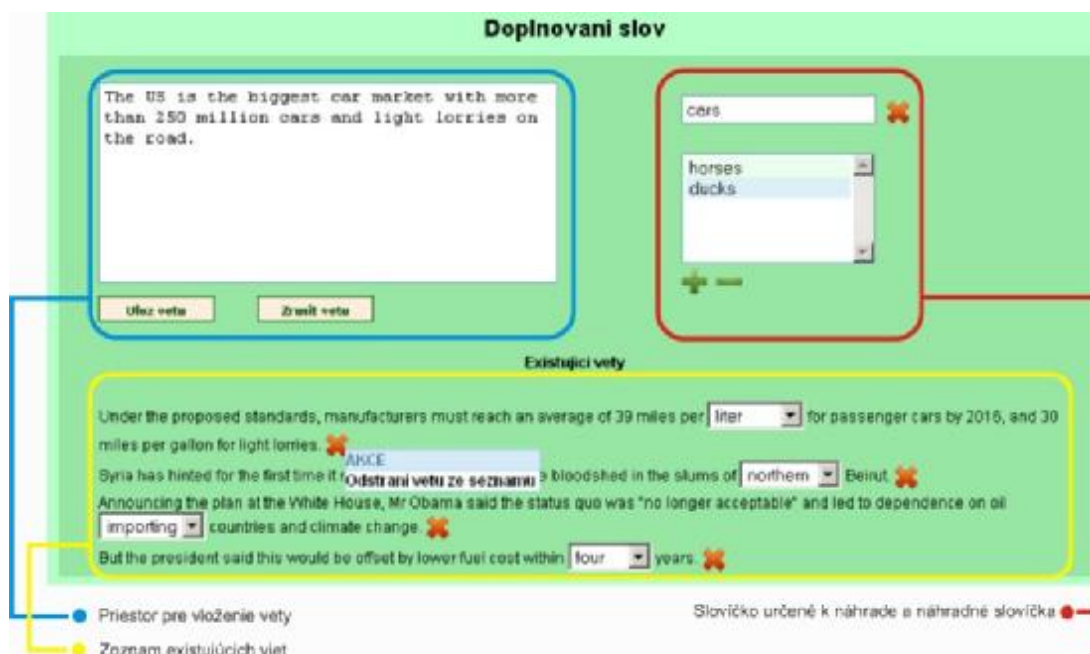


Obr. 13 Správa lekcii – časť pre zadávanie viet na preklad

Časť pre vytváranie viet so slovami na výber

Jednou z požiadaviek bolo vytvorenie systému, ktorý dokáže nahradiť zvolené slovo vo vete za HTML komponent *Select Box*. Pomocou tohto systému vznikne veta s určitou časťou (slovom, alebo ľubovoľným reťazcom) nahradenou komponentom *Select Box* ktorý slúži ako zoznam ponúkaných možností na výber. Študent má za úlohu vybrať správnu možnosť z ponúknutého výberu a tým skompletizovať vetu na správny tvar – zadaný učiteľom. Do priestoru pre vloženie vety (Vid' Obr. 14) sa zadá veta v cudzom jazyku, slovičko na náhradu sa označuje rovnakým spôsobom ako v časti pre

správu článku, t.j. označením pomocou kurzora myši, alebo dvojitém kliknutím – ktorým sa označí slovo oddelené na začiatku aj na konci medzerou. Po označení sa slovo zobrazí v pravej časti, v textovom poli. Pomocou tlačidla „+“ sa pridávajú slová, ktoré budú ponúknuté (spolu so správnym) pri zobrazení vety pre študenta. Tlačidlom „-“ sa slová na výber odstraňujú. Po stlačení tlačidla „Ulož vetu“ sa veta spolu so slovami odošle do databázy, konkrétne do tabuliek: *lekcia_ma_doplnovacku_slovo* a *lekcia_ma_doplnovacku_veta*. Nakoniec je kompletná veta, v konečnom tvare, t.j. aj s komponentom umožňujúcim výber slov zobrazená do nižšej časti tohto úseku, na ukážku. Každá veta je doplnená tlačidlom „x“ pomocou ktorého je možné ju odstrániť.



