

Univerzita Pardubice
Fakulta elektrotechniky a informatiky

Online kurz výuky angličtiny
Filip Vondráček

Bakalářská práce
2018

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2017/2018

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Filip Vondráček**
Osobní číslo: **I15153**
Studijní program: **B2646 Informační technologie**
Studijní obor: **Informační technologie**
Název tématu: **Online kurz výuky angličtiny**
Zadávající katedra: **Katedra informačních technologií**

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

Cílem této práce je vytvoření funkční aplikace, která bude umožňovat online kurz výuky angličtiny. Aplikace bude umožňovat tyto funkcionality: přístup uživatelů do systému podle oprávnění, evidence uživatelů a dosažených výsledků v kurzu, test pro každou lekci, přístup k jednotlivým lekcím v závislosti na výsledcích testů, statistika úspěšnosti testů a generování sestav dle volitelně zadaných kritérií. V teoretické části bude řešerše systémů, které se zabývají touto problematikou. Praktická část bude obsahovat popis použitých technologií, návrh databáze, ER diagram s využitím "Crow's Foot" notace entity-relationship a aplikační řešení. Pro vytvoření aplikace bude využit skriptovací jazyk PHP nebo JAVA a databáze MySQL nebo Oracle.

Rozsah grafických prací:

Rozsah pracovní zprávy: **min. 30 stran**


Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

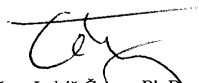
LACKO, Luboslav. Oracle - Správa, programování a použití databázového systému. Brno: Computer Press a.s., 2007. 573 s. ISBN 978-80-251-1490-2.
GROFF, James R. a Paul N. WEINBERG. SQL kompletní průvodce. Brno: Computer Press a.s., 2005. 936 s. ISBN 80-251-0369-2.
NARAMORE, Elizabeth, Jason GERNER, Scouarnec YANN LE and Timothy BORONCZYK. PHP 6, MySQL, Apache: Vytváříme webové aplikace. Brno: Computer Press a.s., 2009. 816 s. EAN:9788025127674.
DRUSKA, P. CSS a XHTML - tvorba dokonalých webových stránek krok za krokem, Grada, 2006. 200 s. ISBN: 80-247-1382-9.

Vedoucí bakalářské práce: **Ing. Miloslav Macháček, Ph.D.**
Katedra informačních technologií

Datum zadání bakalářské práce: **31. října 2017**
Termín odevzdání bakalářské práce: **12. května 2018**


Ing. Zdeněk Němec, Ph.D.
děkan




Ing. Lukáš Čegan, Ph.D.
pověřený vedením katedry

V Pardubicích dne 20. března 2018

Prohlašuji:

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

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

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 29. 4. 2018

Filip Vondráček

PODĚKOVÁNÍ

Mé poděkování patří Ing. Miloslavu Macháčkovi, Ph.D. za odborné vedení, trpělivost a ochotu, kterou mi v průběhu zpracování bakalářské práce věnoval. Děkuji také všem ostatním, a především rodině a přátelům za spolupráci při získávání zdrojů k vypracování této bakalářské práce.

ANOTACE

Cílem této bakalářské práce bylo vytvořit funkční aplikaci pro výukový kurz angličtiny v online podobě. Bakalářská práce je rozdělena na dvě části, a to na teoretickou a praktickou. Teoretická část se zabývá rešeršemi podobných systémů, které se touto problematikou již zabývají. V praktické části jsou popsány použité technologie PHP, MySQL, Javascript a framework Nette. Dále také popis výsledné webové aplikace, umožňující výuku rozdělenou na jednotlivé lekce a zaznamenávání výsledků jednotlivých uživatelů.

KLÍČOVÁ SLOVA

PHP, MySQL, angličtina, kurz, online, databáze

TITLE

Online English language course

ANNOTATION

The aim of this bachelor thesis was to create a functional application for an online course in English. The bachelor thesis is divided into two parts, theoretical and practical. The theoretical part deals with researches of similar systems, which are already dealing with this issue. The practical part describes the PHP, MySQL, JavaScript and Nette technologies. Also, a description of the resulting web application, allowing the lesson to be divided into individual lessons and recording individual users' results.

KEYWORDS

PHP, MySQL, english, course, Online, database

OBSAH

Seznam obrázků	10
Seznam tabulek	11
Seznam zkratk	12
Úvod	13
1 Teoretická část	14
1.1 Online výukové kurzy	14
1.2 Historické zákonitosti	14
1.3 Vývoj aplikačního SW	14
1.3.1 Vývoj webových aplikací	15
1.3.2 Vývoj databází	15
1.3.3 Vývoj programovacích jazyků	16
1.3.4 Druhy programovacích jazyků	16
1.4 Problematika systému	17
1.5 Rešerše podobných systémů	18
1.6 Duolingo.cz	18
1.6.1 Registrace	18
1.6.2 Prostředí	18
1.6.3 Klady	19
1.6.4 Zápory	19
1.7 Onlinejazky.cz	19
1.7.1 Registrace	19
1.7.2 Prostředí	20
1.7.3 Klady	20
1.7.4 Zápory	20
1.8 Englishme.cz	21
1.8.1 Registrace	21
1.8.2 Prostředí	21
1.8.3 Klady	22
1.8.4 Zápory	22
1.9 Angličtina-bez-biflovani.cz	22

1.9.1	Registrace.....	22
1.9.2	Prostředí.....	23
1.9.3	Klady.....	23
1.9.4	Zápory.....	23
1.10	Ohodnocení rešerší.....	24
1.11	Srovnání s vlastní aplikací.....	24
1.12	Srovnání v tabulce.....	25
2	Praktická část.....	26
2.1	Popis použitých technologií.....	26
2.1.1	XAMPP a Apache.....	26
2.1.2	PHP.....	27
2.1.3	MySQL.....	30
2.1.4	Nette.....	32
2.1.5	HTML.....	33
2.1.6	CSS.....	33
2.1.7	JavaScript.....	34
2.1.8	Netbeans.....	34
2.2	Aplikační řešení.....	34
2.2.1	Návrh systému.....	34
2.2.2	Uživatelské role.....	35
2.2.3	Registrace a přihlášení.....	35
2.2.4	Prostředí.....	36
2.2.5	Nová slovíčka.....	37
2.2.6	Procvičování překladu slovíček.....	38
2.2.7	Nová gramatika.....	38
2.2.8	Procvičování gramatiky.....	39
2.2.9	Test z lekce.....	40
2.2.10	Test s vlastním nastavením.....	40
2.2.11	Statistiky uživatelů.....	41
2.3	Databázový model.....	42
2.3.1	Tabulky.....	43
2.4	Ukázky zdrojových kódů.....	44

2.4.1	Použití MySQL v aplikaci	44
2.4.2	Použití PHP v aplikaci	45
2.4.3	Použití JavaScriptu v aplikaci.....	46
2.5	Bezpečnost dotazů	47
2.6	UML Use case diagram	48
2.7	UML Activity diagram	49
2.8	Instalační příručka.....	50
2.9	Uživatelská příručka	50
2.10	Adresářová struktura	50
Závěr	51
Použitá literatura	52
Přílohy	55

SEZNAM OBRÁZKŮ

Obrázek 1: Druhy programovacích jazyků	17
Obrázek 2: Ukázka základního výpisu textu v PHP	28
Obrázek 3: Ukázka základů v PHP	29
Obrázek 4: Ukázka příkazů skupiny DDL	31
Obrázek 5: Ukázka příkazů skupiny DML	31
Obrázek 6: Ukázka příkazů skupiny DCL	32
Obrázek 7: Ukázka registrace	36
Obrázek 8: Ukázka prostředí aplikace	37
Obrázek 9: Ukázka stránky s novými slovíčky	37
Obrázek 10: Ukázka stránky s procvičováním slovíček	38
Obrázek 11: Ukázka stránky s novou gramatikou	39
Obrázek 12: Ukázka stránky s procvičováním gramatiky	39
Obrázek 13: Ukázka stránky s testem z lekce	40
Obrázek 14: Ukázka statistiky první lekce	41
Obrázek 15: Ukázka databázového modelu	42
Obrázek 16: Ukázka pohledů	42
Obrázek 17: Ukázka získávání slovíček pro lekci	44
Obrázek 18: Ukázka získání celkového procentuálního postupu v kurzu	44
Obrázek 19: Ukázka vložení nového uživatele	45
Obrázek 20: Ukázka vykreslení formuláře pro změnu hesla	45
Obrázek 21: Ukázka získání dat pro vykreslení vyhodnocení	46
Obrázek 22: Ukázka změny obrázku při cvičení slovíček	46
Obrázek 23: Ukázka bezpečnosti prvního dotazu do databáze	47
Obrázek 24: Ukázka bezpečnosti druhého dotazu do databáze	47
Obrázek 25: Ukázka bezpečného dotazu do databáze	47
Obrázek 26: UML Use case diagram rolí	48
Obrázek 27: UML Activity diagram	49

SEZNAM TABULEK

Tabulka 1: Tabulka srovnání jednotlivých kurzů.....	25
-----------------------------------------------------	----

SEZNAM ZKRATEK

PHP	Hypertext Preprocessor
MySQL	My Structured Query Language
DDL	Data Definition Language
DML	Data Manipulation Language
DCL	Data Control Language
UML	Unified Modeling Language
XAMPP	Cross-Platform Apache MariaDB PHP Perl
CSS	Cascading Style Sheets
HTML	HyperText Markup Language
NoSQL	No Structured Query Language
BSD	Berkeley Software Distribution
ASP.NET	Active Server Pages
PHP/FI	Hypertext Preprocessor/Form Interpreter
HTTP	Hypertext Transfer Protocol
OOP	Objektově Orientované programování
BLOB	Binary Large Object
MVP	Model View Presenter
XML	Extensible Markup Language
URL	Uniform Resources Locator
IBM	International Business Machines

ÚVOD

Cílem této bakalářské práce je vytvořit online kurz anglického jazyka. Téma jsem zvolil, protože již mám určité zkušenosti s vývojem webových aplikací a chtěl jsem si je dále rozšířit. Vytvořený kurz obsahuje jednotlivé lekce s jednotlivými sekcemi, které jsou pro přehlednost kurzu ve všech lekcích stejné jen s jiným obsahem. Obsahuje sekci pro naučení se nových slovíček, procvičování překladu slovíček jak z angličtiny do češtiny, tak i z češtiny do angličtiny, dále sekci pro naučení se nové gramatiky, procvičení gramatiky z lekce a na závěr test z lekce. Aplikace eviduje jednotlivé statistiky uživatelů v lekcích a souhrnné statistiky každé lekce.

Práce je rozdělena na část praktickou a teoretickou. V teoretické části se nejdříve zabývám historickými zákonitostmi, vývojem aplikačního softwaru a řešenými. Ty tvoří největší část teorie a jsou popsány systémy Duolingo.cz, Onlinejazyky.cz, Englishme.cz, Angličtina-bez-biflování.cz. U jednotlivých systémů popisují nejdříve průběh registrace a prostředí webové aplikace. A poté následuje shrnutí v podobě kladů a záporů. Na závěr teoretické části samozřejmě popisují srovnání s vlastní aplikací a je uvedena přehledná tabulka klíčových vlastností všech systémů.

V praktické části následuje popis použitých technologií. Popisují XAMPP a Apache pro vytvoření webového serveru. Dále jsou uvedeny základní informace o programovacím jazyku PHP, který jsem si vybral pro tvorbu vlastní aplikace, kvůli jeho rozšířenosti a možnostem, které nabízí. Následuje databázový systém MySQL, který byl také použit pro aplikace a uchovává všechny data aplikace. Také jsou zmíněny technologie framework Nette, CSS, JavaScript a vývojové prostředí Netbeans.

Po popisu technologií je popsána vlastní aplikace. A to nejdříve uživatelské role, registrace, přihlášení, prostředí, databázový model s popisem jednotlivých tabulek. Dále také UML Use case diagram, Activity diagram a adresářová struktura aplikace.

1 TEORETICKÁ ČÁST

1.1 Online výukové kurzy

Informační technologie jsou jedny z nejvíce a nejrychleji se rozvíjejících. Ale rychle se rozvíjí i ostatní technologie, což potrhává důležitost anglického jazyka, ve kterém je napsáno nejvíce návodů, dokumentací a doporučení k těmto technologiím.

Mnoho starších generací anglicky moc dobře neumí a nedokáží si najít cenné informace, které v mateřském jazyce nejsou dostupné. Dále také vznikají jazykové bariéry při cestě do zahraničí, a proto se rozhodneme radši zůstat v rodné zemi.

Z těchto důvodů vznikají výukové kurzy, které se zaměřují na výuku anglického jazyka pomocí například odstupňovaných úrovní. Cílem kurzů je lehce a rychle naučit kohokoliv základním dovednostem anglického jazyka na úrovni, kdy se nebudou muset bát, že cesta do zahraničí pro ně bude stresující. Vznikají i přenosné verze, kdy si student zapne aplikaci s kurzem na svém mobilním zařízení a může se tak učit na cestách mnohdy i off-line.

1.2 Historické zákonitosti

Už od historie se lidé rádi učí nové věci a u jazyků tomu není jinak. Například potřeba vojáků za druhé světové války dorozumět se v cizí zemi, čtení cizojazyčné literatury nebo jen chuť cestovat do jinak mluvících zemí. U nás byla v historii výuka cizích jazyků ovlivněna například komunistickým režimem, kdy se používaly spíše metody učení se holé gramatiky, překladu a používali se cenzurované učebnice založené na politické situaci. Poté se čím dál více uplatňuje učení hrou, kdy je snaha zaměstnat více lidských smyslů a učení obsahuje více obrázků a komunikace je hlavní složka výuky. [12]

S příchodem informačních technologií se výuka přesunula na obrazovky a umožnila interaktivnější učebnice a materiály. Příkladem může být přehrávání videa či opakování si sluchové nahrávky za účelem co nejlépe porozumět. Dnes se staly moderními tzv. multiplatformní výukové kurzy, se snahou je co nejvíce rozšířit a učinit je tak dostupnějšími třeba na cestách jen s mobilním telefonem. [12]

1.3 Vývoj aplikačního SW

Pod pojmem aplikační SW si můžeme představit programové vybavení zařízení, které je určeno pro interakci s uživatelem, který jej může ovládat. Jedno z nejvíce rozšířených odvětví jsou

webové aplikace používající pro uchování dat většinou relační databáze a jsou napsané v některém programovacím jazyce, příkladem může být známé PHP.

