

Webová aplikace pro podporu administrativních procesů střediska ECDL

Bakalářská práce

Vedoucí práce:

Ing. Petr Jedlička, Ph.D.

Roman Péchal

Brno 2016

Poděkování

Rád bych poděkoval Ing. Petru Jedličkovi, Ph.D. za cenné rady, věcné připomínky a vstřícnost při konzultacích a vypracování bakalářské práce. Dále bych chtěl poděkovat své rodině za podporu po celou dobu studia.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Webová aplikace pro podporu administrativních procesů střediska ECDL**

vypracoval/a samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*.

Jsem si vědom/a, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.

Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmetná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

V Brně dne 4. ledna 2016

Abstract

Péchal, R. Web application for administrative processes related to ECDL centre. Bachelor thesis. Brno: Mendel University, 2016.

The bachelor thesis deals with implementation of web application for administrative processes related to ECDL centre. In the first part, there is a description of ECDL concept and introduction of administrative processes related to ECDL. The application is designed based on consultations with manager of ECDL centre at Mendel University. Implementation is done with PHP with the use of Nette framework and MySQL database management system.

Keywords

ECDL, web application, PHP, Nette, MySQL

Abstrakt

Péchal, R. Webová aplikace pro podporu administrativních procesů střediska ECDL. Bakalářská práce. Brno: Mendelova univerzita v Brně, 2016.

Práce pojednává o implementaci webové aplikace pro podporu administrativních procesů střediska ECDL. V první části se věnuje popisu konceptu ECDL a seznamuje s administrativními procesy každého testovacího střediska. Práce obsahuje výstupy z konzultací s manažerem střediska při Mendelově univerzitě v Brně, na základě kterých je vytvořen návrh aplikace. Implementace je provedena v jazyce PHP s využitím Nette frameworku a databázového systému MySQL.

Klíčová slova

ECDL, webová aplikace, PHP, Nette, MySQL

Obsah

1	Úvod	7
2	Cíl práce	8
3	Metodika řešení	9
4	Analýza fungování středisek ECDL	10
4.1	Koncept ECDL	10
4.1.1	Program ECDL Core.....	10
4.1.2	Program ECDL Advanced.....	11
4.2	Testovací středisko Mendelovy univerzita v Brně	11
4.2.1	Současný stav.....	11
4.3	Ostatní testovací střediska	12
4.4	Fungování středisek ECDL.....	12
4.5	Administrativní procesy	13
4.5.1	Studenti.....	13
4.5.2	Testeři.....	14
4.6	Shrnutí požadavků.....	15
5	Návrh webové aplikace	16
5.1	Konceptuální databázový model.....	16
5.2	Obrazovky uchazeče.....	17
5.2.1	Uživatelský účet.....	17
5.2.2	Přehled čerpání služeb.....	18
5.2.3	Termíny testování.....	18
5.3	Obrazovky pracovníka střediska.....	19
5.3.1	Objednávky	19
5.3.2	Seznam uchazečů	19
5.3.3	Správa termínů testování.....	20
5.3.4	Vkládání výsledků.....	20
5.3.5	Správa nabízených produktů a služeb	21

5.3.6	Úkoly	22
5.3.7	Statistiky	22
5.4	Technologie	23
5.4.1	Databázový systém.....	23
5.4.2	Programovací jazyk.....	23
5.4.3	Framework	24
5.4.4	Frontend	24
5.4.5	Dokumentace	25
6	Implementace a výsledky	26
6.1	Správa termínu testování.....	26
6.2	Indexy a výsledky	28
6.3	Emailová komunikace	29
6.4	Dokumentace	29
7	Závěr	32
8	Literatura	34

1 Úvod

Počítačová gramotnost je neustálým tématem diskuzí a to především ve vzdělávacích společnostech. Státy se snaží popsat a zhodnotit ICT znalosti a schopnosti obyvatelstva a stanovit budoucí strategii (Ilomäki, 2011).

Na zaměstnance v téměř každém oboru jsou kladeny nároky na určitou úroveň počítačové gramotnosti. Z tohoto důvodu byl skupinou ECDL Foundation (1997) založen koncept s názvem European Computer Driving Licence (ECDL). Jedná se o celosvětově rozšířený vzdělávací koncept v oblasti počítačové (digitální) gramotnosti a digitálních znalostí a dovedností. Studentům je umožněno získat různé úrovně certifikátů, které poté slouží jako doklad o jejich znalostech a schopnostech práce s ICT. Garanty kvality ECDL testování jsou mezinárodní profesní společenství Council of European Professional Informatics Societies (Cepis, 2015).

Certifikační programy ECDL Foundation (1997) jsou navrženy tak, aby byly dostupné všem občanům bez ohledu na jejich věk, rasu a pohlaví. V současné době se programů zúčastnilo již 14 milionů studentů, kteří měli možnost studovat ve 41 jazycích napříč 150 zeměmi prostřednictvím více než 24000 testovacích center.

ECDL není však pouze záležitostí evropských států, jeho certifikáty jsou uznávány po celém světě. V zemích mimo Evropu se tedy setkáváme se zkratkou ICDL (International Computer Driving License) (ECDL Foundation, 1997).

V České republice je možné se s konceptem ECDL setkat od roku 1999, kdy se Česká společnost pro kybernetiku a informatiku stala držitelkou ECDL licence a garantem kvality ECDL certifikátů pro ČR. ČSKI uděluje licenční práva k provádění ECDL testů těm střediskům, která splní kritéria stanovená ECDL Foundation (Česká společnost pro kybernetiku a informatiku, 2015).

ECDL/ICDL je tedy považováno za vedoucí certifikační autoritu počítačové gramotnosti vyhledávanou studenty, zaměstnanci, zaměstnavateli a širokou veřejností (Csapo, 2002).

V současné době probíhají veškeré administrativní procesy mezi studenty a testery téměř všech akreditovaných středisek ECDL ve formě emailové komunikace nebo osobního kontaktu. To je s ohledem na dnešní možnosti uživatelsky nepřívětivé a může v konečném důsledku znamenat nižší zájem o samotné testování a certifikaci. Z těchto důvodů vznikl na Mendelově univerzitě v Brně, která disponuje akreditovaným střediskem pro ECDL testování, požadavek na vytvoření webové aplikace pro podporu těchto procesů. Ta by měla umožnit jak studentům, tak testerům provádět jednotlivé úkony odkudkoli, kde mají přístup k internetu bez nutnosti čekat na reakci druhé strany. Implementace takového informačního systému poté sníží nároky na administraci a umožní značnou část operací automatizovat.

2 Cíl práce

Cílem práce je seznámit se s pravidly fungování testovacích středisek ECDL a se související administrativou. Na základě těchto poznatků a konzultací s manažerem střediska ECDL na Mendelově univerzitě v Brně navrhnout a implementovat webovou aplikaci pokrývající vybrané administrativní procesy střediska ECDL. Aplikaci je nutno doplnit o programátorskou dokumentaci pro případ budoucího zásahu jiných programátorů a také aplikaci otestovat a odstranit případné chyby.