Obr. 14 Správa lekcii – časť pre vytváranie viet s dopĺňovaním

Časť pre Vkladanie viet na domácu úlohu

Správa sa podobne ako časť pre vkladanie dvojjazyčných viet , s tým rozdielom že v tejto časti sa vkladá iba veta v Českom jazyku, keďže nie je potreba zobrazovať správny preklad vety zadanej učiteľom. Študent uvidí vloženú českú vetu, jeho úlohou bude vetu preložiť do cudzieho jazyka a odoslať. Učiteľ si odpoveď pozrie a ohodnotí. Vyobrazené na Obr. 15



Obr. 15 Správa lekcii – časť pre zadávanie domácej úlohy

3.3.3. Správa užívateľov

Táto časť je určená pre zobrazenie informácií o užívateľoch v jednotlivých skupinách, aktiváciu / deaktiváciu ich účtov, prezeranie nimi spracovaných cvičení a hodnotenie ich domácich úloh. Takisto je tu možnosť povolenia prístupu ku knihám a lekciam pre jednotlivé študijné skupiny – čo znamená že učiteľ môže sprístupňovať študentom lekcie podľa toho ako pokračuje vyučovanie.

Na Obr.16 je zobrazený a popísaný rozvrh stránky pre správu užívateľov.



Obr.16 Správa užívateľov - rozloženie prvkov

Oblasť pre správu užívateľských skupín

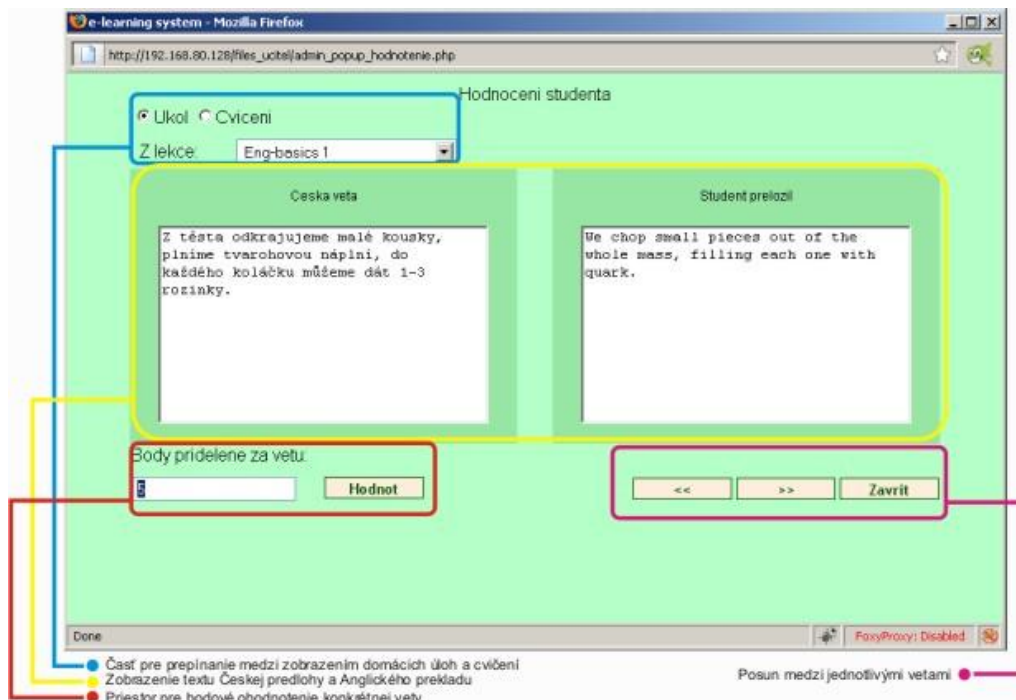
Pomocou tlačidiel „+“ a „-“ učiteľ vytvára / odstraňuje užívateľské skupiny. Tlačidlom „Ucitele“ sa otvorí vyskakovacie okno s informáciami o učiteľoch zaregistrovaných v systéme.

Oblasť pre sprístupnenie kníh a lekcií

Obsahuje dve okná obsahujúce knihy. V pravom okne sú všetky knihy, ktoré boli uložené v systéme, v ľavom okne sa nachádza zoznam kníh ktoré boli sprístupnené pre zvolenú užívateľskú skupinu. Tlačidlo v tvare zelenej šípky smerujúcej doľava vykonáva každé konkrétne povolenie knihy. V pravom okne sa označí kniha, stlačením tlačidla – šípky sa kniha objaví aj v ľavom okne - bola sprístupnená. Podobným spôsobom funguje povoľovanie jednotlivých lekcií v knihách. V tom prípade sa označí lekcia v ľavom okne a stlačí tlačidlo „Povolovani lekcií“, následne sa objaví vyskakovacie okno s rovnakým systémom ako povoľovanie kníh. Na zakázanie knihy / lekcie sa používa tlačidlo „-“, umiestnené pod ľavým oknom.