1.3.1 Vývoj webových aplikací

Webové aplikace a internet jsou součástí našeho každodenního života a lidé je používají takřka každý den. Dříve tomu však tak nebylo a tyto technologie prošly velkými změnami. Vzniku internetu se připisuje rok 1968, avšak webová služba vznikla až roku 1989. Toho roku Tim Berners-Lee specifikoval systém hypertextů a následně na to byl vytvořen první webový prohlížeč s názvem WorldWideWeb, který pak pro přehlednost přejmenoval na Nexus. Byl také vyvinut první webový server a jazyk HTML. [1][10]

Tento značkovací jazyk určoval pomocí značek strukturu a vzhled dokumentu. Ve verzi 4.0 byla přidána podpora CSS pro podrobnější stylizování stránky. V počátcích byly velké rozdíly ve webových prohlížečích, což mělo za následek rozlišné zobrazování stránek. Ty se v dnešní době už co nejvíce sjednocují a vývoj pro více prohlížečů už není tak náročný. S narůstajícím počtem přenosných zařízení se objevuje nový problém, a to zobrazení na menších obrazovkách. [1][10]

V roce 1994 vytvořil Rasmus Lerdorf PHP a o rok později zveřejnil první verzi s licencí Open Source. V dalších verzích se připojila podpora MySQL a objektově orientovaným přístupem. Roku 1999 se objevují první skriptovací jazyky pro změnu stránky na straně klienta, dále pak také technologie flash, která dnes spíše pro její malou bezpečnost mizí. [14]

1.3.2 Vývoj databází

S vývojem webových aplikací přišla potřeba uchovávat informace nejen ve statické podobě ve webových souborech, ale i v dalším úložišti, kde by byly srozumitelně seříděné a rychle dostupné, a proto vznikly první databáze. S uchováním dat se lidstvo stýká už od pradávna, ať už v podobě hliněných desek ve 12. století před naším letopočtem nebo v knihovnách ve starověkém Římě, kde byla snaha data nejen shromažďovat, ale i třídít. V 18. století se objevují první kartotéky, které se databázím podobají více. V roce 1890 vynalezl Herman Hollerith první stroj na děrné štítky, který umožňoval elektromechanické ukládání dat. Další fází bylo ukládání dat na magnetické pásky a disky podporující přímé čtení dat. [15]

V roce 1971 byl definován první systém řízení báze dat, který obsahoval už datový model, entity, integritu dat a další pojmy, jak je známe dnes. Poté se začaly vyvíjet databázové systémy

na základě síťového a hierarchického modelu. Ty měly však hlavní nedostatky při modelování reálného světa, a tak v roce 1970 navrhl zaměstnanec IBM Edgar Frank základ relačního modelu, který obsahoval dnes známé operace pro zpracování dat jako projekci, selekci, sjednocení, spojení či kartézský součin. Roku 1987 byl přijat SQL jazyk jako standard a také vyvinut objektově-relační model se jménem Postgres. Dalším krokem po roce 2000 byl nástup NoSQL systémů, které proslavila hlavně firma Google se svým systémem BigTable pro uchování a zpracování velkého objemu dat. Tyto systémy jsou sloupcově orientované na rozdíl od dosavadních systémů, které byly spíše řádkově orientované. [15]

1.3.3 Vývoj programovacích jazyků

Vývoj programovacích jazyků byl úzce spjat s tím, jak se vyvíjely počítače. U zrodu programovacích jazyků byly hlavně ženy. Základy položila autorka jednoho z prvních programovacích jazyků vědkyně a matematicka Grace Hopperová. Tato žena spolu s dalšími v té době vytvářela nové programy nejdříve na počítač Mark I a pak i ENIAC, pro které vytvářela i manuály. A protože počítače nebyly ještě univerzální, bylo třeba počítač přestavět při každém novém programu. [16]

Dále byl vyvinut assembler jako jazyk symbolických adres, který je i dnes někdy nenahraditelný. Jeden z prvních vyšších programovacích jazyků byl Fortran zveřejněný roku 1958 se zaměřením na vědecké výpočty. Na rozdíl od assembleru byl přenositelný a desetkrát rychlejší. Následovaly jazyky Algol a Cobol, které se snažily odstranit nedostatky Fortranu. [17]

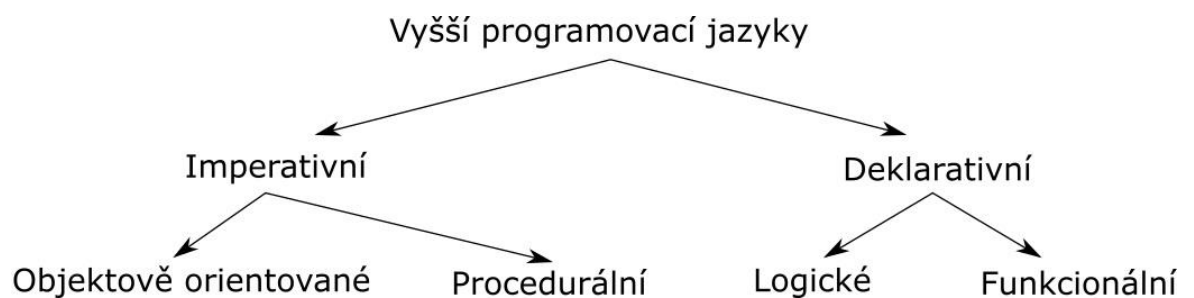
V průběhu sedmdesátých let se staly populárními modulární a objektově orientované jazyky. Objektové jazyky obsahují objekt jako datovou strukturu, což se ukázalo velmi efektivní, a tak vznikly jazyky C# a Java. Tyto a další se používají dodnes. [17]

1.3.4 Druhy programovacích jazyků

Existuje více rozdělení programovacích jazyků. Nejznámější je na vyšší a nižší. Nižší jsou ty, kde musíme psát přímo instrukce pro procesor. Výhodou tohoto je, že můžeme obsluhovat i funkce, které vyšší jazyky neumí. Na druhou stranu je takový kód velmi složitý, nepřehledný a není přenositelný na jiný procesor, než byl určen. Nižší programovací jazyk je například Assembler a strojový kód. Druhým typem, jak už bylo zmíněno, jsou vyšší programovací jazyky. Ty jsou přesný opak. Jsou jednodušší, mají logickou strukturu a nejsou závislé na

procesoru. Výsledný program je třeba ještě přeložit kompilátorem na strojový kód. Takovéto jazyky jsou například Fortran, Basic, Pascal, Java a C++. [18]

Dalším typem může být rozdělení na logické, funkcionální a imperativní. U funkcionálních programátor popisuje úlohy pomocí funkcí, což se může stát trochu nepřehledné. Imperativní jazyky popisují program pomocí příkazů jdoucích po sobě. Tento postup připomíná recepty nebo návody z běžného života. Další podkategorií imperativních jazyků může být již výše zmíněný objektově orientovaný jazyk. Ten se skládá z objektů umožňující zapouzdření a dědičnost. Posledním rozdělením, které bych chtěl zmínit, je dělení na interpretované a kompilované. Interpretované se překládají během vykonávání programu, což má za následek pomalejší vykonávání kódu. Kompilované jazyky se nejdříve přeloží a až poté jsou vykonány. To umožňuje rychlejší vykonávání. [18]



Obrázek 1: Druhy programovacích jazyků

1.4 Problematika systému

Cílem je vytvořit systém, který je snadno a rychle ovladatelný s požadavkem na přidávání dalších částí a modulů. Systém musí být uživatelsky snadno srozumitelný i bez hlubších znalostí informatiky, které nejsou pro studium pomocí kurzu potřebné. Systém by měl být správně strukturovaný, tudíž je potřeba rozdělit kurz na jednotlivé lekce obsahující podsekcce. Dále musí být dostupný online a všechnen obsah uchovaný v databázi. V databázi je potřeba také uchovat výsledky uživatelů a ty pak srozumitelně prezentovat uživatelům v rámci jejich účtů.

1.5 Rešerše podobných systémů

V následující části postupně popíšu a srovnám čtyři online kurzy angličtiny, které jsem vybral. Budu popisovat základní informace o aplikaci, průběh registrace, prostředí, zápory a klady.

1.6 Duolingo.cz

Prvním porovnávaným systémem je výukový online kurz od společnosti Duolingo, který se nachází na adrese duolingo.cz a má dostupné aplikace na operační systémy Android, iOS a Windows. Zdarma je verze s reklamou, jinak se platí zhruba deset dolarů za měsíc. Dále je ještě k dispozici verze pro školy. Zaměřuje se hlavně na angličtinu, ale nabízí i dalších 27 jazyků k učení. [21]

1.6.1 Registrace

První je nutná registrace, ta probíhá nejdříve volbou jazyka, poté je vybrán minimální denní cíl od pěti po 20 minut denně. Na dodržení upozorňuje mobilní aplikace v oznámení na liště mobilního telefonu. Dále se vybírá, zdali chceme zahájit rozřazovací test, či začít úplně od základů. Po zvládnutí prvního testu se požaduje registrace, kdy vkládáme věk a e-mailovou adresu. [21]

1.6.2 Prostředí

Celý kurz je rozdělen do několika úrovní pojmenovaných podle gramatiky, či tématu slovíček. Úrovně se dají přeskočit pomocí testů, kdy se ověří, že již máte danou problematiku nastudovanou a nezdržuje vás to tak v postupu. Po kliknutí na jednu z úrovní se zobrazí jedna až deset lekcí podle výběru a následované popisem gramatiky probírané v lekci. Zde je trochu škoda, že se gramatika nezobrazuje i v mobilní aplikaci, kde by pro ni určitě bylo také využití i místo. [21]

Testování probíhá formou psaní překladu či kliknutí na obrázek slovíčka. Zde bych chtěl trochu vytknout, že většinu učení slovíček strávíte tím, že jen klikáte na obrázky, což vás nenaučí, jak dané slovíčko napsat. Na stranu druhou je to pěkné učení hrou a ve verzi pro školy určitě dobře využitelné. Pokud dané slovíčko nebo větu splete, objeví se na konci lekce a opakujete ho, dokud není správně. Dále je tu k dispozici procvičování poslechu, mluvení, psaní nebo čtení.

Na úvodní obrazovce se zobrazuje postup v kurzu v podobě procent a dosažených úspěchů. Zde mi trochu chybí souhrnné statistiky a historie testů, ale jinak celkově hodnotím kurz jako velmi podařený. [21]

1.6.3 Klady

- Kurz je zdarma
- Přehledně rozčleněné úrovně
- Příjemný design
- Aplikace na všechny platformy
- Obsahuje poslech a možnost mluvení [21]

1.6.4 Zápory

- Jen do úrovně B2
- Nemožnost vybrat si úroveň
- Gramatika se na mobilní aplikaci nezobrazuje
- Žádné statistiky nebo historie [21]

1.7 Onlinejazyky.cz

Dalším srovnávaným online kurzem angličtiny je kurz na adrese onlinejazyky.cz od společnosti Online learning s.r.o. Tato společnost nabízí výuku anglického jazyka, němčiny, francouzštiny, španělštiny a další. Bohužel kurz nemá k dispozici mobilní aplikaci na žádný z operačních systémů, ale má aspoň plně responzivní design, takže i na mobilu je kurz přehledný. [22]

1.7.1 Registrace

Na začátku máme několik možností, jak spustit kurz. První je přihlášení pomocí účtu vašeho Facebooku nebo registrací pomocí e-mailové adresy. Další z možností je vložení pouze e-mailové adresy, kdy se jen spustí jedna lekce na vyzkoušení. Poté je nutné si zaplatit jednu úroveň jazyka, kterou si vybereme v e-shopu, za necelých 2500 korun. Zde bych viděl první nevýhodu, protože musíme platit za každou úroveň zvlášť na rozdíl od konkurence, která nabízí

celou výuku zdarma. Na druhou stranu je zde záruka toho, že lekce budou opravdu kvalitní. Dále jsou nabízeny licence pro firmy a školy, kdy přihlášení probíhá pomocí údajů poskytnutých firmou nebo školou. [22]

1.7.2 Prostředí

Po přihlášení do zkušebního kurzu následuje načtení první stránky s úvodním videem, kde je popsán základní princip kurzu. Celou lekci doprovází téma jídlo a stolování. Postupně jsou ukázány, jaké druhy cvičení jsou uplatňovány. Nejdříve vás lekce naučí pár základních slovíček, u kterých zobrazí obrázky a dá možnost si poslechnout výslovnost. Další metody jsou vybírání správnosti odpovědi, doplňování slov do věty při současném poslechu, křížovka, fráze s poslechem, seřazení vět v rozhovoru a mnoho dalších. Další chybou, kterou jsem objevil, bylo, že se nezobrazovalo slibované video po každém cvičení. To mělo dát zpětnou vazbu, co se vám v lekci nepovedlo. [22]

V celé úvodní lekci se zobrazují videa, které vás postupně naučí, jak co dělat nebo ukáží, co všechno je v kurzu k dispozici. V menu je možnost napsat zprávu lektorovi, když něčemu nerozumíte nebo si připlatit video konverzace. Také je možné zobrazit si nástěnku s úspěchy a s postupem v kurzu. Také si lze v nastavení zapnout či vypnout opravu interpunkce a opravování velkých písmen, což je dobré, pokud se chcete zabývat pouze výukou daného jazyka. [22]