3 Metodika řešení

První část práce bude věnována popisu a analýze fungování středisek ECDL a s nimi spojenými administrativními procesy. Budou uvedeny požadavky vznesené manažerem střediska na výslednou webovou aplikaci.

Na základě zjištěných požadavků bude navržen konceptuální databázový model a statická podoba aplikace, která bude ve spolupráci s manažerem střediska ECDL iterativním způsobem upravována do požadované podoby. Bude navržen vzhled a obsah jednotlivých obrazovek, na základě kterých poté proběhne implementace webové aplikace. V této fázi budou zvoleny odpovídající technologie vhodné pro tvorbu informačního systému, konkrétně systém řízení báze dat, programovací jazyk pro backendovou část aplikace a technologie pro frontend.

V kapitole Implementace budou předvedeny ukázky z hotové aplikace a vytvořené dokumentace. Dále budou popsány důležité části informačního systému, které budou doplněny o ukázky fyzického modelu implementovaného v konkrétním SŘBD. Bude předvedena podoba výsledného uživatelského rozhraní a způsob fungování důležitých částí aplikace.

Na závěr budou shrnuty dosažené výsledky, zhodnocena výsledná aplikace a proběhne porovnání nového řešení s původním. Dále budou nastíněny možnosti rozšíření aplikace do budoucna.

4 Analýza fungování středisek ECDL

4.1 Koncept ECDL

Přínos konceptu ECDL (European Computer Driving Licence, v mimoevropských zemích označovaný jako ICDL – International Computer Driving Licence) spočívá v tom, že prostřednictvím ECDL Sylabů definuje vzdělávací obsah, který odráží potřeby trhu práce a života v současné informační společnosti zejména v oblasti přenositelných digitálních kompetencí a současně nabízí mezinárodně uznávanou, standardizovanou, objektivní a nezávislou metodu pro ověření výsledků vzdělávání v této oblasti. Svým rozsahem koncept ECDL pokrývá prakticky všechny oblasti, ve kterých se informační a komunikační technologie v pracovním i osobním životě běžně využívají (ECDL: ECDL Czech Republic, ©1999-2015).

Koncept ECDL zahrnuje celou škálu vzdělávacích a certifikačních programů v oblasti přenositelných digitálních kompetencí, z nichž nejrozšířenější jsou programy ECDL Core (počítačová, resp. digitální gramotnost a digitální kvalifikace) a program ECDL Advanced (profesionální uživatelské znalosti a dovednosti). Pro úplné začátečníky jsou pak určeny programy pro digitální začleňování EqualSkills a e-Citizen. K dispozici jsou také programy zasahující částečně do oblasti specifických digitálních kompetencí, např. program Digitální fotografie nebo programy pro technické kreslení či informační systémy, ale i programy pro specifické sociální a profesní cílové skupiny (ECDL: ECDL Czech Republic, ©1999-2015).

Každý studijní program obsahuje seznam modulů, které mají přiřazen syllabus, kde student najde požadavky na hloubku a rozsah jeho znalosti, které jsou vyžadovány pro splnění modulu (Portál ECDL, 2015). Student, který úspěšně absolvuje daný modul, by tedy měl mít znalosti odpovídající syllabu. Ty jsou periodicky a centrálně aktualizovány a následně národními licenciáty ECDL lokalizovány (ECDL: ECDL Czech Republic, ©1999-2015).

4.1.1 Program ECDL Core

Program ECDL Core (základní) pokrývá základní, resp. běžně očekávané uživatelské počítačové znalosti a dovednosti. Je synonymem pro počítačovou (digitální) gramotnost a digitální kvalifikaci. Je určen široké veřejnosti a vede k získání některého z následujících mezinárodně platných ECDL certifikátů.

Certifikát ECDL Start (ECDL Start Certificate) je mezinárodní doklad počítačové, (digitální) gramotnosti držitele (Digital Literacy). Dokládá, že držitel tohoto certifikátu je digitálně gramotný, tj. že ovládá základy práce s počítačem, resp. s informačními a komunikačními technologiemi (ICT), a to v nejběžnějších oblastech jejich využití. Certifikát ECDL dokládá, že držitel je plně digitálně gramotný (ECDL: ECDL Czech Republic, ©1999-2015).

4.1.2 Program ECDL Advanced

Program ECDL Advanced (pokročilý) pokrývá profesionální uživatelské počítačové znalosti a dovednosti v oblasti kancelářských aplikací, tj. v oblasti práce s textem, kalkulačními tabulkami, databázemi a prezentacemi. Je určen především zkušeným uživatelům výpočetní techniky a vede k získání některého z následujících mezinárodně platných ECDL certifikátů.

Certifikát ECDL Advanced (ECDL Advanced Certificate) je mezinárodní doklad profesionálních uživatelských počítačových znalostí a dovedností držitele (Digital Excellence) v odpovídající oblasti. Držitel tohoto certifikátu je schopen účelně, efektivně a na profesionální úrovni využívat kancelářské aplikace v oblasti, pro kterou byl doklad vydán. V případě certifikátu ECDL Expert je držitel schopen účelně, efektivně a na profesionální úrovni využívat kancelářské aplikace ve všech běžných oblastech (ECDL: ECDL Czech Republic, ©1999-2015).

4.2 Testovací středisko Mendelovy univerzity v Brně

Akreditované testovací a školicí středisko při Mendelově univerzitě v Brně nabízí testování ve dvou programech a modulech uvedených v Tab. 1. Univerzita získala akreditaci v listopadu 2005 a je jediným střediskem se sídlem v Brně, které má právo pořádat testy z programu ECDL Advanced (veškeré údaje platí v době zpracování práce).

Tab. 1 Nabízené programy a moduly testovacím střediskem Mendelovy univerzity v Brně

ECDL Core	ECDL Advanced
Modul M2: Základy práce s počítačem a správa souborů	AM3 - Pokročilé zpracování textu
Modul M3: Zpracování textu	AM4 - Pokročilá práce s tabulkovým procesorem
Modul M4: Práce s tabulkami	AM5 - Pokročilé použití databází
Modul M5: Použití databází	AM6 - Pokročilá tvorba prezentací
Modul M6: Prezentace	
Modul M7: Základy práce s internetem a komunikace	
Modul M12: Bezpečné používání informačních technologií	

Zdroj: Portál ECDL, 2015.

4.2.1 Současný stav

Požadavek na vytvoření webové aplikace pro podporu administrativních procesů střediska ECDL vznikl z důvodu absence jakéhokoli aplikačního řešení.

Veškeré úkony studentů nebo potenciálních studentů, typicky objednávka služeb, přihlašování na testování atd., probíhá v rámci elektronické komunikace nebo osobně v testovacím středisku. Pokud chce uchazeč informace o stavu testování, musí si je vyžádat a čekat na odezvu ze střediska.