Informácie o študentoch a hodnotenie

Výpis do tabuľky s údajmi o identifikačnom čísle študenta, plnom mene, e-mailovej adrese a počte dosiahnutých bodov. V tabuľke sú ďalej umožnené akcie: aktivácia / deaktivácia účtu, hodnotenie študentových domácich úloh a prezeranie vykonaných cvičení. Pre vstup do časti pre hodnotenie domácich úloh slúži odkaz „Hodnotenie“ ktorý je umiestnený v tabuľke pre každého užívateľa. Po zvolení akcie hodnotenie sa objaví vyskakovacie okno (Vid' Obr.17), kde má učiteľ na výber medzi prezeraním cvičení alebo domácich úloh, ktoré ma zároveň možnosť aj bodovať.



Obr.17 Hodnotenie užívateľov – vyskakovacie okno

3.4. Prihlásenie do systému ako študent

Po prihlásení je študentovi ponúknutá možnosť výberu medzi vstupom do časti pre cvičenie alebo pre odovzdanie domácej úlohy. Zvolením požadovaného smeru pokračovania programu a stlačením tlačidla „Pokracovat“ nasleduje voľba konkrétnej lekcie. V listovacích komponentoch sú načítané knihy a lekcie ktoré boli povolené pre užívateľskú skupinu v ktorej sa študent nachádza. Po vybraní lekcie je študent presmerovaný do časti programu ktorú zvolil predtým, a to cvičenie, alebo domáca úloha.

Cvičenie

Do ľavého rámca sa načítajú odkazy na adresy pre pohyb v aplikácii študent. Názvy týchto odkazov sú dosť výrečné takže ich netreba ďalej popisovať. Každý odkaz (zobrazený na Obr. 18) presmeruje hlavný rámec aplikácie na požadovanú časť cvičenia.



Obr.18 Obsah ľavého rámca po prihlásení do systému ako študent

3.4.1. Práca s článkom

V tejto časti cvičenia sa zobrazí článok s anglickým textom, ktorý bol pre zvolenú lekcii vložený v administrátorskej časti pre správu lekcii. Text môže obsahovať označené (zvýraznené) cudzie slovíčka ktoré obsahujú už zmieňovaný *tooltip* – pri prechode kurzorom myšou nad nimi sa objaví

preložené slovičko. Pomocou práce s článkom (Obr. 19) sa má študent možnosť naučiť správny kontext pre používanie cudzích slovíčok v cudzojazyčnom texte, ako aj preklad označených slovíčok.

Nasledujúci obrázok ukazuje skutočný vzhľad časti pre prácu s článkom



Obr.19 Aplikácia študent – práca s článkom

3.4.2. Pexeso

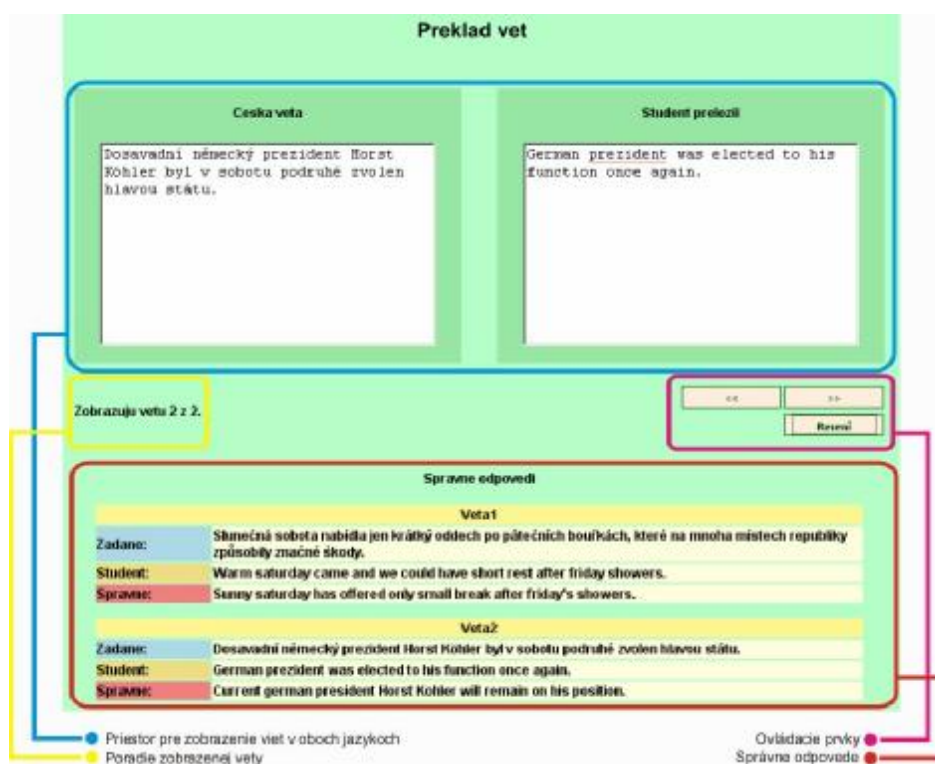
Názov pexeso nie je presný z hľadiska funkcie, ale výstižný z pohľadu rozloženia slovíčok na ploche stránky. Je to dvojrozmerné pole so slovíčkami v dvoch jazykoch. Tieto sú načítané do tabuľky (Vid' Obr. 20) a s riadkami a stĺpcami a vytvárajú tak grafické znázornenie podobné rozdanej hre – pexeso. Každé slovo je opatrené možnosťou zmeny na preložené slovo po kliknutí naň.