1.7.3 Klady

- Povedené videa obsahující průvodce kurzem
- Vysoká úroveň výuky
- Mnoho možností procvičování
- Videohovory s učitelem [22]

1.7.4 Zápory

- Jen jedna lekce zdarma
- Vyšší cena jedné úrovně jazyka
- Nefunkční vyhodnocovací videa

- Nemá mobilní aplikaci [22]

1.8 Englishme.cz

Jako třetí bych chtěl srovnat kurz na adrese englishme.cz od společnosti Vitware s.r.o. Tato aplikace se zabývá anglickým jazykem a další už nenabízí. Opět není k dispozici mobilní aplikace, ale jen webové rozhraní. Trochu negativně bych hodnotil design aplikace, který je možná až příliš strohý. [23]

1.8.1 Registrace

Aplikace nabízí sedmidenní zkušební období, které není nijak obsahově omezeno. Poté je možné si zakoupit předplatné na tři, šest nebo dvanáct měsíců. Předplatné je nabízeno od 290 až po 790 korun. Kladně bych zhodnotil hromadné slevy například pro školy nebo firmy. Registrace probíhá vložím e-mailové adresy a hesla. Po zadání údajů aplikace vyzve výběru úrovně angličtiny, kde první úroveň je spíše ve smyslu seznámení se s jazykem, pak už je možné studovat. [23]

1.8.2 Prostředí

Celé prostředí úvodní obrazovky je rozděleno na tři části, a to na slovíčka, gramatiku a obecné. V obecné části je například překlad, hláskování nebo psaní číslic. V kurzu je možné vytvořit skupinu nebo se do některé přidat. Skupiny jsou nejvhodnější, když chcete kurz používat ve škole, pak je možné zadat úkol a ten se zobrazí všem patřícím do skupiny. V nastavení se dá určit, zda preferujeme britskou, či americkou angličtinu, nastavit si důkladnost opravy slovíček, kdy při nejpřísnějším nastavení budete muset slovíčko opakovat, dokud není absolutně správně. [23]

Při procvičování slovíček se zobrazuje nejdříve anglický popis a příklady využití ve větách, to je velmi povedené. Při dokončení procvičování se zobrazí výsledek nejen v procentech, ale i v přehledných grafech. Jeden takový graf například ukazuje, jaký je poměr mezi slovíčky, která dělaly problémy, velké problémy nebo byly úplně bez chyby. V celém kurzu je bohužel velmi málo druhů cvičení, takže může vyplňování rychle omrzet. [23]

1.8.3 Klady

- Ve zkušebním období je kurz bez omezení
- Kvalitní testy
- Skupiny vhodné hlavně pro školy
- Nižší cena
- Přehledné statistiky [23]

1.8.4 Zápory

- Málo druhů testů
- Strohý design
- Nemá mobilní aplikaci
- Chybí rozřazovací test [23]

1.9 Angličtina-bez-biflovani.cz

Posledním porovnávaným kurzem je Angličtina bez biflování od společnosti Euvit, s.r.o. Společnost nabízí tři druhy kurzů, a to základní angličtinu za pět tisíc korun, obchodní angličtinu za šest tisíc korun a angličtinu pro děti za tři a půl tisíce korun. Cena mi přijde dost vysoká oproti konkurenci, která nabízí mnohem více obsahu a možností studia. Aplikace nabízí pouze jazyk angličtina a nedisponuje mobilní aplikací. [24]

1.9.1 Registrace

Registrace má pouze jednu možnost, a to přes e-mailovou adresu, na kterou jsou zaslány přihlašovací údaje včetně hesla. To si myslím, že není příliš bezpečná metoda. Před zadáním e-mailové adresy se ještě zobrazí video s dialogem vysvětlující princip kurzu. Tím je registrace dokončena a kurz spustí první lekci popisující vše ještě jednou, ale tentokrát už v anglickém jazyce. [24]

1.9.2 Prostředí

Na levé straně aplikace se nachází menu s jednotlivými cvičeními. Jedno z osmnácti cvičení vždy obsahuje pouze jedno téma a nejsou zde žádné odstupňované lekce, jak tomu bývá u jiných jazykových kurzů. Cvičení nabízí pouze fráze, které jsou systémem čteny a studentem poslouchány a další druhy cvičení už nenabízí. Což je na jednu stranu originální metoda učení, ale za peníze, které požadují je to celkem málo. [24]

Jednotlivé cvičení vždy postupně prochází frází s hlasem, který je čte. Slova ve frázích se postupně zvýrazňují podle toho, jak je systém čte. Pod anglickou frází se také zobrazuje doslovný český překlad, pro případ že některé slovíčko student nezná. Doslovný překlad je možné v každé lekci vypnout. Lze také nastavit zvuky na pozadí mluveného slova, což pomáhá vžít se do reálné situace. Dále je možné zvolit rychlost přehrávání hlasu nebo ho zcela změnit. Prostředí nabízí upozornění na e-mailovou adresu, které se dá nastavit na určitou hodinu a intenzitu. Na hlavní obrazovce aplikace se nachází statistický přehled, ukazující počet pokusů a hodin strávených nad učením. [24]

1.9.3 Klady

- Video průvodce
- Široké nastavení poslechu a cvičení
- Zasílá upozornění na e-mailovou adresu
- Originální metoda učení
- Sedmidenní verze bez omezení [24]

1.9.4 Zápory

- Nemá mobilní aplikaci
- Nelze psát, obsahuje jen čtení frází
- Méně obsahu za vyšší cenu
- Málo druhů cvičení [24]

1.10 Ohodnocení rešerší

Postupně byly srovnány čtyři online kurzy angličtina a to Duolingo.cz, OnlineJazyky.cz, Englishme.cz a Angličtina-bez-biflování.cz. U všech těchto kurzů jsem se zaměřil hlavně na základní popis, uživatelskou registraci a prostředí. To bylo u každé aplikace takřka originální a vždy se objevilo něco nového. Záměrně jsem vybral co nejvíce rozdílné kurzy, kdy každý nabízí něco dalšího. Duoligo.cz má originální skládání slov do vět na mnoho způsobu. Je to takové odlehčení, když neznáme všechna slovíčka. Onlinejazyky.cz mají originální doplňování například slov do křížovky, rozhodování, jestli je tvrzení správně nebo občas vloží zajímavost o státu, ze kterého je jazyk. Angličtina-bez-biflování.cz využívá metodu poslechu a čtení frází, které říkají Imitum.

Všechny aplikace až na Duolingo.cz nemají mobilní aplikaci. Kompenzují to aspoň responzivním designem. Nejvíce povedeně bych hodnotil Jazykyonline.cz a Duolingo.cz a poté englishme.cz. Jako nejméně povedené bych hodnotil kurz Angličtina-bez-biflování.cz, která vůbec neobsahovala psané cvičení.

1.11 Srovnání s vlastní aplikací

Snahou bylo vytvořit vlastní aplikaci, která se vyrovná ostatním porovnávaným kurzům. Vlastní aplikace je oproti jiným zcela zdarma, obsahuje uživatelské a celkové statistiky. Má různé druhy cvičení v podobě psaní a doplňování slov do vět, testy na konci lekcí a v každé lekci je podrobný popis gramatiky, která je využita ve cvičeních a testu. Jako ostatní má responzivní design a nemá mobilní aplikaci. Dále oproti jiným aplikacím neobsahuje poslech nebo mluvení.

1.12 Srovnání v tabulce

Tabulka 1: Tabulka srovnání jednotlivých kurzů

	Duolingo.cz	Onlinejazyky.cz	Englishme.cz	Anglictina-bez-biflovani.cz	Vlastní aplikace
Plná verze zdarma	ANO	NE	NE	NE	ANO
Zkušební verze	ANO	ANO	ANO	ANO	ANO
Mobilní aplikace	ANO	NE	NE	NE	NE
Responzivní design	ANO	ANO	ANO	ANO	ANO
Volba úrovně jazyka	NE	ANO	ANO	NE	NE
Odstupňované lekce	ANO	ANO	NE	NE	ANO
Uživatelské statistiky	NE	ANO	ANO	ANO	ANO
Mluvení	ANO	ANO	NE	ANO	NE
Čtení	ANO	ANO	ANO	ANO	ANO
Psaní	ANO	ANO	ANO	NE	ANO
Poslech	ANO	ANO	ANO	ANO	NE
Více druhů cvičení	ANO	ANO	NE	NE	ANO
Obrázky u slovíček	ANO	ANO	ANO	NE	ANO
Doplňování gramatiky	ANO	ANO	ANO	NE	ANO
Popis gramatiky	ANO	ANO	NE	NE	ANO
Překlady slovíček	ANO	ANO	ANO	NE	ANO
Celkový výsledek	14/16	14/16	10/16	6/16	12/16

2 PRAKTICKÁ ČÁST

V první části jsou popsány použité technologie. Druhá část obsahuje popis vlastní aplikace, kde je probrán návrh systému, registrace, přihlášení, uživatelské role a celé prostředí aplikace. Dále také popis datového modelu, UML use case diagramu, UML aktivity diagramu a v závěru je pak popis adresářové struktury projektu a instalační příručka.

2.1 Popis použitých technologií

Následující kapitoly popisují technologie využité přímo pro aplikaci. Xamp a Apache pro zprovoznění webového serveru, na kterém byla práce vytvářena. Dále popis skriptovacího jazyka PHP, databáze MySQL, Nette framework pro webové aplikace, CSS pro stylování, a nakonec JavaScript a vývojové prostředí Netbeans, které bylo použito na vývoj.

2.1.1 XAMPP a Apache

Aplikace anglického online kurzu byla vytvářena na operačním systému Microsoft Windows a byl použit snadno ovladatelný nástroj XAMPP. Je to program určený pro vývoj aplikací v jazyce PHP s podporou databáze. Obsahuje předpřipravený webový server Apache, prostředí pro práci s databází MySQL a další nástroje pro vývoj. Nástroj je ke stažení zcela zdarma ze stránek projektu. Podporuje i platformy Linux a OS X. [1]

Apache je webový server s otevřeným kódem, určený pro platformy Microsoft Windows, Linux a BSD. Na tomto nástroji běží většina webových stránek. Historie Apache začíná v roce 1993 na univerzitě v Illinois, kde získal jméno NCSA a později byl přejmenován. Jeho vývoj byl v roce 1998 ukončen. Správci webových serverů Bria Behlendorf a Cliff Skolnick založili e-mailovou konferenci pro shromažďování úprav a záplat. První veřejná verze byla vypuštěna v roce 1995 a poté založena vývojářská skupina Apache Group, která vydržela dodnes. Apache je dnes součástí až 73 procent webových serverů. [1][2]

Pro instalaci nástroje XAMPP s Apache pro platformu Microsoft Windows je třeba stáhnout balíček na adrese <https://www.apachefriends.org/download.html>, kde si vybereme požadovanou verzi nástroje obsahující požadovanou verzi PHP jazyka. Instalaci balíčku s velikostí zhruba 120 megabajt provedeme poklepaním na balíček a postupujeme podle pokynů průvodce. V první fázi se nás zeptá, co požadujeme nainstalovat a poté na cílové umístění. Pokud necháme výchozí nastavení nainstaluje se do složky xampp na systémový disk.

Spuštění musí proběhnout s právy správce a poprvé je třeba povolit upozornění firewallu na komunikaci. Pro start služeb Apache a MySQL je třeba je zapnout v konzoli nástroje. Poté už by měla být dostupná stránka serveru na adrese <http://localhost>, kterou zadáme do webového prohlížeče. Pokud chceme zobrazit vlastní stránku, smažeme všechny soubory a složky v umístění `/xampp/htdocs` a vložíme vlastní soubor s názvem `index.php`. Pro přístup k souborům z jiného počítače v síti stačí zadat adresu IP patřící počítači, na kterém běží webový server. Adresu opět zadáváme do webového prohlížeče. Vývoj tak lze jednoduše testovat i na mobilních zařízeních. [1]

Pro instalaci na platformě Linux je třeba stáhnout archiv ve formátu tar ze stejné adresy jako v předešlé instalaci. Instalaci lze provést příkazem `tar` do adresáře `/opt`, kde se vytvoří adresář `lampp` a systém spustit příkazem `/opt/lampp/lampp start`. Pro ověření funkčnosti lze napsat do adresářového řádku prohlížeče adresu <http://localhost>. Zobrazená stránka se nalézá v adresáři `/opt/lampp/htdocs/`, kam lze také vkládat vlastní soubory. [3]

2.1.2 PHP

PHP je nejrozšířenější skriptovací jazyk pro vytváření dynamického webu. V tomto jazyce je napsáno přes osmdesát procent webových aplikací a jeho obliba se stále zvyšuje. Jeho konkurentem je ASP.NET, který tvoří sedmnáct procent webových stránek. Jazyk je oblíbený hlavně proto, že v něm lze napsat takřka cokoliv. Můžou to být katalogy, redakční systémy, internetové obchody, diskuzní fóra, databázové klienty a mnoho dalšího. Jazyk je lehce pochopitelný a má jednoduchou syntaxi. [1][4]

Jazyk vznikl roku 1994 a jeho tvůrcem je Rasmus Lerdorf a původně to měl být systém, který bude počítat přístup na jeho stránky. Prvotní verze tohoto jazyka byla napsána v PERLu a pojmenována Personal Home Page. Tato verze nebyla však dostatečně rychlá, a tak bylo PHP přepsáno do jazyka C. O rok později byla vydána druhá verze s názvem PHP/FI, která spolupracovala s dalším projektem autora, a to s Form Interpreterem. To umožňovalo práci s formuláři a databázovým serverem. V roce 1998 vyšla třetí verze, kterou napsal s kolegy Zeevem Suraskim, Andim Gutmansem a jazyk přejmenovali na Hypertext Preprocessor. Další verze přišla na svět v roce 2000 s podporou HTTP sessions. O tři roky později přišla verze 5 s podporou OOP. Nejnovější verze v současnosti je 7.2.3, která má 64 bitovou podporou. [1][4]