Vypisování termínů, které provádí akreditovaní testeři, probíhá na webu střediska při Mendelově univerzitě¹. Veškeré informace musí být udržovány v elektronických dokumentech, např. Excel dokumenty. Pokud student pravidelně nekontroluje webové stránky střediska, může mu vypsání termínu uniknout, možnost, jak tomu zabránit je zaslat při každém vypsání termínu informaci na všechny studenty, to však vyžaduje další čas, který musí tester vynaložit.

4.3 Ostatní testovací střediska

Zpracování práce předcházelo hledání řešení správy administrativních procesů v jiných střediscích než na Mendelově univerzitě. Naprostá většina středisek má však shodné řešení bez využití jakékoli webové nebo jiné aplikace.

Za zmínku stojí testovací středisko na České zemědělské univerzitě v Praze, které má vlastní informační systém, do kterého studenti po objednání potřebných služeb, obdrží přihlašovací údaje a mohou se jeho prostřednictvím přihlašovat na termíny a zobrazovat výsledky. Aplikaci pro potřeby univerzity vytvořila firma Direction.cz (ECDL - European Computer Driving Licence – ČZU, ©2001-2011).

Ve své bakalářské práci se Laník (2014) věnoval implementaci webové podpory pro ECDL umožňující přihlašování studentů na termíny, stahování studijních materiálů a vkládání výsledků. Informační systém byl řešen na základě požadavků testovacího střediska na Přírodovědecké fakultě Ostravské univerzity v Ostravě². Testování v tomto středisku probíhá pouze z jednoho programu a tomu byl uzpůsoben návrh systému. V případě potřeby rozšíření nabízených programů by bylo nutné provést rozsáhlé úpravy návrhu a implementace aplikace. Informační systém dále nenabízí správu platforem pro testování a různých verzí testů. Dále aplikace neumožňuje jednoduchým způsobem převádět indexy studentů a již absolvované moduly z jiných středisek. Podle webových stránek střediska nebyla dosud aplikace spuštěna do ostrého provozu.

4.4 Fungování středisek ECDL

Každý student, který má zájem účastnit se testování, musí mít platný index, do kterého je možné zapisovat splněné moduly. Každý program nabízí minimálně jeden index, do kterého lze výsledky z daného programu zapisovat. Student si tedy zvolí index v požadovaném programu, který uhradí na konto střediska, jakmile je platba přijata, je studentovi index přiřazen. Dále musí mít uchazeč zaplacen každý pokus, který chce absolvovat. Většina středisek proto nabízí balíčky, které obsahují index

¹ Dostupné na adrese www.pefka.mendelu.cz/ecdl/

² Dostupné na adrese prf-czv.osu.cz/ecdl/

a předplacené pokusy včetně certifikátu, na který budou mít po splnění podmínek nárok. Uchazeči je však umožněno nakupovat jednotlivé služby zvlášť.

Po splnění podmínek vlastnictví indexu a zaplaceného pokusu se student může přihlásit na testování. Vyplní, z kterých modulů chce být testován (na jednom termínu lze absolvovat více modulů) a v některých případech si zvolí platformu (aplikační vybavení), na které bude test vykonávat. Pokud se student na testování nedostaví, budou mu moduly nebo modul uznány s negativním výsledkem a přijde tedy o zaplacený termín. Pokud se řádně z testování omluví, je možné termín prominout a nepřijít tak o zaplacený pokus nebo pokusy.

Po ukončení termínu provede libovolný počet testerů ohodnocení testů a následně zapsání úspěšných modulů do indexů. Studenti si musí sami kontrolovat, zda mají nárok na vystavení certifikátu. Je také důležité zohlednit podniková pravidla. Testy starší 3 let jsou neplatné, indexy mají také omezenou platnost, zpravidla 3 až 5 let (záleží na typu indexu). Pokud student splňuje požadavky na udělení certifikátu, může si o něj požádat. Tester poté zažádá o vystavení certifikační autoritu.

Výhodou konceptu ECDL je možnost přenášet indexy mezi jednotlivými středisky. Znamená to, že je možné absolvovat část modulů v jednom středisku a jinou část v jiném a přesto získat nárok na certifikát.

Je důležité zmínit, že pro fungování testovacího střediska a transparentnost testování je nutné zadávat informace o koupi indexů a certifikátů nebo o konání testovacích termínů do informačního systému WASET³ od společnosti Certion. Jednotlivým termínům, indexům, certifikátům jsou v tomto systému přidělovány identifikátory. Systém se také stará o to, aby všechny termíny byly řádně vypsány a studenti na ně byli řádně přihlášení. Není možné provádět dodatečné změny po uplynutí termínu.

4.5 Administrativní procesy

4.5.1 Studenti

Převážnou část procesů, které budou implementovány v rámci webové aplikace, vykonává akreditovaný tester. To však neznamená, že by nebyla pro uchazeče přínosem, v jejich případě se jedná konkrétně o následující úkony.

- Registrace uchazeče o testování
- Správa osobních údajů
- Objednávky služeb
 - Indexy
 - Jednotlivé pokusy
 - Certifikáty

³ Web Administrated Information System for ECDL Testing

- Balíčky
- Přihlašování na termíny – konkrétní moduly a platformy
- Zobrazení přehledu studia
 - Výsledky testování
 - Nárok na vystavení certifikátů
 - Služby, které čerpá
- Žádosti o vystavení certifikátu

Na základě konzultací s manažerem testovacího střediska ECDL panem Ing. Petrem Jedličkou, Ph.D. a vznesení jeho požadavků bylo rozhodnuto, že budou implementovány všechny výše zmíněné procesy do výsledné webové aplikace. Registrace bude tedy umožněna komukoli, pouze s tím rozdílem, že lidé se statutem studenta platí za jednotlivé služby nižší částku.

Platba za objednané služby nebude řešena v rámci aplikace implementací platební brány, ale po uskutečnění objednávky bude student informován o čísle bankovního účtu, na který danou částku odešle s odpovídajícími údaji. Následně tester zkontroluje, zda byla platba skutečně připsána na účet a uloží potvrzení o platbě do systému.

4.5.2 Testeři

Aplikace má za cíl především zjednodušit práci testerům a automatizovat úkony, u kterých je to možné. Procesy, kterých se zaměstnanci střediska účastní, byly manažerem střediska rozděleny na ty, které budou do aplikace implementovány a zbylé, které implementovat nelze nebo to není nutné. Do první skupiny jsme tedy zařadili následující operace.

- Správa nabízených služeb a produktů (indexy, certifikáty, moduly, programy, platformy a nabízené balíčky)
- Správa ceníku služeb a produktů
- Přijímání a potvrzování objednávek uchazečů
- Validace osobních údajů uchazečů
- Vypisování termínů testování
- Správa termínů (uzavírání termínů pro přihlašování, změny v termínech)
- Zadávání výsledků testů
- Převod zaplacených produktů z jiných certifikovaných středisek
- Zpracování žádosti uchazečů o certifikáty

Zbylé úkony se týkají především interakce s informačním systémem WASET. V současné době není testovacím střediskům poskytováno žádné aplikační rozhraní, které by umožnilo eliminovat manuální práci s tímto systémem. Jedná se o tyto operace.