Obr.20 Aplikácia študent – slovička zobrazené ako pexeso

3.4.3. Preklad viet

Táto časť cvičenia pozostáva z okamžitého prekladu viet zadaných učiteľom s možnosťou zobrazenia správnych odpovedí hneď po ukončení cvičenia (Vid' Obr. 21). Pri vstupe do tejto sekcie sa overuje, či študent už toto cvičenie absolvoval a odoslal preložené vety učiteľovi. Z dôvodu znemožnenia viacnásobného posielania jedného cvičenia je po prvom odoslaní zakázaný prístup k opätovnému vypracovávaniu cvičenia a zobrazia sa len správne odpovede spolu so zadanými vetami. Ak študent spracováva cvičenie prvýkrát, do priestoru pre zobrazenie viet sa načítajú po jednej vety v oboch jazykoch. Medzi vetami je možné sa pohybovať pomocou tlačidiel so šípkami. Ak sú zodpovedané všetky vety, je povolené odoslať ich do databázy a tak sprístupniť lektorovi na nahliadnutie, hneď potom sú v spodnej časti obrazovky zobrazené ako zadané vety v oboch jazykoch, tak aj vety preložené študentom pre kontrolu.



Obr.21 Aplikácia študent – preklad viet v cvičení

3.4.4. Doplnovanie viet

Študentovi sa zobrazí zoznam viet (Obr. 22), v ktorých jedno slovo alebo časť vety bude nahradená komponentom *selectbox*, ktorý bude obsahovať viacero položiek na výber. Od študenta sa požaduje aby vybral položku, ktorá do vety patrí, aby bola gramaticky a významovo správne tak, ako ju do systému vložil učiteľ. Po stlačení tlačidla „Vyhodnotiť“ budú doplnené vety spracované a vyhodnotené. Správne tvary viet sa následne zobrazia v spodnej časti okna pre porovnanie.

Doplnovani vet

Under the proposed standards, manufacturers must reach an average of 39 miles per for passenger cars by 2016, and 30 miles per gallon for light lorries.

Syria has hinted for the first time it may send in troops to halt the bloodshed in the slums of Beirut.

Announcing the plan at the White House, Mr Obama said the status quo was "no longer acceptable" and led to dependence on oil countries and climate change.

But the president said this would be offset by lower fuel cost within years.

Spravne reseni

Under the proposed standards, manufacturers must reach an average of 39 miles per gallon for passenger cars by 2016, and 30 miles per gallon for light lorries.

Syria has hinted for the first time it may send in troops to halt the bloodshed in the slums of southern Beirut.

Announcing the plan at the White House, Mr Obama said the status quo was "no longer acceptable" and led to dependence on oil exporting countries and climate change.

But the president said this would be offset by lower fuel cost within three years.