V klasické PHP struktuře se kód vkládá přímo mezi HTML elementy a dokument má příponu tečka php. Dnešní frameworky umožňují oddělit PHP kód od HTML elementů čímž se projekt stává přehlednější a více rozšiřitelnější. Tento systém bude vysvětlen v kapitolách dále. [3]

PHP obsahuje osm základním datových typů jako v jiných programovacích jazycích, ovšem není potřeba u proměnné typ dopředu definovat, protože PHP ho automaticky rozpozná podle obsahu. [3]

Datové typy:

- Float pro hodnoty s plovoucí desetinou čárkou
- Boolean pro hodnoty pravda či nepravda.
- Integer pro celočíselné záporné a kladné hodnoty.
- String pro řetězce.
- Objekt pro objektovou proměnnou.
- NULL pro datový typ bez hodnoty.
- Array pro vytvoření pole jakéhokoliv typu.
- Resources pro zdroje. [3]

Pro identifikaci PHP kódu se používají značky menší, otazník a větší, otazník uvedené níže na obrázku. Základní výpis se provádí klíčovým slovem echo následovaným uvozovkami s textem pro výpis. Všechny příkazy končí středníkem. [3]

```
1 <?php
2 echo "Nový projekt.";
3 ?>
```

Obrázek 2: Ukázka základního výpisu textu v PHP

Na následujícím obrázku je znázorněno, jak vytvářet proměnné, pole, cyklu, funkce a větvení. Proměnná se vytvoří znakem dolaru a názvem proměnné. Pokud chceme vytvořit proměnnou s číslem stačí ho zapsat za rovná se. Pokud chceme textovou, zapíšeme ho do dvojitých uvozovek a pokud chceme zobrazit ve výpisu i uvozovky, je třeba celý text ohraničit jednoduchýma uvozovkami. Pro vytváření se používá klasická proměnná a za rovná se zapíšeme klíčové slovo array, poté následují závorky. V závorkách přiřadíme klíčům jejich hodnoty pomocí rovná se a větší. Pro výpis hodnoty prvku pole zadáme jméno pole a do

hranatých závorek jméno klíče v poli, jak je ukázáno na devatenáctém řádku v následujícím obrázku. Komentář začíná dvěma lomítky, jak je znázorněno na řádku pět.

Pokud chceme vytvořit funkci stačí zadat klíčové slovo function, název funkce, závorky a parametry. Volání probíhá přes jméno funkce s kulatými závorkami obsahující nastavené parametry funkce. V PHP jako v jiných jazycích existují cykly pro daný počet opakování kódu. Cyklus for se tvoří klíčovým slovem a závorkami, které obsahují inicializaci proměnné, podmínky a inkrementaci proměnné. Cyklus for na řádku šest se provede pětkrát. Obsahuje také podmínku, která udává že výpis bude proveden, jen pokud je proměnná s názvem číslo menší, jak číslo pět.

```
1 <?php
2 $cislo = 10;
3 $text = "Text v proměnné";
4
5 //Cyklus for
6 for($i = 1; $i <6; $i++){
7 if($cislo < 5){
8 echo $i;
9 }
10 }
11
12 //Volání funkce
13 tvorbaPole("Vondráček");
14
15 function tvorbaPole($prijmeni){
16 $pole = array(
17 "Jméno" => "Filip",
18 "Příjmení" => $prijmeni);
19 echo $pole["Příjmení"];
20 }
21 ?>
```

Obrázek 3: Ukázka základů v PHP

PHP podporuje objektově orientované programování. OOP obsahuje objekty, které komunikují prostřednictvím metod a obsahují zapouzdřená data v podobě privátních proměnných. Je to jednoduchý způsob, jak strukturovat data, aby byl kód lehce rozšiřitelný. Můžeme mít například objekt obdélník, který má soukromé vlastnosti šířka a výška. Tyto vlastnosti bude možné získat a nastavit jen prostřednictvím funkcí. Dále může obsahovat funkci pro výpočet obsahu. Objekt v PHP se vytváří pomocí klíčového slova class, názvu třídy a těla třídy ohraničeného složenými závorkami. Pro vytvoření objektu je třeba definovat konstruktor, ten se tvoří slovem `__construct` a v závorce obsahuje parametry pro inicializaci proměnných. [3]

2.1.3 MySQL

MySQL je otevřený a vícevláknový systém pro relační databáze vyvinutý švédskou společností TcX. Je uvolněn zdarma k vyžívání, a proto se stal velmi oblíbeným u nízkonákladových projektů. Využívá strukturovaný dotazovací jazyk SQL pro manipulaci s daty, vytváření nebo zobrazování dat. Je to tedy systém pro správu tabulek, jejich sloupců a řádků. Umožňuje také správu databáze, a tedy určit kdo k ní má přístup a jaké práva. [5]

Jak už bylo řečeno MySQL je systém pro relační databáze. Takováto databáze se skládá z tabulek, které jsou na sebe navzájem propojeny pomocí sloupců jiných tabulek. Tento systém databází vytvořil E. F. Coddem v sedmdesátých letech, kdy byl velmi oblíbený hierarchický model a učinil tak velký zvrat ve vytváření databází a uchovávání dat. [5]

Tabulka může mít svůj primární klíč pro jednoznačnou identifikaci záznamu. Každý záznam má pak ve sloupci primárního klíče unikátní hodnotu. Pomocí primárního klíče se dají vytvářet relace mezi tabulkami. Rozlišujeme relace typu jedna ku jedné, jedna ku N, N ku jedné a M ku N. Relace jedna ku jedné spojuje přesně jen dvě tabulky. Je to například čárový kód u knížky, Znamená to, že jedna knížka bude mít jeden čárový kód a jeden kód může patřit k jen jedné knížce. Relace typu jedna ku N je například vztah adresy k osobě. Tedy osoba má pouze jednu adresu, ale na stejné adrese může bydlet více lidí. Relace typu M ku N může být vztah knížka a autor. Knižka může mít více autorů, a přitom autor může napsat více knížek. [7]

Dále bych chtěl popsat datové typy.

Datové typy:

- INT s velikostí čtyři bajty pro celočíselné hodnoty.
- FLOAT s velikostí čtyři bajty pro hodnoty s desetinným číslem.
- DOUBLE s velikostí osm bajtů pro hodnoty s desetinným číslem.
- DATE pro rok, měsíc a den.
- DATETIME pro rok, měsíc, den a čas.
- TIME jen pro čas.
- VARCHAR s velikostí 255 bajtů pro text.
- TEXT pro velmi dlouhé texty.
- BLOB pro data v binární podobě. [5]

Dotazy v MySQL by se dali rozdělit na tři hlavní skupiny, a to na DDL, DML a DCL. DDL se týká vytváření objektů pomocí CREATE, editace existujících objektů s příkazem ALTER a mazání pomocí DROP. Druhá skupina DML má na starosti manipulaci s daty. Patří sem příkazy typu SELECT pro výběr dat, UPDATE pro úpravu dat, INSERT pro vkládání dat a DELETE pro mazání. Třetí skupinou je DCL pro řízení dat. Obsahuje příkazy GRANT pro nastavení práv uživatelů na objekty, naopak REVOKE pro odstranění práv. [6]

Dále bych chtěl popsat jednotlivé příklady dotazů.

```
1 CREATE TABLE zamestnanci (  
2 id_zamestnanec INT(11) UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
3 jmeno VARCHAR(30) NOT NULL,  
4 prijmeni VARCHAR(30) NOT NULL,  
5 mail VARCHAR(50),  
6 cas_registrace TIMESTAMP  
7 );  
8  
9 ALTER TABLE zamestnanci ADD vek INT(3);  
10  
11 DROP TABLE zamestnanci;
```

Obrázek 4: Ukázka příkazů skupiny DDL

První série dotazů patří do skupiny DDL. První příkaz začínající na řádce jedna je pro vytvoření tabulky. Začíná klíčovými slovy CREATE TABLE a následuje jméno tabulky a v závorce názvy jednotlivých sloupců s jejich datovými typy a maximální délkou v závorce. Sloupce se oddělují čárkou a každý dotaz končí středníkem. U prvního sloupce je navíc nastaveno, že se jedná o primární klíč s automatickým zvyšováním hodnoty. U jména a příjmení je nastaveno, že sloupce nesmí být prázdné. V opačném případě budeme upozorněni chybou v dotazu při vkládání hodnot.

Druhý dotaz na řádce devět slouží pro úpravu tabulky. V tomto případě se jedná o přidání nového sloupce do tabulky zamestnanci s názvem věk, typem INTEGER a délkou tři. Dotaz na posledním řádce slouží pro smazání celé tabulky zamestnanci i s daty.

```
1 INSERT INTO zamestnanci (jmeno, prijmeni, mail, vek)  
2 values ('Filip', 'Vondráček', 'st49704@student.upce.cz', 32);  
3  
4 UPDATE zamestnanci set vek = 22 WHERE vek = 32;  
5  
6 SELECT * FROM zamestnanci WHERE jmeno = 'Filip';  
7  
8 DELETE FROM zamestnanci WHERE vek = 22;
```

Obrázek 5: Ukázka příkazů skupiny DML

Druhá série dotazů se týká skupiny DML. První dotaz je pro vkládání dat do tabulky. Začíná klíčovými slovy INSERT INTO a následuje jméno tabulky. V první závorce jsou názvy

sloupců, do kterých vkládáme data a ve druhé závorce ve stejném pořadí jsou hodnoty, které chceme vložit. Na řádce čtyři je dotaz pro úpravu dat v tabulce. Nastavuje se cílová tabulka, sloupce a jejich nové hodnoty a podmínka, podle které se rozhodne, které záznamy se budou upravovat. Pokud nezadáme podmínku, budou upraveny všechny záznamy v tabulce. Dotaz na řádce šest je pro výběr a zobrazení dat z tabulky. Dotaz začíná slovem SELECT, za kterým následují jména sloupců, které chceme vybrat oddělené čárkou. Pokud chceme vybrat všechny sloupce, nemusíme je všechny vypisovat, stačí zadat hvězdičku. Po výčtu sloupců se opět zadává jméno požadované tabulky a na konci už jen stačí zadat podmínku, podle které se bude filtrovat. Poslední dotaz slouží pro mazání dat z tabulky. Zadáváme název a podmínku. V tomto případě se vymažou všechny záznamy, kde zaměstnanec má věk 22.

```
1 CREATE USER uzivatel@localhost IDENTIFIED BY 'heslo';
2 GRANT SELECT, UPDATE, DELETE ON firma.* TO uzivatel@localhost
3
4 REVOKE DELETE ON firma.* FROM uzivatel@localhost;
```

Obrázek 6: Ukázka příkazů skupiny DCL

Poslední zmiňovanou skupinou je DCL, do které také patří dotazy na obrázku výše. První pro vytvoření uživatele se jménem uživatel a heslem heslo. Na řádce dva je přiřazení práv pro databázi firma na všechny tabulky tomuto uživateli. Přidávají se oprávnění na výběr, úpravu a mazání dat. V posledním dotazu je zrušení práv mazání z tabulek v databázi firma.

2.1.4 Nette

Nette je český PHP framework pro snadnější a rychlejší vývoj webových aplikací. Využívá objektově orientované programování, nově PHP 7, vznikl v roce 2004 a v roce 2008 byl uvolněn jako open source projekt. Jeho autorem je český vývojář David Grudl. V tomto frameworku se dají vytvářet malé aplikace s několika stránkami, ale také e-shopy, blogy, redakční systém a mnoho dalšího. Pro vytvoření nového projektu je třeba si stáhnout sandbox, což je základní kostra aplikace, která se postupně rozšiřuje o nové věci. Framework využívá MVP architekturu. Což je koncept využívající tři základní druhy komponent. Mají na starosti řízení, logiku a zobrazování. Díky tomuto je pak projekt snadněji rozšiřitelný a celkově přehlednější. Pro řízení se využívají komponenty označené jako presentery. Je to vrstva mezi daty a zobrazením dat. Presentery získávají data od modelů a předávají je uživatelům k vykreslení na obrazovku. Model slouží pro zpracování dat, různé výpočty, a hlavně obsahuje všechny dotazy do databáze. Model tedy předává data do presenteru, který je zas pošle

komponentě view. Ta má na starosti vykreslení a uspořádání webové stránky. Veškeré HTML prvky se tedy nacházejí právě v komponentě view. V Nette jsou například napsány weby jako ČSFD, GE Money bank nebo Slevomat. [8][9]

2.1.5 HTML

HTML je zkratka HyperText Markup Language, v českém překladu znamená hypertextový značkovací jazyk. Vymyslel ho Tim Berners-Lee a slouží pro tvorbu webových stránek, konkrétně pro tvorbu obsahu stránek. [1][10]

Jeho první verze vznikly na počátku devadesátých let, kdy toho tento jazyk zatím ještě moc neuměl. V roce 1995 vyšla verze 3 a v roce 1997 verze 4 s podporou kaskádových stylů, skriptů, tabulek a multimediálních souborů. V roce 1999 vyšla verze 4.01, kterou nedávno nahradil standard HTML 5. Tato verze přinesla nové elementy article, section, figure a umožňuje přímé přehrávání zvukových a souborů a videí. [1][10]