- Registrace termínů testování do celostátní databáze
- Registrace uchazečů do celostátní databáze
- Generování testů
- Žádosti o vystavení indexů a certifikátů u národního správce

4.6 Shrnutí požadavků

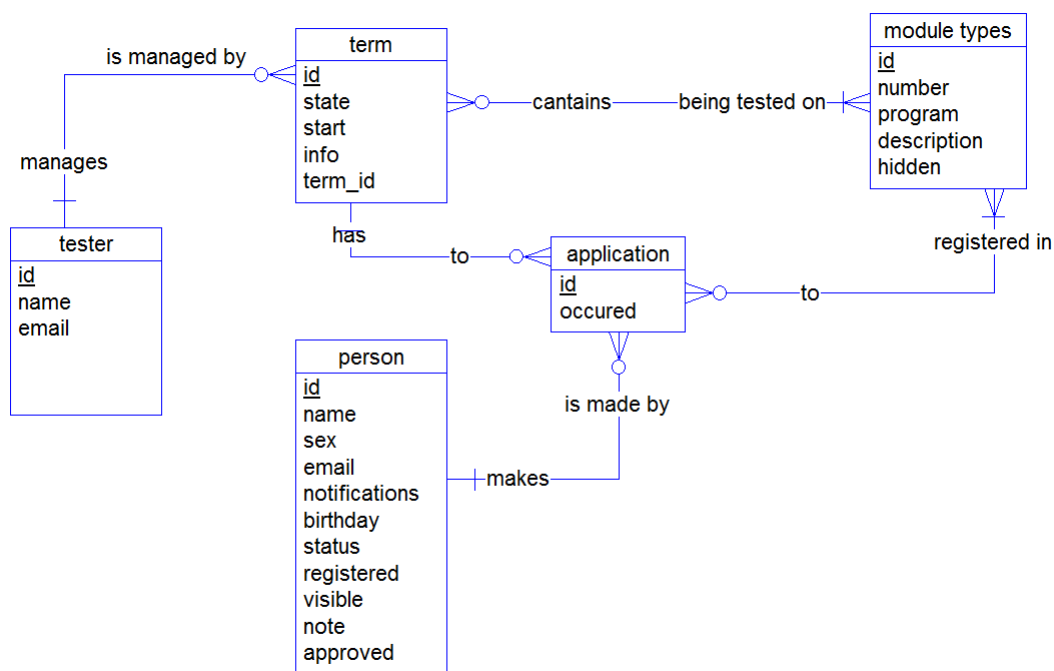
Z uvedených seznamů procesů a úkonů lze vyčíst poměrně vysoké nároky na výslednou webovou aplikaci. V podstatě se z části jedná o implementaci jednoduché podoby e-shopu v kombinaci s informačním systémem vzdělávací instituce, který by se dal svou povahou přirovnat k informačnímu systému naší univerzity. Požadavkem je také umožnit testerům střediska kontrolovat nabízené produkty a služby. To také znamená vyšší nároky na implementaci a na kvalitní návrh celého řešení. Je třeba brát ohled na požadovanou provázanost aplikace od registrace uchazeče, přes objednávku služeb až po samotné testování a vystavení certifikátů a při tom umožnit zaměstnancům střediska kontrolovat a spravovat velkou část obsahu. Cílem je tedy vytvořit moderní a komplexní webovou aplikaci pro podporu vybraných administrativních procesů střediska ECDL vedoucí ke snížení především časových nároků.

5 Návrh webové aplikace

5.1 Konceptuální databázový model

Návrh konceptuálního databázového modelu byl řízen postupy, které ve své knize popisuje Toerey (1999). Bylo nutné popsat problematiku v dostatečném detailu, aby bylo možné definovat jednotlivé databázové komponenty. Důležité jsou dva druhy dat. V první řadě se jednalo o konkrétní vstupy do databáze (natural data requirements), tzn. údaje o studentech, objednávkách, termínech atd. V druhé fázi byly zjištěny výstupy z databáze (processing requirements). Správně definované požadavky na výstupy z databáze jsou kritické pro vhodný návrh celé aplikace. Jedná se o data, která se budou zobrazovat na jednotlivých stránkách a budou tvořit informační výstup z celé aplikace.

Veškerá data souvisela s procesy, které byly definovány na základě konzultací s vedoucím střediska, a jejich popisu byla věnována předchozí kapitola. Výsledkem této analýzy požadavků (requirements analysis) byl entitně relační diagram popisující jednotlivé entity pomocí atributů a vztahů.



Obr. 1 Výřez z ERD obsahující termín a související entity

Na Obr. 1 je vyobrazena pouze část ER diagramu obsahující termín a související entity. Termín má vlastní atributy jako stav (otevřený, uzavřený, uplynulý), datum a čas konání zkoušky, informace pro uchazeče a identifikátor termínu, který

bude přiřazen centrálním střediskem pro ECDL testování v České republice prostřednictvím systému WASET.

Důležité jsou také vztahy mezi entitami a jejich význam. Každý termín tedy musí být spravován právě jedním testerem, musí obsahovat minimálně jeden modul, na který se studenti přihlašují. Pokud se student přihlásí na termín, do databáze bude zaznamenáno, kdy se přihlásil a na jaké konkrétní moduly.

Na základě požadavků bylo u řady entit navrženo inteligentní mazání záznamů, které znamená, že záznam v databázi zůstane, ale nebude se nadále zobrazovat v některých částech aplikace, díky čemuž se může zdát, že je záznam z databáze vymazán. Na Obr. 1 je v případě entity *module types* vidět atribut *hidden*, který bude v naprogramované části využit pro vytvoření efektu vymazání modulu. Výhoda takového řešení je možnost dohledání úprav, které zdánlivě mohly být ztraceny, dále dodržení omezení v podobě cizích klíčů apod. Konkrétně je tento způsob navrhnout a později implementován pro následující entity.

- Moduly
- Certifikáty
- Programy
- Indexy
- Platformy
- Balíčky

Celkově při návrhu volím přístup nemazat záznamy, pokud to není nezbytně nutné nebo to nedává smysl, jako například mazání přihlášek na termín v případě, že si student testování rozmyslí a nechce se termínu zúčastnit.

5.2 Obrazovky uchazeče

5.2.1 Uživatelský účet

Pokud má student zájem účastnit se ECDL testování, musí se nejprve zaregistrovat. Registrace bude umožněna komukoli, kdo má přístup k webovým stránkám střediska prostřednictvím internetu.

V průběhu registrace student zadává osobní údaje a zvolí si přihlašovací jméno a heslo, aby se mohl do aplikace přihlašovat. Student má také možnost zadat číslo indexu, pokud nějaký vlastní, a může tak pokračovat v testování, které započal v jiném středisku. V případě, že se student již zaregistroval, ale zapomněl heslo, může si nechat na emailovou adresu uvedenou ve svém účtu zaslat odkaz, pomocí kterého si heslo změní.