● Časť pre zvolenie správnych slov vo vetách
● Časť so správnymi odpoveďami

Obr.22 Aplikácia študent – dopĺňanie slov do viet

3.4.5. Domáca úloha

Domáca úloha pozostáva zo sady viet v Českom jazyku, ktoré ma študent za úlohu preložiť. Predtým ako je študent pripustený k vypracovaniu a odovzdaniu domácej úlohy, dostane sa na stránku kde si prečíta pokyny a musí ešte raz potvrdiť svoju voľbu. Je to kvôli jedinému pokusu, ktorý má študent na to, aby úlohu zvládol. Po potvrdení a pristúpení k samotnému vypracovaniu úlohy sa do jedného zobrazovacieho komponentu (Vid' Obr. 23) načíta prvá zo zoznamu dostupných viet v Češtine. Takisto sa okamžite spúšťa časomiera, zobrazená v strede pod zobrazovacími komponentmi, ktorej hodnotu (dĺžku behu) určil učiteľ pri úprave lekcie. Študent vetu preloží a zapíše do druhého zobrazovacieho komponentu. Medzi vetami sa môže pohybovať pomocou tlačidiel so šípkami vľavo a vpravo. Keď sú všetky vety preložené, študent stlačí tlačidlo „Odeslať“, pomocou ktorého sa všetky odpovede odošlú do databázy, kde ich má k dispozícii učiteľ. V tomto okamžiku sa zastaví časomiera, domáca úloha sa v databáze označí ako už odovzdaná, čo znamená že ju študent už nemôže odovzdať druhý krát. Pri opätovnom pokuse vypracovať tú istú domácu úlohu bude informovaný že táto akcia nie je povolená. V prípade vypršania času skôr ako študent odoslal odpovede sa úloha označí ako odovzdaná ale prekladané vety sa neuložia, čo znamená, že študent stratil svoju šancu a príde o body.



Obr.23 Aplikácia študent – preklad viet ako domáca úloha

4. Záver

E-learningové systémy môžu byť veľmi výhodnou alternatívnou metódou pre doplnkové štúdium študentov ako na školách tak zamestnancov firiem, alebo pre celú širokú skupinu ľudí ktorá má záujem o túto formu vzdelávania.

Boli spomenuté výhody aj nevýhody využívania e-learningu, ako aj jeho možnosti použitia v praxi. V tejto práci bolo v rámci semestrálneho projektu vypracované teoretické riešenie zadania, ako aj základný funkčný model administrátorskej aplikácie, ktorý poslúžil ako základ pre ďalší vývoj. Nasledujúca časť práce na projekte bola venovaná aplikácii určenej pre študentov, realizácii zadaných požiadaviek, ako aj grafickej časti projektu.

V praktickej časti je nami navrhovaný projekt realizovaný ako aplikácia klient – server. Klientska aplikácia sa dá z pohľadu svojej funkčnosti a práv rozdeliť na aplikácie učiteľa a študenta. Ako úložisko dát slúžil v projekte databázový server MySQL, ku ktorému sa prihlasovali klientské aplikácie. Zvýšená bezpečnosť prístupu do systému bola dosiahnutá šifrovaním hesla zadávaného pri prihlasovaní pomocou hashovej funkcie MD5 a následným porovnávaním so zašifrovaným heslom uloženým v databáze. Praktické vyhotovenie aplikácie je realizované tak, aby sa dalo použiť v reálnom prípade.

Systém bol testovaný realizáciou na Linuxovej verzii Apache Web serveru, ktorý je v dnešnej dobe najpoužívanejší voľne šíriteľný webový server. Aplikácia je optimalizovaná pre internetový prehliadač Mozilla Firefox. Podmienky zadania boli splnené, aplikácia plne využíva programovací jazyk Javascript a technológiu Ajax pre čo najväčšie možné pohodlie užívateľa pri práci s programom. V texte boli rozobrané dôvody pre vytvorenie vlastnej funkčnej aplikácie namiesto použitia jedného z voľne dostupných riešení, ako je napr. Moodle. Funkčná verzia programu je pre testovacie účely dostupná na web-hostingovom serveri.