Jazyk HTML používá k popisu stránky takzvané elementy. Elementy uzavírají svůj obsah do značek, které mohou mít různé atributy, jako například nastavení stylu nebo typu formulářového prvku. Elementy se dají rozdělit na tři základní skupiny, a to na strukturální pro rozložení dokumentu, popisné pro popis obsahu a stylistické pro vzhled. Stylistické se již v novém standardu HTML 5 nedoporučuje používat a popisovat je jen CSS styly. Každý HTML dokument se skládá ze základních elementů. Skládá se z HTML tagu, head tagu pro nastavení cest k CSS souborům, skriptům, metadata a popis stylů. Dále také obsahuje hlavní element body pro obsah stránky. [10]

2.1.6 CSS

Zkratka CSS znamená Cascading Style Sheet neboli kaskádové styly, které vznikly roku 1996. Jsou určeny pro nastavení vzhledu dokumentů psaných ve značkovacím jazyce jako třeba HTML a XML. S příchodem CSS se tak oddělila struktura obsahu od nastavení designu. Styly dokumentů se určují pomocí pravidel, které jsou aplikovány na jednotlivé elementy. Pravidlo je složeno ze selektoru a deklaračního bloku. Selektor může být například h1, tímto řekneme, že se nastavení v deklaračním bloku bude týkat jen nadpisů první úrovně. V deklaračním bloku pak uvádíme jednu nebo více vlastností a jejich hodnoty. Vlastnost může být například color a hodnota red pro nastavení písma na červenou barvu. Styly se uvádí buď přímo k elementu do atributu style nebo do hlavičky dokumentu mezi párové tagy style a poslední možností, jak

nastavit styl, je vložit všechno do samostatného souboru a ten připojit v hlavičce dokumentu. Tím zaručíme dokonalé oddělení a přehlednost struktury od nastavení designu. [1][10]

2.1.7 JavaScript

JavaScript je programovací jazyk, který slouží pro dynamické zpracování kódu na straně uživatele ve webovém prohlížeči. Je to jazyk interpretovaný a rozlišuje velká písmena v zápise kódu. Lze pomocí něho například vytvořit hodiny, hry, validovat data ve formulářích, měnit obsah elementů a mnoho dalšího. Nevýhodou implementace jazyka do webových aplikací je to, že programátor musí počítat s tím, že uživatel může zakázat takovéto skripty a zapříčinit tak nesprávné fungování webu. Skript se dá zapisovat přímo do HTML dokumentu mezi tagy script nebo do externího souboru s příponou js, který je nutno připojit a stránka si obsah sama vloží. [11]

2.1.8 Netbeans

Je to vývojové prostředí pro jazyky Java, PHP, C, C++ a jiné. Projekt Netbeans vznikl v roce 1996 v České republice s původním názvem Xelfi jako studentský projekt. V roce 1998 byl prodán Sun Microsystems. [1]

2.2 Aplikační řešení

V následujících kapitolách bude popsáno aplikační řešení bakalářské práce.

2.2.1 Návrh systému

Vlastní aplikaci jsem nazval English-course, jméno vychází ze zaměření kurzu. Cílem bakalářské práce bylo navrhnout funkční systém, který se bude zabývat výukou anglického jazyka. Umožnit evidenci a přístup uživatelů do systému podle oprávnění. Ukládání dosažených výsledků a vypisování statistik kurzu. Přístup k lekcím na základě výsledků přechozích testů a generování sestav dle volitelně zadaných kritérií. Všechny tyto funkcionality byly v kurzu vytvořeny a dodrženy.

Systém je určen pro studenty anglického jazyka, kteří se chtějí učit online. Je koncipován tak, aby co nejvíce ulehčoval a zkvalitňoval studium. V kurzu jsou použity obrázky pro výuku nových slovíček v lekcí nebo různé doplňování slov či rovnou celých frází, které po zadání

system vyhodnotí. Je umožněno také studium na mobilním telefonu, protože systém je responzivní a dá se tedy pohodlně spustit i kdekoliv na cestě.

2.2.2 Uživatelské role

Pro udržení bezpečnosti aplikace využívá práva jednotlivých uživatelů. Bez práv by každý mohl přistoupit k datům, ke kterým by chtěl a napáchat tak velké škody. Pro jednotlivá práva jsou přiřazeny role a do těch začlenění uživatelé.

Základní role je nepřihlášený uživatel. Takovýto uživatel nemá skoro žádnou pravomoc, může jen přistoupit a prohlížet úvodní stránku, registrovat se nebo pokud má svůj účet, tak se také přihlásit.

Další rolí je registrovaný uživatel. Je to uživatel, který vyplnil registrační formulář a zapsal se tak do databáze aplikace. Tato role je v aplikaci nejpoužívanější, nejdůležitější a týká se všech studentů kurzu. Má právo spustit a vyplňovat jednotlivé lekce. Zobrazovat své výsledky a postup v kurzu. Spustit test s vlastními parametry. Upravit ve svém profilu jméno, příjmení, věk, poznámku a změnit heslo. Dále také zobrazit historii pokusů o zvládnutí jednotlivých testů.

Poslední role v aplikaci je administrátorská role. Má oprávnění jako dvě předešlé role, a navíc může zobrazovat tabulku se všemi uživateli a jejich informacemi. Uživatele může vymazat nebo zobrazit podrobnost o historii vyplňování testů. Dále také zobrazovat souhrnné statistiky a úspěšnost jednotlivých testů.

2.2.3 Registrace a přihlášení

Pro získání role registrovaný s jejími právy je třeba se registrovat. Do registračního formuláře se vstupuje pomocí odkazu registrovat, který je umístěn v hlavičce stránky hned vedle názvu kurzu.

V registračním formuláři je třeba vyplnit e-mailovou adresu, která nesmí již být v databázi, heslo a heslo znovu pro potvrzení. Kvůli bezpečnosti musí být heslo minimálně šest znaků dlouhé, aby nešlo útočníkem lehce odhadnout. Poté stačí jen stisknout tlačítko registrovat a aplikace provede zápis do databáze. Po registraci je možné se přihlásit. K tomu se opět dostaneme pomocí odkazu z hlavičky. Zobrazí se formulář, kde je třeba vyplnit e-mailovou adresu a zvolené heslo. Po stisknutí tlačítka přihlásit je provedena autentizace uživatele.



 **Registrace**

E-mail:

Heslo: Alespoň 6 znaků

Heslo znovu:

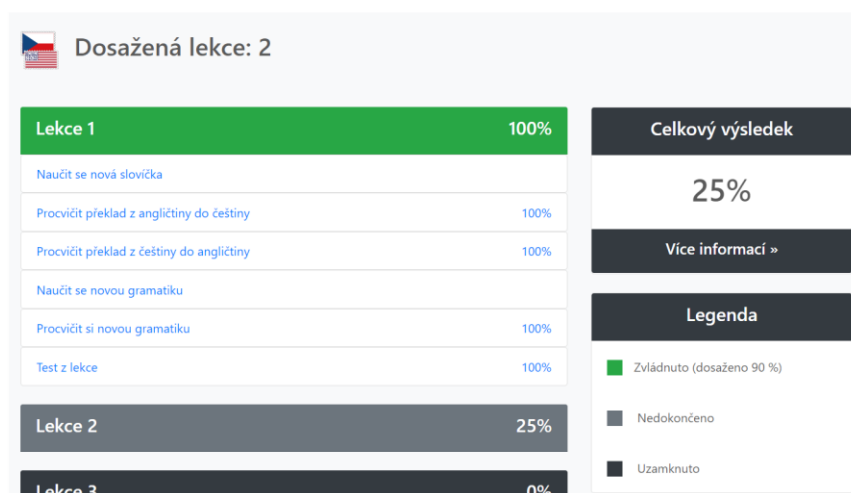
[Již máte svůj účet? Přihlašte se.](#)

© English course 2018, Autor: Filip Vondráček

Obrázek 7: Ukázka registrace

2.2.4 Prostředí

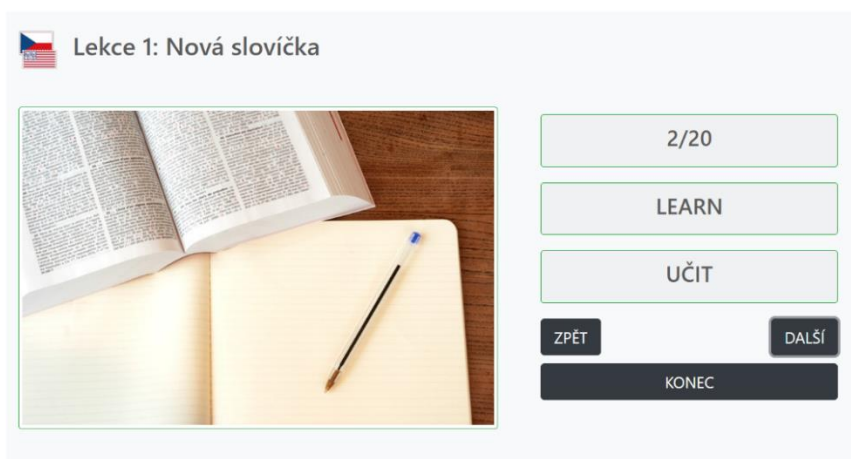
Celá hlavní strana kurzu po přihlášení je rozdělena na část s výčtem jednotlivých lekcí a na část, která zobrazuje celkový výsledek v procentech, legendu a odkaz pro spuštění testu s vlastním nastavením. Při prvním spuštění je odemčená pouze první lekce a pro postup do další je třeba dosáhnout minimálně sedmdesát procent z té předešlé. Kvůli přehlednosti jsou zamčené lekce černé, odemčené jsou šedivé a lekce, které student dokončil aspoň na devadesát procent jsou označeny zeleně. Jednotlivé lekce se skládají z šesti částí. První slouží pro osvojení si nových slovíček, druhá pro překlad slovíček z angličtiny do češtiny, třetí pro překlad slovíček z češtiny do angličtiny, čtvrtá pro naučení nové gramatiky, další pro procvičování gramatiky a poslední pro celkový test z lekce. Slovíčka a gramatika naučená v předešlých lekcích jsou využívána v těch následujících.



Obrázek 8: Ukázka prostředí aplikace

2.2.5 Nová slovíčka

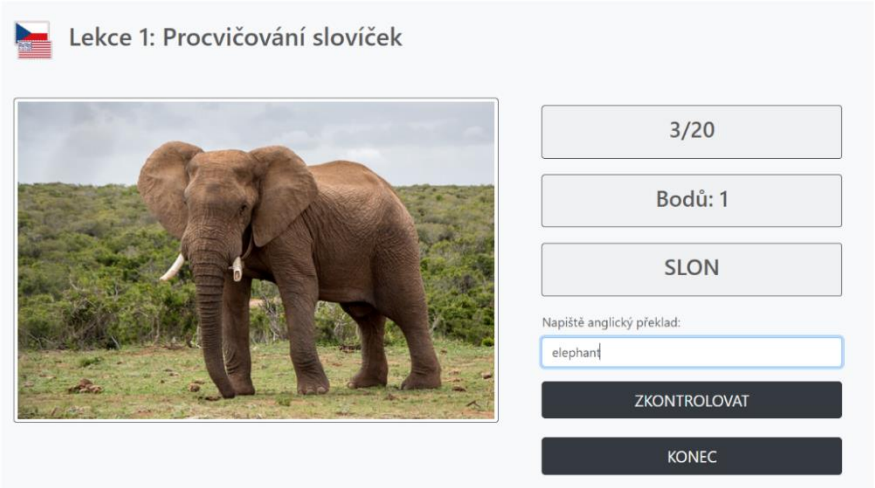
Po kliknutí na první část se otevře nová stránka a načtou se slovíčka z databáze patřící do dané lekce. Každé slovíčko zobrazuje svůj obrázek pro lepší zapamatování, český a anglický překlad. V horní části je počet slovíček v lekci a v dolní tlačítka pro přepínání dopředu, zpět a pro vrácení se zpět na úvodní stránku.



Obrázek 9: Ukázka stránky s novými slovíčky

2.2.6 Procvičování překladu slovíček

Procvičování překladu slovíček do češtiny nebo do angličtiny jsou udělány skoro totožně. U překladu do češtiny se obrázek slovíčka zobrazí až po vložení českého překladu, aby se student zaměřil na dané slovíčko a ne obrázek. V pravé části jsou zobrazeny body, které se zvyšují při správném vyhodnocení slovíčka. Dále je tam i počet slovíček a anglický nebo český překlad. Uživatelský překlad se píše do textového pole hned pod překladem, pro vyhodnocení slovíčka slouží tlačítko zkontrolovat. Po stisknutí se zobrazí nápis správně a zezelená nebo špatně a zčervená. Poté je možné pokračovat na další slovíčko pomocí tlačítka další. Po projití všech slovíček se zviditelní tlačítko vyhodnotit, které přesměrovává na stránku s vyhodnocením. Opět je možné ukončit zkoušení tlačítkem konec. Ve vyhodnocení jsou zobrazeny procentuální úspěšnost, počet dosažených bodů a maximální možný počet bodů. Dále se také zobrazí chyby se správným anglickým a českým překladem.




The screenshot shows a web interface for a vocabulary exercise. At the top, it says 'English Course' and 'filip.vondracek@gmail.com Statistky Odhlásit'. The main heading is 'Lekce 1: Procvičování slovíček'. Below the heading is a photo of an elephant. To the right of the photo, there are several elements: a box showing '3/20', a box showing 'Bodů: 1', a box showing 'SLON', a text input field with 'elephant' entered, and two buttons: 'ZKONTROLOVAT' and 'KONEC'.

Obrázek 10: Ukázka stránky s procvičováním slovíček

2.2.7 Nová gramatika

Učení nové gramatiky probíhá podobně jako učení slovíček. Jsou načteny informace z databáze, které jsou rozděleny do více částí, aby bylo učení přehlednější. V gramatice se většinou objevují tabulky nebo texty s výkladem. Přepínání mezi částmi probíhá pomocí tlačítek další a zpět.