Po úspěšné validaci zadaných údajů je vytvořen účet a student se může přihlásit. Uživateli je umožněno měnit dodatečně všechny vyplněné údaje při registraci, pouze do té doby, dokud nejsou ověřeny pracovníkem střediska ECDL. Po ověření údajů lze měnit pouze kontaktní údaje jako je email, poštovní adresa nebo telefon-

ní číslo. Pro změnu ostatních údajů musí kontaktovat zodpovědnou osobu uvedenou v příslušné části aplikace.

5.2.2 Přehled čerpání služeb

Poté co se student přihlásí, dostane se na obrazovku, kde najde veškeré informace o stavu čerpání zaplacených služeb, objednávek a testování. Konkrétně zde uživatel nalezne, zda mu byly přiřazeny produkty, které si objednal a zaplatil nebo v jakém stavu se jeho objednávky nachází.

Pokud při registraci zadal číslo indexu z jiného střediska, bude tento index zobrazen v tabulce vlastní indexy, kam si prostřednictvím tlačítka pod přehledem může přidat další indexy v případě, že jich vlastní více.

Následuje přehled čerpání služeb, ve kterém se nachází indexy studenta včetně období, ve kterém jsou platné⁴ a informace, zda má index v držení student nebo testovací středisko. Dále v přehledu student nalezne, zda má nárok na zaplacený certifikát, pokud požaduje podrobnější informace jako například, které konkrétní moduly musí splnit, aby na certifikát měl nárok, musí přejít na podrobný stav nároků na certifikáty. Tato část bude také obsahovat informaci o počtu zaplacených pokusů v daném programu, stejně tak jako počet aktivních přihlášek na termín a počet absolvovaných termínů, vše ve vztahu k určitému studijnímu programu.

Poslední částí obrazovky bude přehled stavu testování, kde student v přehledné podobě nalezne seznam všech modulů, které absolvoval nebo na které je přihlášen. U absolvovaných modulů bude obsažena informace, zda úspěšně nebo neúspěšně.

5.2.3 Termíny testování

Jednou z nejdůležitějších částí aplikace je přihlašování studentů na termíny testování. Prostřednictvím menu bude možné přejít na stránku se všemi vypsány termíny, kde student nalezne potřebné informace, jako datum a čas konání testování, na jaké moduly je možné se na daném termínu přihlásit atd. Důležitou informací je také stav termínu, který studenty opravňuje k určitým úkonům. Přihlašovat se a provádět změny v přihláškách je totiž možné pouze v případě, že je termín otevřen.

Při přihlašování se bude kontrolovat, zda má student dostatek zaplacených pokusů v daném programu a bude mít v době testování platný index opět ve vztahu ke studijnímu programu. V takto citlivých operacích, které jsou jádrem aplikace, je nutné využít transakcí poskytovaných InnoDB nezávislým formátem uložení dat.

⁴ Indexy mají definovanou platnost, začínající datem prvního absolvovaného testování, zpravidla 3 roky

5.3 Obrazovky pracovníka střediska

Pracovník střediska se nebude registrovat stejně jako student, ale uživatelský účet mu musí být vytvořen jiným testerem. Takové řešení bylo navrženo na základě konzultací s manažerem střediska především z toho důvodu, že počet testerů je nízký a riziko zneužití pravomoci není vysoké. Celkově je volen přístup, že jeden tester může měnit nastavení ostatních testerů a naopak. V kritičtějších částech aplikace však bude do databáze zaznamenáváno, který tester daný úkon provedl, aby se předešlo zneužívání pravomocí.

5.3.1 Objednávky

Vytvoření objednávky je jedním z prvních úkonů, které musí uchazeč provést, protože bez zaplacených služeb se nelze účastnit testování. Objednávka se po jejím vytvoření dostává do systému, kde vyžaduje řadu úkonů od testera. Obvyklý životní cyklus objednávky vypadá následovně.

1. Vytvoření objednávky studentem
2. Přijmutí objednávky pracovníkem střediska ECDL
3. Student do systému zadá, že objednávku uhradil na bankovní účet střediska
4. Pracovník přiřadí dané produkty uchazeči
5. Poté co se v univerzitním systému SAP objeví potvrzení přijetí platby, zadá tester do systému potvrzení o uhrazení objednávky

Tester může zrušit objednávku pouze v případě, že do té doby nebyly přiřazeny uchazeči odpovídající produkty nebo nebyla potvrzena úhrada objednávky.

Důležitou součástí nabídky produktů a služeb jsou jejich ceny, ty jsou navíc rozdílné pro studenty a pro ostatní uchazeče a mohou se v čase měnit. To je důvodem pro implementaci takového řešení, které umožní pro nové objednávky zobrazovat aktuální cenu, ale u již provedených pracovat s tou, která byla aktuální v době objednání. Pracovník střediska tak může měnit ceny produktů, aniž by ovlivnil již uskutečněné objednávky.

5.3.2 Seznam uchazečů

Na stránce se seznamem uchazečů najde tester přehled všech studentů. Studenty, kteří absolvovali vše, co si zaplatili, ale i jakékoli jiné, je možné skrýt a zjednodušit tak hledání konkrétních studentů v seznamu. Jedná se také o rychlý způsob, jak editovat osobní údaje jednotlivých studentů nebo přejít na stránku s úpravou čerpání služeb uchazečů.

Úprava čerpání služeb je určena především pro převádění stavu testování z jiných středisek, který je možné zjistit ze systému WASET na základě identifikátoru libovolného indexu. Testerům je umožněno přiřazovat indexy, certifikáty, měnit počet zaplacených pokusů a zapisovat do indexů výsledky testování.

5.3.3 Správa termínů testování

Tester se na rozdíl od studenta na termíny nepřihlašuje, ale vypisuje a spravuje je po dobu jejich životního cyklu. Každý termín obsahuje informace o datu a času testování, modulech dostupných pro testování a stav, ve kterém se termín aktuálně nachází. Mimo těchto informací tester zadává při tvorbě termínu jeho identifikátor, který získá ze systému WASET a dále může zvolit libovolný počet testerů, kteří budou informováni v případě, že se na daný termín přihlásí další uchazeč.

Editovat lze veškeré údaje, které se definují při vytváření termínu, aplikují se však některá omezení. Nelze zrušit testování z některého z modulů, pokud je na tento modul někdo ze studentů přihlášen. Datum a čas konání zkoušky je možné změnit pouze u otevřených termínů, a pokud se změní, všichni přihlášení studenti dostanou emailem zprávu o provedené změně. V případě, že se změní stav termínu, může si tester zvolit, zda o tom bude studenty informovat prostřednictvím emailu nebo ne.