Zoznam použitej literatúry

- [1] Lacko, L. : PHP a MySQL - hotová řešení, CP Books, 2005. ISBN 80-251-1249-7
- [2] Kosek, J. : PHP podrobný pruvodce, GRADA 1999. ISBN 80-7169-373-1
- [3] Vojtěch Smidek a kol.: Základy jazyka PHP Praktické ukázky PHP aMySQL, KOPP, 2003, ISBN 80-7232-162-5
- [4] Ryan Asleson, Nathaniel T. Schutt: Vytváříme vysoce interaktivní webové aplikace, COMPUTER PRESS, 2006, ISBN 80-251-1285-3
- [5] William Horton, Katherine Horton: E-learningové nástroje a technológie, John Wiley & Sons, 2003, ISBN 0-471-444588
- [6] Kopecký, Kamil: E-learning (nejen) pro pedagogy, HANEX, 2006. 1. vyd. 125 s. ISBN 80-85783-50-9
- [7] Ruth Colvin Clark, Richard E. Mayer: e-Learning and the Science of Instruction: Proven Guidelines for Consumers and Designers of Multimedia Learning, Pfeiffer, ISBN 07-8798-683-6
- [8] Michael Allen: Designing Successful e-Learning, Pfeiffer, 2008, ISBN-10 07-8798-299-7
- [9] Marc J. Rosenberg: E-Learning: Strategies for Delivering Knowledge in the Digital Age (Hardcover), McGraw-Hill, 2007, ISBN 0071362681
- [10] Oficiálna www stránka programu Geany, URL: <http://www.geany.org/>
- [11] Oficiálna www stránka projektu Selenium, URL: <http://seleniumhq.org/>
- [12] Informačný portál nástroja GIT, URL: <http://git.or.cz/course/svn.html>
- [13] David Flanagan: JavaScript - Kapesní příručka - Pro aktivní webové stránky, HELION, 2004, ISBN: 83-7361-466-4
- [14] Online tutoriál, URL adresa: http://www.linuxsoft.cz/article.php?id_article=172
- [15] Online tutoriál, URL adresa: <http://gtk.php.net/manual1/cs/html/intro.whatis.php.history.html>
- [16] Online tutoriál, URL adresa: <http://www.zive.cz/Clanky/PHP---25-dil--uvod-do-databazi/sc-3-a-118655/default.aspx>

- [17] Zdroj voľne šíriteľných ikon, použitých v programe. URL adresa:
<http://iconeden.com/icon/milky-a-free-vector-iconset.html>
- [18] Zdroj pre voľne šíriteľný „Boxover“ plugin URL adresa:
<http://boxover.swazz.org/>
- [19] Kosek, Jiří: HTML - tvorba dokonalých WWW stránok, Grada, 1998, 296 s., ISBN:80-7169-608-0
- [20] Nicholas C. Zakas, Jeremy McPeak, Joe Fawcett: Ajax – profesionálne, Zoner Press, 2007, 672 s., ISBN: 978-80-86815-77-0

Prílohy

CD s elektronickou verziou projektu

CD obsahuje zdrojové kódy kompletného projektu. Súborov php a javascript, ako aj exportovanú databázu MySQL. Samotná konfigurácia serveru a nahrávanie súborov naň, ako aj spozojzdenie nebude nutné, pretože celý systém je funkčný na webhostingovom serveri pre testovacie účely.

Zjednodušený návod na použitie a predvedenie funkčnosti

1. Pomocou internetového prehliadača Mozilla Firefox navštívte virtuálny server <http://diploma.mine.nu/>
2. Prihláste sa pod užívateľským menom email: „kratochvil@vutbr.cz“ + heslo: „kratochvil“ ako učiteľ
3. Kliknutie na ikonu knihy – možnosť vytvárať, odstraňovať knihy, presúvať lekcie medzi jednotlivými knihami.
4. Kliknutie na ikonu lekcie – možnosť vytvárať, odstraňovať lekcie, úprava zloženia lekcii. Editácia všetkých súčastí
5. Kliknutie na ikonu užívateľa – vytváranie, odstraňovanie študijných skupín, zobrazenie zoznamu študentov do nich spadajúcich. Aktivácia /

deaktivácia študentských účtov, hodnotenie študentov, povoľovanie / zakazovanie prístupu študentských skupín ku knihám a lekciam.

6. Prihláste sa pod užívateľským menom email: „peter@vutbr.cz“ + heslo: „kratochvil“ ako študent
7. Zvoľte možnosť „cvičenie“, následne jednu zo sprístupnených lekcii.
8. V navigačnom menu postupne vstúpte do všetkých položiek a riad'íte sa pokynmi.
9. Po odovzdaní domácej úlohy sa odhláste a opakovane prihláste ako učiteľ v kroku 2.
10. V správe užívateľov nájdite študenta Peter Kratochvíl. Pri jeho mene zvoľte možnosť „hodnotiť“.
11. V objavenom okne nájdite lekciiu, s ktorou ste pracovali ako študent
12. Zobrazí sa zadanie domácej úlohy, ako aj preklad študenta, tento výkon môžete ohodnotiť bodmi.
13. Po prepnutí na položku cvičenie môžete vidieť študentovu aktivitu v preklade viet v cvičení. Toto nie je možné bodovať. Po zavretí vyskakovacieho okna sa zobrazí kompletný počet bodov. Ak sa údaj neaktualizoval, treba znovu načítať stránku (F5).