 Lekce 2: Nová gramatika

Sloveso HAVE / HAVE GOT — Dávat si / Mít

Sloveso have/have got se používá pro vytvoření věty, kdy chceme říct že něco vlastníte nebo si něco dáváte.

Sloveso **mít(have)** se v britské angličtině používá se slovem **got** a znamená, že něco vlastníte. V americké angličtině se používá pro stejný význam jen sloveso **have**. Sloveso se dá tak použít pro význam věty ves smyslu něco si dát, pak se slovo got nepřidává.

Ve třetí osobě se používá has a všechny tvary se dají zapsat zkráceným způsobem. Více v tabulce.

Bristká varianta — jednotné číslo

Osoba	Česky	Anglicky
1. osoba	Já mám	I have got / I've got
2. osoba	Ty máš	You have got / You've got
3. osoba	On má	He has got/ He's got

1/4


ZPĚT DALŠÍ

KONEC

Obrázek 11: Ukázka stránky s novou gramatikou

2.2.8 Procvičování gramatiky

Procvičování gramatiky probíhá podobně jako procvičování sloviček. Jsou zde tlačítka pro přepínání, kontrolu a závěrečné vyhodnocení. Místo obrázku je zde velké textové pole, do kterého se vyplňuje správná odpověď. Nad ním je vždy zobrazena věta, či slovo a typ doplňování. Prvním typem je doplňování slova, které chybí ve větě a je možné napsat i celou větu. Dalším typem je přeložením věty do češtiny. Poslední typ v doplňování gramatiky je správné seřazení věty, která je zpřeházená a jednotlivá slova jsou oddělena svislou čarou. Na doplněná slova nemá vliv jejich velikost nebo tečka nebo otazník na konci věty. Ve vyhodnocení se opět zobrazí chyby a úspěšnost.

 Lekce 1: Procvičování gramatiky

Doplňte chybějící slovo

Kladná: I ___ a lion.

I am a lion

1/12

Bodů: 0

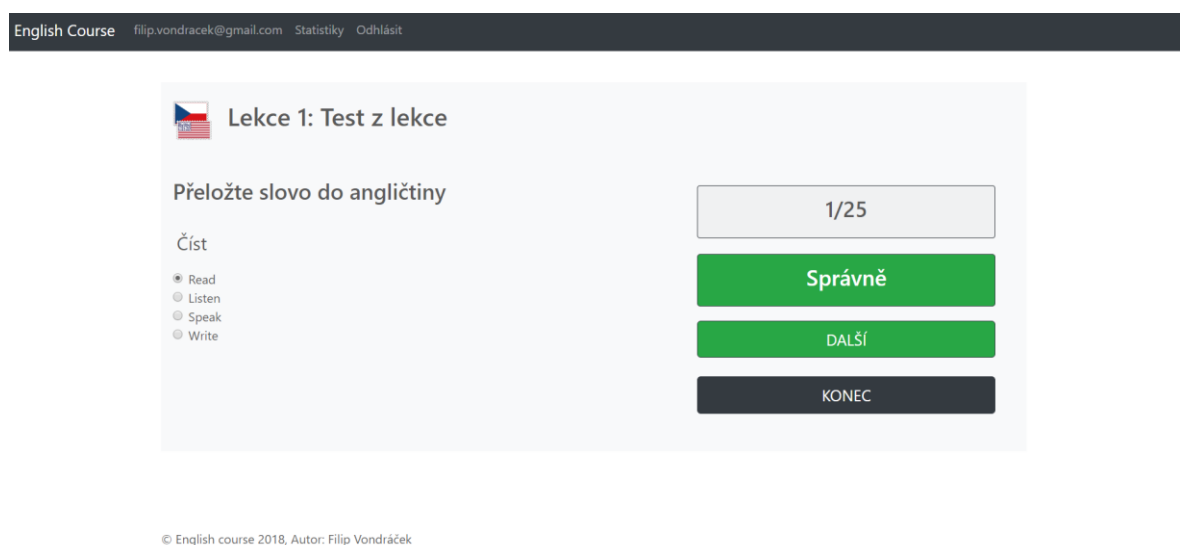
ZKONTROLOVAT

KONEC

Obrázek 12: Ukázka stránky s procvičováním gramatiky

2.2.9 Test z lekce

Test je nejdůležitější část z celé lekce. Je odrazem toho, co se student zatím naučil. Probíhá podobně jako předešlé testování. Je zde na výběr ze čtyř možností a správná je jen jedna. Po označení odpovědi je nutné opět kliknout na tlačítko zkontrolovat. Dá se tak změnit odpověď, kdyby si jí student rozmyslel. Test obsahuje několik druhů testových otázek. Obsahuje doplňování chybějícího slova ve větě, přeložení věty z angličtiny nebo z češtiny a přeložení slovíčka do angličtiny nebo češtiny. Na konci testu je opět vyhodnocení s úspěšností a s chybami.



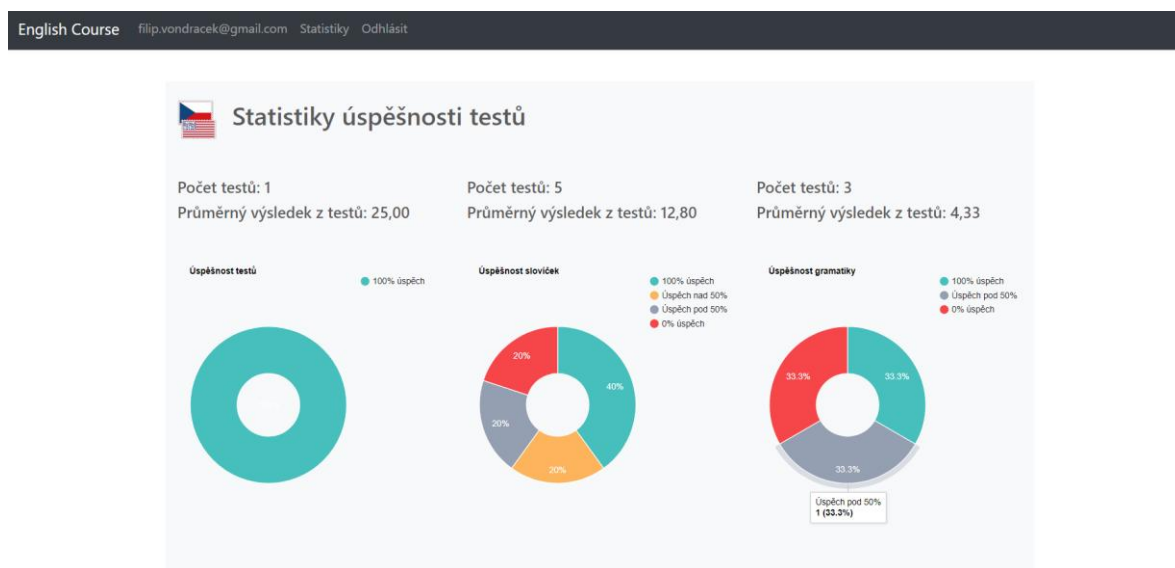
Obrázek 13: Ukázka stránky s testem z lekce

2.2.10 Test s vlastním nastavením

Test s vlastním nastavením neboli generování sestav je pro uživatele, kteří si chtějí vygenerovat test s vlastními parametry. Odkaz na test se nachází na úvodní stránce vpravo dole. Pro přesměrování se zobrazí výčet lekcí a druhů testů, které jsou k dispozici. Uživatel si zvolí jednu nebo více lekcí a jeden nebo více druhů testů. Pokud nic nezaškrtně, automaticky se vyberou všechny volby. Poté je možné vygenerovat test zmáčknutím tlačítka pokračovat a postupovat stejným způsobem jako při testu z lekce. Konečné vyhodnocení se nezapočítává do celkového výsledku kurzu ani do výsledků v lekci.

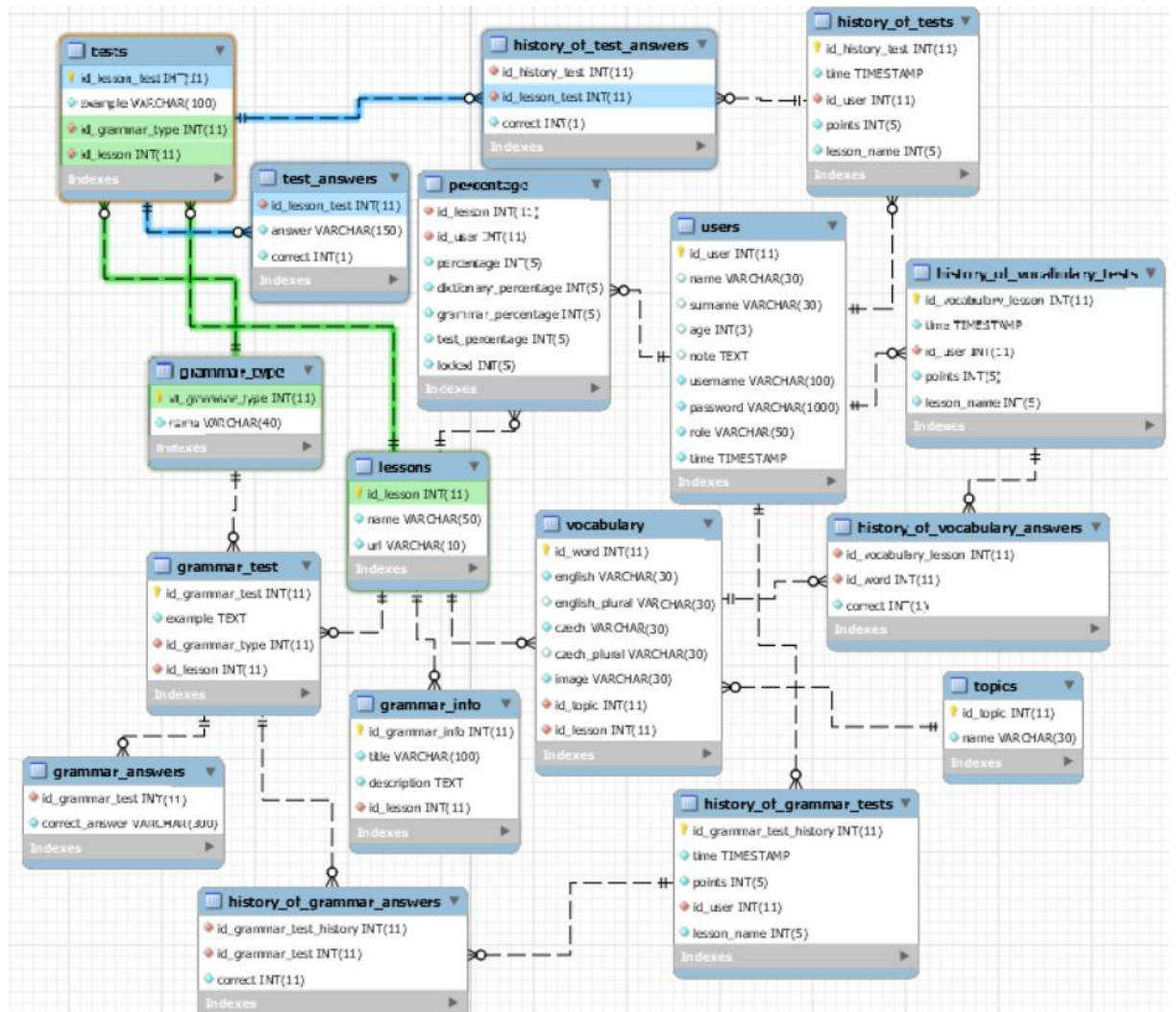
2.2.11 Statistiky uživatelů

Statistiky jsou rozděleny na dva výpisy. První je pro přihlášeného uživatele, který si může zobrazit pouze vlastní historii. Druhý výpis je pro administrátora kurzu. Odkaz je umístěn v hlavičce stránky a zobrazuje se jen uživateli s rolí administrátor. Ostatním je kvůli bezpečnosti informací skryt. Po kliknutí na odkaz se zobrazí dvě tabulky, jedna se seznamem uživatelů a druhá se seznamem lekcí. První tabulka obsahuje statistiky uživatele. Druhá tabulka s lekcemi odkazuje posledním sloupcem na podrobnější statistiky s koláčovým grafem.

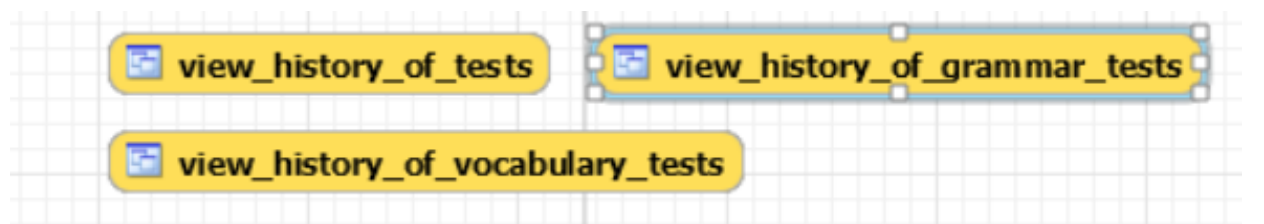


Obrázek 14: Ukázka statistiky první lekce

2.3 Databázový model



Obrázek 15: Ukázka databázového modelu



Obrázek 16: Ukázka pohledů

Aplikace spolupracuje s databázovým systémem MySQL, na kterém běží většina dnešních webových aplikací. Byl vybrán pro svoji jednoduchost a dostupnost zdarma. Na obrázcích výše jsou uvedeny pohledy a databázový model, který zobrazuje uspořádání dat v aplikaci. Model obsahuje všechny entity, atributy, cizí klíče a relace mezi entitami. Skládá se ze 17 tabulek, které jsou popsány níže.

2.3.1 Tabulky

Nejdůležitější tabulka má název Lessons v češtině lekce. Obsahuje seznam všech dostupných lekcí v kurzu. Má atributy id lekce, název lekce a url pro vytvoření odkazu.