Obrazovka se seznamem termínů bude sloužit jako rozcestník k jednotlivým termínům. Tester si může zobrazit detail termínu, kde vidí, kdo je na daný termín přihlášen, na jaké konkrétní moduly je přihlášen a jakou si pro testování zvolil aplikační platformu. Pokud již testování proběhlo a jsou do systému vloženy výsledky testování, může je tester vidět také v detailu termínu. Přímou z přehledu lze také jednotlivé termíny vymazat, na termín ale nesmí být nikdo přihlášen.

5.3.4 Vkládání výsledků

Vkládat výsledky bude možné dvěma způsoby. Hromadně prostřednictvím CSV⁵ souboru se stejnou strukturou, jakou podporuje systém WASET nebo po jednotlivých modulech přímo v aplikaci. Varianta s využitím CSV bude pravděpodobně preferovanou volbou právě díky kompatibilitě se systémem WASET.

Soubor obsahuje především následující informace, které se zpracovávají po nahrání do systému.

1. ID termínu testování
2. Identifikátor indexu studenta
3. Označení testovaného modulu
4. Hodnocení (ano/ne)
5. ID hodnotitele (testera)

Výše uvedené údaje budou porovnány s daty v databázi a poskytnuty testerovi pro kontrolu. Údaje, u kterých nedojde při zpracování ke shodě, budou pro zjednodušení práce odlišeny červenou barvou. Odlišeny budou i výsledky, které byly již v minulosti vloženy do systému. Z výsledků, které mají shodu v databázi a nebyly dosud vloženy do systému, si tester bude moci vybrat ty, které chce uložit a zpracovat tak pouze část dat. Systém umožní práci více testerů současně a je na-

⁵ CSV (Comma Separated Values) – soubor obsahující hodnoty oddělené čárkami

vržen tak, aby nebylo možné vložit jeden výsledek vícekrát. Důležité je opět využití transakcí.

Kromě zmíněných možností lze přihlášku na termín prominout, v takovém případě nebude uchazeči pokus započítán a nepřijde tak o zaplacený pokus. Pokud se student na testování bez omluvy nedostaví, bude možné přímo zadat tuto skutečnost do systému a nebude tak nutné vyplňovat žádné další detaily.

5.3.5 Správa nabízených produktů a služeb

Tester bude mít možnost přidávat a měnit část nabízených produktů a služeb. Na základě konzultací s manažerem střediska ECDL byla dohodnuta implementace řešení pro přidání pouze části typů produktů. Kompletní implementace by znamenala výrazné zvýšení nároků a na čas potřebný pro vývoj. Kompromisem je návrh aplikace do takové podoby, aby v budoucnu nebylo nutné upravovat databázovou strukturu v případě doplnění řešení pro kompletní správu produktů. Pro výslednou aplikaci to tedy znamená, že část produktů je zanesena v databázi a nebude možné přidávat nové prostřednictvím aplikace.

Na základě zkušeností manažera bylo určeno, že nejčastěji může dojít k přidání nové platformy pro testování. Platformou může být operační systém v kombinaci se softwarovým vybavením počítače, jejich nabídku správce ECDL testování pravidelně aktualizuje, aby respektovala technologické trendy. Do systému bude tedy implementována plná podpora správy platformem.

Kromě platformem bude umožněno testerům přidávat nové testovací moduly, protože podobně jako u platformem dochází k rozšiřování nabídky modulů relativně často.

Zbýlé produkty, kterými jsou konkrétně indexy, certifikáty, programy a balíčky nebude možné v rámci aplikace vytvořit. K rozšíření nabídky však u těchto produktů dochází zřídka, a proto není implementace jejich vytváření součástí požadavků na aplikaci. Implementace by byla složitá především z následujících důvodů.

- Indexy se liší tím, že je do nich možné zapisovat pouze předem nadefinované moduly, např. do indexu SELECT programu ECDL Core je možné zapsat výsledky z modulů M2, M3, M7 a 4 dalších libovolných modulů programu ECDL Core. Tester by tedy musel být schopen prostřednictvím aplikace nadefinovat jednotlivá pravidla.
- Podobně jako indexy, vyžadují certifikáty konkrétní moduly nebo počty modulů určitého programu pro splnění nároku. Jedná se znovu o umožnění definovat pravidla.
- Umožnit přidávat nové programy nedává smysl, pokud není umožněno doplnit nabídku indexů a certifikátů.

Pokud tedy nastane situace, kdy bude nutné doplnit nabídku výše zmíněných produktů, bude to vyřešeno přidáním potřebných záznamů přímo do databáze, která je tomu uzpůsobena.

Aplikace tedy bude obsahovat předem nakonfigurovanou nabídku produktů, kterou bude možné v aplikaci upravovat pouze skrytím nebo opětovným zařazením jednotlivých produktů mezi aktivní.

5.3.6 Úkoly

Jedním z požadavků na aplikaci je vytvoření přehledu úkolů, které by měli testeři střediska provést. Tyto informace ve spojení s přehledem objednávek by měli výrazně zjednodušit práci oproti současnému stavu a také předejít možným problémům.

Tester by zde měl najít informace o žádostech o certifikáty, na základě kterých objednává vystavení certifikátů u národního držitele ECDL licence. Díky aplikaci tak tester nebude muset řešit, jestli má uchazeč skutečně na certifikát nárok, protože systém by studentovi požádat o certifikát neumožnil, pokud by na něj nárok neměl.

Přehled chybějících výsledků testování bude také součástí úkolů a zaručuje, že tester opraví veškeré testy, které byly na termínu napsány. Z přehledu se tester bude moci dostat na detail daného termínu nebo přímo na vkládání výsledků.

Následovat bude přehled studentů, u kterých nebyly dosud potvrzené osobní údaje. Potvrzením osobních údajů znemožňuje tester měnit informaci o statutu studenta, jménu a datu narození.

Další část bude obsahovat seznam indexů, kterým chybí doplnit některý z důležitých údajů. Index se zde objeví v případě, že ho uchazeč zadal jako vlastní index vystavený v jiném středisku. Tester v takovém případě musí na základě identifikátoru⁶ indexu zjistit ze systému WASET, o jaký typ se jedná, jaké výsledky jsou v něm zapsány a kdy započalo období jeho platnosti. Druhou možností je, že student si objednal balíček nebo přímo nějaký index a musí být do systému doplněn jeho identifikátor.

Poslední informaci, kterou tester nalezne na stránce s úkoly, je seznam modulů, které nemají přiřazenou žádnou aktivní platformu. Pokud taková situace nastane, nebude možné přihlásit se na testování z daného modulu. Cílem je předejít situacím, kdy tester skryje všechny platformy, které je možné zvolit pro testování a neuvědomí si možné následky.

5.3.7 Statistiky

Posledním požadavkem je možnost jednoduše analyzovat testování ve středisku ECDL na základě definovaného časového období. Po zadání časového okna dostává tester informaci o počtu vypsání termínů, počtu přihlášek na jednotlivé moduly (jeden student může být přihlášen na více modulů na jednom termínu), počtu úspěšných testování, počtu testování s výsledkem nedostavil se a o počtu prominentých testování.

⁶ Každý index má unikátní identifikátor, např. CZ112233

5.4 Technologie

Součástí požadavků na výslednou webovou aplikaci nebyla specifikace technologií, jediným doporučením bylo využití MySQL nebo PostgreSQL jako systému řízení báze dat. Cílem tedy bylo zvolit vhodné nástroje pro tento typ webové aplikace.

5.4.1 Databázový systém

Ačkoliv se MySQL a PostgreSQL liší, pro webové aplikace menších rozměrů nemůže být ani jedna varianta špatnou volbou. Rozdíly jsou nejvíce znát při větších objemech dat nebo při nutnosti využití procedurálního programování. Ani jeden z těchto případů neplatí pro webovou aplikaci pro středisko ECDL. Vše by mohla rozhodnout finanční stránka obou variant, ale v obou případech se jedná o open-source řešení, které je aktivně ve vývoji s dostatečnou podporou komunity. Čistě z důvodů větší zkušenosti se systémem byl zvolen MySQL. Zvolený SŘBD disponuje řadou nezávislých formátů uložení dat, z kterých byl vybrán InnoDB především z důvodu podpory transakcí, které budou v implementační fázi značně využívány.

5.4.2 Programovací jazyk

Důležitou volbou je programovací jazyk, ve kterém bude webová aplikace napsána. Nabízí se jazyk PHP, který je de facto standardem při tvorbě webových aplikací a velká část webů je napsána právě v tomto jazyce. Pro PHP hovoří řada argumentů, jedná se o open-source technologii, která je kontinuálně ve vývoji, PHP má také velkou komunitu, což je znát i na vysokém počtu frameworků postavených právě nad tímto jazykem. Velice důležitý je také ekonomický aspekt této volby, kromě toho, že se jedná o open-source je také důležité zmínit, že zvláště v České republice, ale i v zahraničí, se vyskytuje velké množství PHP programátorů, což v důsledku znamená nižší náklady na případné úpravy a rozšíření aplikace. Na základě těchto argumentů byl zvolen programovací jazyk PHP jako ten, ve kterém bude aplikace naimplementována. Momentálně se nachází ve verzi 5.6.15 a neustále se vyvíjí. PHP verze 7 je ve fázi RC⁷, verze 6 nakonec nikdy nebyla vydána (The PHP Group, ©2001-2015).

Kombinací PHP a MySQL výborně doplňuje HTTP (webový) server Apache, který je také volně dostupný. Apache jako projekt vznikl v roce 1995 a od té doby se na jeho vývoji podílely stovky vývojářů, kteří i v současnosti přidávají novou funkcionalitu, opravují chyby a tvoří dokumentaci (The Apache Software Foundation, ©1997-2015).

Inspirace a podněty k řešení problémů byly čerpány z knihy „Web Programming with PHP and MySQL: A Practical Guide“ od Bramera (2015), který se na příkladech věnuje spojení jazyka PHP a databázového systému MySQL. Stejněmu tématu se také věnuje Boronczyk (2009), konkrétně pak základům jazyka SQL na úrovni jednotlivých dotazů, správou databázového systému, návrhu databázové

⁷ RC - release candidate – verze, která má na to stát se finální verzí produktu

struktury aplikace a v neposlední řadě propojením samotné aplikace napsané v jazyce PHP a databázového systému MySQL pomocí knihovny PDO.

Webové stránky Jakuba Vrány, který je v současnosti jedním z autorů dokumentace pro jazyk PHP (Vrána, ©2005-2015), byly také častým zdrojem potřebných informací. Autor v roce 2010 také publikoval knihu o PHP s názvem „1001 tipů a triků pro PHP“, která je považována za užitečný zdroj informací, v ní se věnuje praktickým ukázkám např. syntaxi PHP, bezpečnosti webových aplikací atd. (Vrána, 2010).

5.4.3 Framework

V případě implementace v jazyce PHP se nabízí řada frameworků, které umožňují zjednodušit implementaci a částečně vynutit vhodnou stavbu aplikace. Pro implementaci byl vybrán Nette Framework, který je díky českým kořenům používán především na území České a Slovenské republiky, kde má početnou komunitu uživatelů. Pro tuto volbu hovoří opět to, že se jedná o open-source řešení otestované na velkém množství produkčních aplikací. Nette také pomáhá zabezpečit aplikaci proti známým způsobům útoků, které ve své práci popisuje Tresner (2007), jedná se například o XSS⁸, CSRF⁹, session hijacking atd. (Nette Foundation, © 2008, 2015). Ne však všechny bezpečnostní hrozby lze ošetřit na úrovni frameworku. S ohledem na bezpečnost je nutné psát celou aplikaci a vyvarovat se tak například SQL injection jak popisuje Tresner (2007).

Jednou z výhod použití frameworku je také nutnost psát aplikaci takovým způsobem, aby respektovaly MVC¹⁰ architekturu, což znamená rozdělení aplikace do tří logických celků – model, view a controller. Výsledný kód aplikace je poté přehlednější a je jednodušší provádět dodatečné změny nebo aplikaci rozšiřovat (Reenskaug, 2009).

5.4.4 Frontend

Se vznikem frontend frameworků se výrazně zjednodušila práce na vzhledu webových aplikací. Použití frameworku dává smysl u projektů, na kterých nepracuje webový designer a vzhled aplikace by tak mohl zhoršovat dojem, přestože je backendová část aplikace dobře navrhnutá a naimplementována.

Vzhledem k tomu, že důraz této práce je kladen na funkční stránku, bude využito frontend frameworku a to konkrétně jednoho z nejznámějších Bootstrapu. Jedná se o kvalitní framework, který si klade za cíl zjednodušit vývoj responzivních webových aplikací. Disponuje robustním grid systémem, ulehčující pozicování jednotlivých HTML elementů. Součástí frameworku jsou také JavaScript knihovny, přinášející dynamičnost (Otto, 2010). Přestože je Bootstrap určen pro vytváření

⁸ XSS - Cross-site scripting

⁹ CSRF - Cross-site request forgery

¹⁰ MVC - Model-view-controller

aplikací pro všechny platformy od mobilů po počítače, nebude cílem vytvořit mobilní verzi celé aplikace.