Na tabulku Lessons je napojená tabulka Percentage. Tato tabulka obsahuje postup jednotlivých uživatelů v kurzu. Obsahuje atribut id lekce a id uživatele pro přiřazení procent ke konkrétnímu uživateli. Dále obsahuje atribut uchovávací celkový procentuální postup v lekci, atributy pro procentuální úspěšnosti cvičení ze slovíček, gramatiky a testů. Posledním atribut určuje, zda je lekce již odemčená, či ne. Tato tabulka se aktualizuje po každém ukončení testu či cvičení. Pokud je celkový procentuální postup větší, jak sedmdesát procent nastaví se další lekce jako odemčená.

Dále jsou lekce napojené na tabulku Vocabulary, která uchovává seznam všech slovíček v kurzu. Má atributy id lekce pro přiřazení slovíčka do lekce, id slovíčka a id tématu slovíček, ve které jsou uloženy názvy témat. Obsahuje také atributy pro anglický, český překlad a cestu k obrázku v úložišti aplikace. Každé slovíčko tedy vlastní obrázek.

Pro uchovávání informací o gramatice slouží tabulka Grammar_info. Tato tabulka obsahuje titulek gramatiky, popis a id lekce.

Další napojenou tabulkou na lekce je tabulka tests. Má za úkol uchovat jednotlivé otázky testů pro lekce. Obsahuje id, příklad a id typu gramatiky. Typem gramatiky může být třeba doplnění slova, český, anglický překlad a jiné. Tabulka tests je napojená na tabulku Test_answers obsahující správné odpovědi na otázky.

Poslední napojenou tabulkou na lekce je tabulka grammar_tests. Ta obsahuje seznam všech cvičení gramatiky. Má atributy id, příklad jako zadání, typ testu stejné jako u testů z lekce a id lekce pro zařazení do správné lekce. Je napojená na tabulku grammar_answers pro přiřazení správných odpovědí.

Stejně důležitou tabulkou jako lekce je tabulka uživatelé. Ta uchovává informace všech uživatelů, kteří vyplnili a odeslali registrační formulář. Informace si uživatel může později upravit ve svém profilu po kliknutí na svůj e-mail v hlavičce stránky. Má atributy id uživatele pro identifikaci, roli, která může nabývat hodnotu registrovaný a administrátor. Dále také uživatelské jméno, které je zároveň přihlašovací, heslo pro autentizaci uživatele a čas vytvoření. Tabulka také obsahuje nepovinné atributy, které uživatel nenastavuje při registraci, ale ve svém účtu a to jméno, příjmení, věk a poznámku.

Dále už zbývá jen popsat tabulky pro ukládání historie testů a cvičení. Tyto data se používají pro vytvoření statistik a historie uživatele. Všechny tyto tabulky jsou napojeny na tabulku uživatele pro správné přiřazení k uživateli.

První takovouto tabulkou je `history_of_tests`, uchovává čas vykonání testu, id a počet získaných bodů. Na tuto tabulku je napojena tabulka `history_of_tests_answers`, která uchovává špatné odpovědi v daném testu pro zobrazení chyb při výsledném vyhodnocení testu. Obsahuje id historie testů a id testu. Historie slovíček a gramatických cvičení funguje na stejném principu jako historie testů.

2.4 Ukázky zdrojových kódů

Na následujících řádcích bych chtěl popsat příklady využití MySQL dotazů, PHP a Javascript kódu.

2.4.1 Použití MySQL v aplikaci

```
public function getDictionaryForLesson($lesson) {  
  
    return $this->database->query('select english, english_plural, czech, czech_plural, image, topics.name as topic,  
        lessons.name as lesson, id_word, id_lesson  
        from vocabulary  
        join topics using (id_topic)  
        join lessons using (id_lesson) where url = ?', $lesson);  
}
```

Obrázek 17: Ukázka získávání slovíček pro lekci

První popisovaná metoda vrací pole slovíček pro danou lekci. Vybírá se anglický, český překlad, obrázek, název tématu, jméno lekce, id slovíčka a lekce. Dotaz se vybírá z tabulky slovíčka a je spojen pomocí `join` s tabulkou témata a lekce. Celý dotaz je omezen jen na slovíčka, která mají přiřazený název lekce ve sloupci `url`.

```
public function getGlobalScore($id_user) {  
    return round($this->database->fetchField('select avg(percentage) '  
        . 'from percentage where id_user = ?', $id_user), 0);  
}
```

Obrázek 18: Ukázka získání celkového procentuálního postupu v kurzu

Tato metoda získá celkový procentuální postup v kurzu. Nejdříve se vybere průměr procent ze všech lekcí a poté je omezen jen na jednoho uživatele. Id uživatele je dosazeno z parametru metody.

```

public function add($username, $password) {
    try {
        $this->database->query('insert into users (username, password) '
            . 'values (?, ?);', $username, Passwords::hash($password));
    } catch (Nette\Database\UniqueConstraintViolationException $e) {
        throw new DuplicateNameException;
    }
}

```

Obrázek 19: Ukázka vložení nového uživatele

Poslední ukázkou dotazů je vkládání nového uživatele při registraci. Je použita metoda `add` s parametry uživatelské jméno a heslo. Vkládá se pomocí `insert` nastavují se sloupce uživatelské jméno a heslo. Ostatní jsou nepovinné a je možné je přidat později z uživatelského účtu. Heslo je zabezpečeno pomocí PHP funkce `hash`. Celý dotaz je obalen výjimkou, která zachytí případné vkládání duplicitního uživatelského jména.

2.4.2 Použití PHP v aplikaci

```

protected function createComponentPasswordForm()
{
    $form = new Form();
    $form->addPassword('password', 'Heslo: ')
        ->setOption('description', 'Alespoň 6 znaků')
        ->addRule(Form::FILLED, 'Vyplňte Vaše heslo')
        ->setHtmlAttribute('class', 'form-control')
        ->addRule(Form::MIN_LENGTH, 'Heslo musí mít alespoň %d znaků', 6);
    $form->addPassword('password2', 'Heslo znovu: ')
        ->setHtmlAttribute('class', 'form-control')
        ->addConditionOn($form['password'], Form::VALID)
        ->addRule(Form::FILLED, 'Heslo znovu')
        ->addRule(Form::EQUAL, 'Hesla se neshodují', $form['password']);
    $form->addSubmit('send', 'Změnit heslo')
        ->setHtmlAttribute('style', 'margin-top:15px;')
        ->setHtmlAttribute('class', 'btn btn-dark btn-block btn-lg');
    $form->onSuccess[] = [$this, 'passwordFormSucceeded'];
    return $form;
}

```

Obrázek 20: Ukázka vykreslení formuláře pro změnu hesla

Výše uvedená funkce slouží pro vykreslení formuláře na změnu uživatelského hesla. Nejdříve se vytvoří nová instance třídy `Form`. Pomocí `addPassword` je přidáno nové textové pole pro textový vstup uživatele. Je mu nastaveno jméno na `password` a popisek na heslo. Dále je nastaven popis pro upozornění, že heslo musí být alespoň šest znaků dlouhé, povinnost ho vyplnit a třída pro nastavení stylu. Nakonec je nastaveno pravidlo, které souvisí s výše uvedeným popisem o délce hesla. Druhý formulářový prvek je identický, ale je u něho navíc

nastaveno pravidlo, že text hesla se musí shodovat s tím předešlým. Poslední prvek je tlačítko pro potvrzení změny hesla. Je mu nastaveno jméno na odeslat a popis na změnit heslo. Dále pak styl a třída stylu. OnSuccess volá metodu pro odeslání dat po úspěšném vyplnění formuláře. Na závěr se už jen pomocí return vrátí formulář k vykreslení.

```
public function renderEvaluation($lesson) {
    if (!$this->getUser()->isLoggedIn()) {
        $this->redirect('Sign:in');
    }

    //Vrátí počet bodů do proměnné
    $points = $this->dictionaryManager->getPointsResult($lesson, $this->getUser()->id);
    //Vrátí do proměnné počet sloviček
    $count = $this->dictionaryManager->getWordsCount($lesson);
    //Získá chyby ve slovičkách pro vykreslení
    $this->template->results = $this->dictionaryManager->getResult($lesson, $this->getUser()->id);
    //Vrací počet bodů ze cvičení pro vykreslení
    $this->template->points = $points;
    //Vrací počet sloviček pro vykreslení
    $this->template->count = $count;
    //Vrací procentuální úspěšnost cvičení ze sloviček pro vykreslení
    $this->template->percentage = number_format(($points/$count)*100,2,".", " ");
    //Ukládá procentuální úspěšnost do databáze
    $this->dictionaryManager->savePercentage(($points/$count)*100, $lesson, $this->getUser()->id);
}
```

Obrázek 21: Ukázka získání dat pro vykreslení vyhodnocení

Další funkce je pro získání dat z databáze pro vykreslení šablony pro vyhodnocení. Nejdříve se pomocí funkce isLoggedIn zjistí, zda je uživatel přihlášen. Pokud není, tak je přesměrován na přihlašovací stránku, jinak se pokračuje dále v kódu. Načte se do proměnné počet bodů z databáze daného uživatele a lekce, dále se načte i počet sloviček v lekci pro zjištění maximálního počtu bodů. Také jsou vypsány všechny chyby, co uživatel udělal, které se vypíší s obrázkem, se správným anglickým a českým překladem. Také se spočítá procentuální úspěšnost v lekci vydělením bodů počtem sloviček a následným vynásobením stem pro získání procent. To celé se ještě zaokrouhlí a nastaví se desetinná čárka pomocí funkce number_format, kde první parametr je zobrazované číslo, druhý počet desetinných míst, třetí pro oddělovač celých čísel a poslední pro oddělovač tisíců. Nakonec se ještě procentuální úspěšnost uloží do databáze, ke konkrétnímu uživateli.

2.4.3 Použití JavaScriptu v aplikaci

```
function setImage(name, title) {
    document.getElementById("wordImage").src = cesta + "/img/dictionary/" + name;
    document.getElementById("wordImage").alt = title;
    document.getElementById("wordImage").title = firstToUpper(title.toLowerCase());
}
```

Obrázek 22: Ukázka změny obrázku při cvičení sloviček

Pro změnu na straně klienta je používán Javascript. Cvičení slovíček obsahují více slovíček, mezi kterými se přepíná tlačítkem. Výše uvedená funkce nastavuje obrázek při kliknutí na tlačítko další. Obsahuje parametry se jménem a titulkem obrázku. V prvním příkazu se mění cesta k jinému obrázku. Druhý mění popis a třetí titulek, který se zobrazí při najetí myši na obrázek.

2.5 Bezpečnost dotazů

Velké bezpečnostní riziko pro databáze jsou formuláře, kde uživatel zadává volitelné textové řetězce a může tak lehce napadnout databázi.

```
$sql = "SELECT id, name FROM users where id= ".$_POST["name"];
```

Obrázek 23: Ukázka bezpečnosti prvního dotazu do databáze

Pokud by útočník zadal do formuláře řetězec obsahující `0 or 1=1`, dotaz by vypsal všechny uživatele a útočník by se tak mohl například lehce autentizovat. Další možnost, jak napadnout databázi, je vložit řetězec `0; delete from users`. U tohoto dotazu musíme znát jméno tabulky a způsobí vymazání tabulky `users`.

```
$sql = "update users set age = ".$_POST["name"]." where id= 6";
```

Obrázek 24: Ukázka bezpečnosti druhého dotazu do databáze

Druhý příklad je bezpečnost při aktualizaci tabulky `users`. Při tomto dotazu může útočník vložit řetězec obsahující `22, bonus = 1000` a nastavit si tak nejen věk na 22, ale i bonus na hodnotu 1000. Je zde potřeba znát název sloupce `bonus`.

```
$this->database->fetchField('SELECT id, name FROM users WHERE id= ?', $promennaZFormulare);
```

Obrázek 25: Ukázka bezpečného dotazu do databáze

Řešením je předávat vkládaná data jako parametry dotazu a říct tak, co jsou data a co dotaz. Vložená podmínka se tak převede na řetězec a bude tak součástí porovnávaného údaje.

2.6 UML Use case diagram

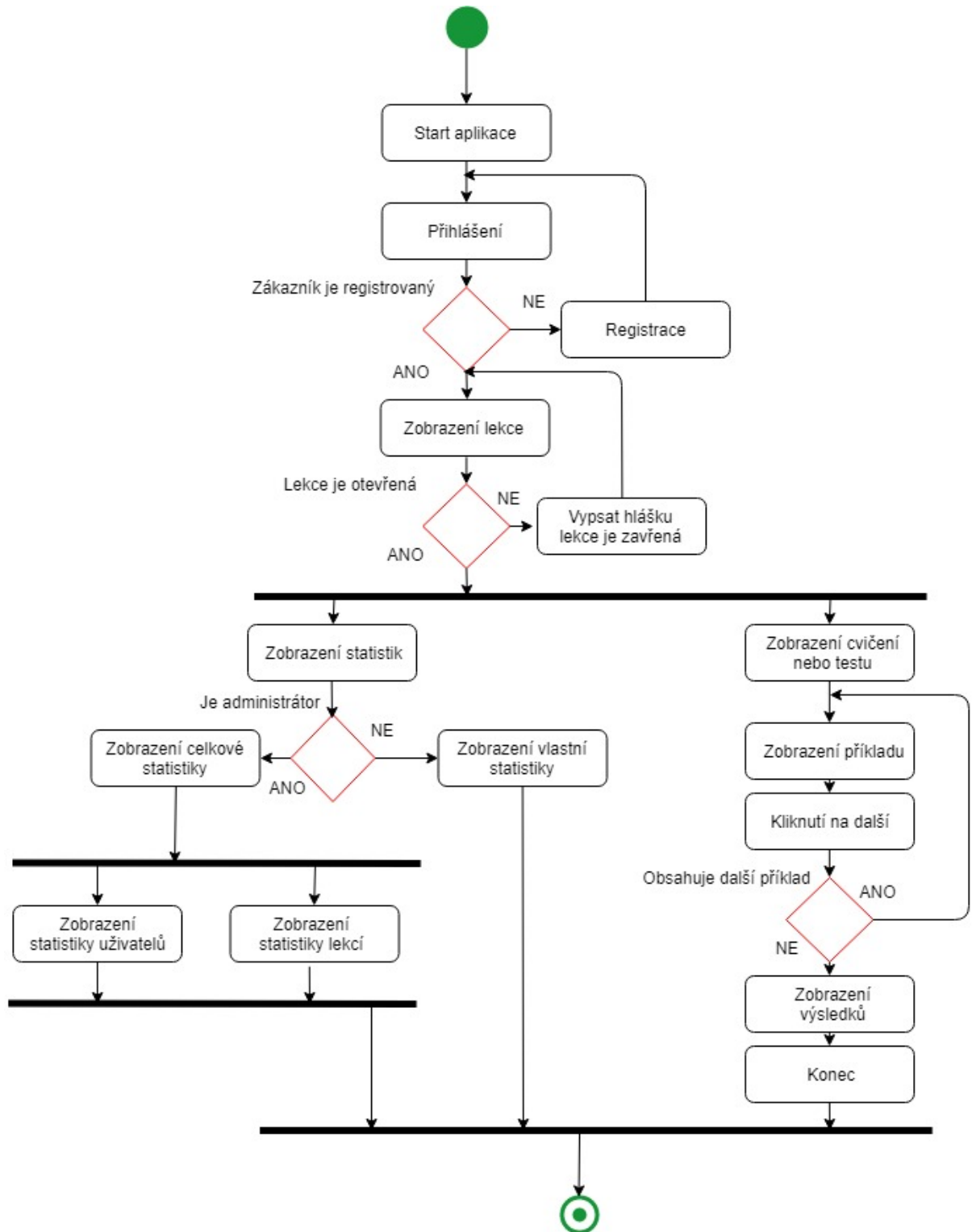


Obrázek 26: UML Use case diagram rolí

Use case diagram, v českém překladu diagram případů užití, vykresluje používání aplikace z pohledu běžného uživatele. Popisuje možnosti, které uživatel má v aplikaci a zobrazuje je pro něj v čitelnější podobě. Diagram už nezachycuje, jak se dané problémy řeší. Je to jeden z prvních kroků při konzultaci se zákazníkem, který aplikaci požaduje. Pomáhá to tedy lépe porozumět jeho požadavkům. Diagram výše popisuje tři uživatelské role a jejich možnosti v systému. [19]

2.7 UML Activity diagram

Aktivity diagram obsahuje nejdříve přihlášení a poté průběh testování, cvičení a zobrazení statistik lekcí a uživatelů.



Obrázek 27: UML Activity diagram

2.8 Instalační příručka

Pro zprovoznění kurzu je třeba nainstalovat Xampp server a vložit do složky htdocs přiložené soubory, poté už jen stačí zobrazit v prohlížeči stránku localhost a stránka se načte. Instalace je popsána v předešlé kapitole 2.1.1 XAMPP a Apache. Dále je nutné vytvořit a naplnit databázi pomocí příkazů, které jsou přiloženy v příloze.

2.9 Uživatelská příručka

Po instalaci potřebného softwaru, která je popsána v předešlé kapitole, je třeba otevřít stránku localhost ve webovém prohlížeči.

Pro přístup do kurzu je nejdříve nutné se registrovat pomocí odkazu registrovat. Zde nalezneme formulář, kam vložíme údaje a zmáčkneme tlačítko registrovat. Poté je možné se do kurzu přihlásit pomocí odkazu přihlásit, kde vložíme své přihlašovací údaje a potvrdíme.

Po přihlášení je možné spustit první lekci kurzu, ve které najdeme požadované cvičení. Pro návrat stačí stisknout tlačítko konec. Více podrobných informací s popisem funkcí obsahuje kapitola 2.2 Aplikační řešení.

2.10 Adresářová struktura

Pro vytvoření online kurzu byl použit framework Nette, jehož adresářovou strukturu bych chtěl dále popsat. V hlavní složce se nachází složka app, www a ostatní složky se týkají přímo frameworku a není potřeba do nich nijak během vývoje zasahovat. Složka www obsahuje složku css pro vytvoření souboru pro styly. Tyto styly je třeba připojit v hlavičce stránky, aby byly dostupné. Dále je tam složka js pro soubory s Javascriptem, které se taktéž musí připojit odkazem. Také obsahuje složku image, kde se nachází všechny obrázky použité v aplikaci. Již výše zmiňovaná složka app obsahuje všechny php soubory pro vykreslení stránky a načítání z databáze. Obsahuje složku router, ve které je soubor s nastavením, jak se bude přepisovat url adresa na hezčí tvar. Dále složka coursemodule, která obsahuje složku config s nastavením připojení do databáze. Také se ve složce coursemodule nachází složka model. Ta obsahuje všechny metody s dotazy do databáze. Složka presenters obsahuje načítání dat do šablon a složka templates obsahuje samotné šablony. Šablony tvoří html značky a makra definované ve frameworku.

ZÁVĚR

Na základě zadání bakalářské práce byla vytvořena aplikace sloužící pro výuku anglického jazyka. Byly implementovány všechny požadavky jako lekce, statistiky, uživatelské účty a testy lekcí. V textové části byly popsány podobné systémy a na základě toho pak navrhnout vlastní systém.

Také byly popsány použité technologie. Na výběr bylo použít programovací jazyk Java nebo PHP a databázový systém Oracle nebo MySQL. Zvolil jsem jazyk PHP, který je pro online kurz vhodnější a MySQL, protože je levnější na správu, a především je otevřený a zdarma. Pro práci s PHP jsem využil český framework Nette, který je zdarma a také otevřený. Usnadnil tak práci s databází a tvořením nových stránek. Také umožnil lepší přehlednost a možné budoucí rozšíření.

Dále jsem použil technologii HTML pro tvorbu struktury aplikace a CSS pro design. CSS bylo využito společně s frameworkem Bootstrap, který je snadno naučitelný a vytvořená aplikace je i responsivní. To znamená, že se přizpůsobí i menšímu zařízení, jako jsou mobilní telefony a tablety. Pro tvorbu jednotlivých testů se nejlépe hodilo využít technologii JavaScript. Ta má na starosti dynamicky měnit obsah testů, například při přepínání jednotlivých slovíček bez nutnosti znova se dotazovat na server.

V poslední části byla popsána celá aplikace se všemi možnostmi, které nabízí. Aplikace slouží pro školní účely a není připravená do ostrého komerčního provozu vzhledem k tomu, že by se mohly vyskytnout chyby, které při vývoji nabyly zachyceny. Bylo by třeba aplikaci více otestovat a doplnit kvalitnější obsah pro výuku anglického jazyka, protože zatím obsahuje jen pár základních lekcí.

Při vývoji kurzu jsem se naučil s frameworkem Nette, který byl pro mě nový. Bylo nejdříve nutné pochopit, jak celý systém funguje a naučit se MVP architekturu, která je popsána v kapitole věnující se Nette frameworku. Také jsem se nově naučil pracovat s CSS frameworkem Bootstrap, který mě překvapil svojí jednoduchostí a absencí opakování nastavování stylů u jednotlivých elementů. S prací jsem spokojen a určitě se k vývoji webových aplikací vrátím u jiných projektů.

POUŽITÁ LITERATURA

- [1] ČEGAN, Lukáš. *Vývoj webových aplikací v PHP a NetBeans*. Pardubice: Univerzita Pardubice, 2015. ISBN 978-80-7395-858-9.
- [2] BAAR, Ondřej. Webové aplikace na domácím PC. In: *Pcworld.cz* [online]. Praha: IDG Czech Republic, 2008 [cit. 2018-04-01]. Dostupné z: <https://pcworld.cz/archiv/webove-aplikace-na-domacim-pc-17306>
- [3] KOFLER, Michael a Bernd ÖGGL. *PHP 5 a MySQL 5: průvodce webového programátora*. Brno: Computer Press, 2007. ISBN 978-80-251-1813-9.
- [4] KOZINA, Jakub. První díl o PHP – Historie a budoucnost. In: *Phgame.cz* [online]. ©2010-2015 [cit. 2018-04-01]. Dostupné z: http://phgame.cz/PHGame_serialy/serialy/php/prvni-dil-o-php-historie-a-budoucnost
- [5] MASLAKOWSKI, Mark. *Naučte se MySQL za 21 dní*. Praha: Computer Press, 2001. Všechny cesty k informacím. ISBN 80-722-6448-6.
- [6] HORDĚJČUK, Vojtěch. Jazyk SQL. In: *Voho.eu* [online]. ©2008-2018 [cit. 2018-03-22]. Dostupné z: <http://voho.eu/wiki/sql/>
- [7] WELLING, Luke a Laura THOMSON. *MySQL: průvodce základy databázového systému*. Brno: CP Books, 2005. ISBN 80-251-0671-3.
- [8] GRUDL, David. Nette Framework: zvyšte svoji produktivitu. In: *Zdrojak.cz* [online]. Praha: Devel.cz Lab, 2009 [cit. 2018-02-07]. Dostupné z: <https://www.zdrojak.cz/clanky/nette-framework-zvyste-svoji-produktivitu/>
- [9] ČAPKA, David. 1. díl – Úvod do Nette frameworku pro PHP. In: *Itnetwork.cz* [online]. Praha, ©2018 [cit. 2018-04-08]. Dostupné z: <https://www.itnetwork.cz/php/nette/zaklady/uvod-do-php-frameworku-nette/>
- [10] CASTRO, Elizabeth a Bruce HYSLOP. *HTML5 a CSS3: názorný průvodce tvorbou WWW stránek*. Brno: Computer Press, 2012. ISBN 978-80-251-3733-8.
- [11] JavaScript. In: *Tvorba-webu.cz* [online]. ©2003-2008 [cit. 2018-03-22]. Dostupné z: <https://www.tvorba-webu.cz/javascript/>

- [12] ZELINKOVÁ, Šárka. Pohled na metody výuky cizích jazyků 20. století. In: *Lingvistikapraha.ff.cuni.cz* [online]. Praha, 2013 [cit. 2018-02-11]. Dostupné z: <http://lingvistikapraha.ff.cuni.cz/node/182>
- [13] BORONCZYK, Tim. *PHP 6, MySQL, Apache: vytváříme webové aplikace*. Brno: Computer Press, 2009. ISBN 978-802-5127-674.
- [14] Historie PHP. In: *Garth.cz* [online]. [cit. 2018-03-22]. Dostupné z: <http://www.garth.cz/uvod-do-php/historie-php/>
- [15] KOPAL, Ondřej. Historie a trendy ve vývoji databází. In: *Web-integration.info* [online]. Praha: Lundegaard, 2015 [cit. 2018-03-22]. Dostupné z: <http://www.web-integration.info/cs/blog/historie-a-trendy-ve-vyvoji-databazi/>
- [16] TRONNER, Pavel. Jak se zrodilo programování: Grace Hopper. In: *Zive.cz* [online]. Praha: CN Invest a.s, 2015 [cit. 2018-03-22]. Dostupné z: <https://www.zive.cz/clanky/jak-se-zrodilo-programovani-grace-hopper/sc-3-a-180421/default.aspx>
- [17] KRÁL, Jaroslav. Obálka čísla Půlstoletí programovacích jazyků. In: *Vesmir.cz* [online]. Praha: Vesmír, 2008 [cit. 2018-04-08]. Dostupné z: <https://vesmir.cz/cz/casopis/archiv-casopisu/2008/cislo-12/pulstoleti-programovacich-jazyku.html>
- [18] 14. Programovací jazyky. In: *Ivt.mzf.cz* [online]. ©2018 [cit. 2018-04-08]. Dostupné z: <http://www.ivt.mzf.cz/seminar/14-programovaci-jazyky/>
- [19] ČAPKA, David. 2. díl – UML – Use Case Diagram. In: *Itnetwork.cz* [online]. Praha, ©2018 [cit. 2018-04-08]. Dostupné z: <https://www.itnetwork.cz/navrh/uml/uml-use-case-diagram>
- [20] DRUSKA, Peter. *CSS a XHTML: tvorba dokonalých webových stránek krok za krokem*. Praha: Grada, 2006. Průvodce (Grada). ISBN 80-247-1382-9.
- [21] *Duolingo.com/* [online]. [cit. 2018-04-23]. Dostupné z: <https://www.duolingo.com/>
- [22] *Onlinejazyky.cz/* [online]. Hradec Králové: Online learning, ©2007-2017 [cit. 2018-04-23]. Dostupné z: <https://www.onlinejazyky.cz/>

- [23] *Englishme.cz* [online]. Přelouč: Vitware, ©2013 [cit. 2018-04-23]. Dostupné z: <https://www.englishme.cz>
- [24] *Anglictina-bez-biflovani.cz* [online]. Praha: Euvit [cit. 2018-04-23]. Dostupné z: <https://www.anglictina-bez-biflovani.cz/>
- [25] *Pixabay.com* [online]. ©2018 [cit. 2018-05-02]. Dostupné z: <https://pixabay.com/>

PŘÍLOHY

Příloha A – Instalační CD	56
---------------------------------	----

PŘÍLOHA A – INSTALAČNÍ CD

Příložené CD obsahuje následující části:

- Plné zdrojové kódy aplikace.
- Databázový model, obrázek Use-case diagramu, obrázek UML aktivity diagramu.
- Soubor s SQL příkazy pro vytvoření a naplnění databáze aplikace.
- A také obsahuje tento dokument.