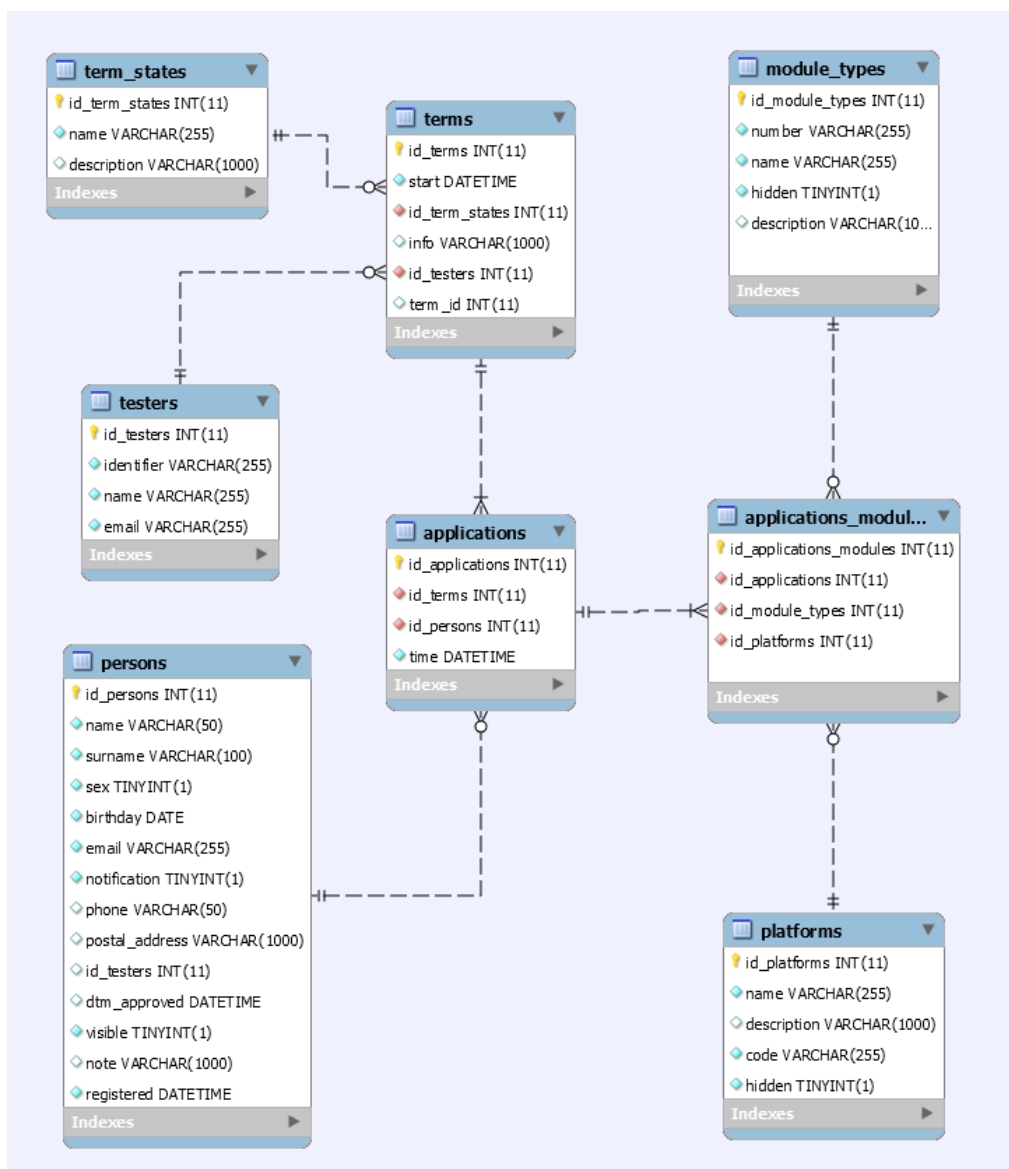
5.4.5 Dokumentace

Požadavkem na bakalářskou práci je také vytvořit programovou dokumentaci k výsledné aplikaci. Nette foundation, Votruba, Hanslík a Nešpor jsou autoři volně dostupného nástroje pro tvorbu profesionální dokumentace určeného přímo pro jazyk PHP. Apigen umožňuje dokumentovat třídy, funkce a konstanty, dále má podporu GoogleAnalytics a množství dalších možností úprav (Votruba, Hanslík, Nešpor, Nette Foundation, 2015). Alternativou pro tvorbu dokumentace přímo z programového kódu je nástroj s názvem PhpDocumentor, který umožňuje generovat dokumentaci kódu napsaného například v Zend frameworku nebo Symfony. Stejně jako Apigen umožňuje prohlížení tzv. Todo úkolů (Eichorn, 2014). Vzhledem k přímé podpoře Nette frameworku budu dokumentaci generovat s použitím nástroje Apigen.

6 Implementace a výsledky

6.1 Správa termínu testování

Na základě konceptuálního databázového modelu a návrhu stránek pro správu termínů byl vytvořen fyzický datový model. Jednotlivé entity včetně atributů jsou uvedeny na Obr. 2.




Obr. 2 Výřez fyzického datového modelu popisující termín a související entity

Ve fyzickém modelu se již objevují cizí klíče a M:N vztahy jsou rozděleny na vztahy typu 1:N. Každý termín musí mít přiřazen právě jeden stav, naopak každý stav mů-

že a nemusí být přiřazen k jednomu nebo více termínům. Každý termín musí mít právě jednoho vedoucího testování, tedy testera.

Uchazeči se hlásí nejenom na daný termín testování, ale také na libovolný počet vypsaných modulů na termínu. Entita *applications* pak reprezentuje přihlášku daného studenta na určitý termín a je doplněna o atribut vyjadřující čas uskutečnění přihlášky. Společně v rámci jedné databázové transakce však vzniká jeden nebo více záznamů do entity *applications_module_types*, která spojuje informaci o tom, na který z modulů se uchazeč hlásí a jakou si pro testování zvolil platformu. Všechny tyto atributy jsou povinné a jsou vynucené jak v rámci formulářů v aplikaci tak omezeními (constraints) v databázi.

Mendelova univerzita v Brně
Testovací středisko ECDL CZ136

 Akreditované středisko pro ECDL testování
 [Seznam uchazečů](#)
[Správa termínů a testování](#)
[Objednávky služeb](#)
[Úkoly](#)
[Statistiky](#)
Ing. Petr Slaný, PhD.

Termin testování

ID Terminu	18258
*Datum zkoušky	27. 10. 2015
*Čas zkoušky	14:00
Vedoucí testování:	CZT0682 Ing. Petr Slaný, PhD.
Stav termínu:	uplynulý
Zasílání e-mailů o nových přihláškách:	CZT0682 Ing. Petr Slaný, PhD.
Informovat emailem účastníky o změně v termínu	<input checked="" type="checkbox"/>
*Při změně času termínu budou přihlášení studenti informováni vždy prostřednictvím emailu	
Uložit stav termínu	

Moduly na termínu

Z termínu lze smazat pouze moduly, na které není nikdo přihlášen.

Modul	Počet přihlášek	Smazat
M1 - Základní pojmy informačních a komunikačních technologií	0	Smazat
M4 - Práce s tabulkami	0	Smazat
M5 - Použití databází	1	Smazat
M6 - Prezentace	1	Smazat
M7 - Základy práce s internetem a komunikace	1	Smazat

Přidat další moduly

Přidávat lze pouze moduly s přiřazenými platformami.

M2 - Základy práce s počítačem a správa souborů
 M3 - Zpracování textu
 AM3 - Pokročile zpracování textu
 AM4 - Pokročilá práce s tabulkami
 AM5 - Pokročile použití databází
 AM6 - Pokročile...

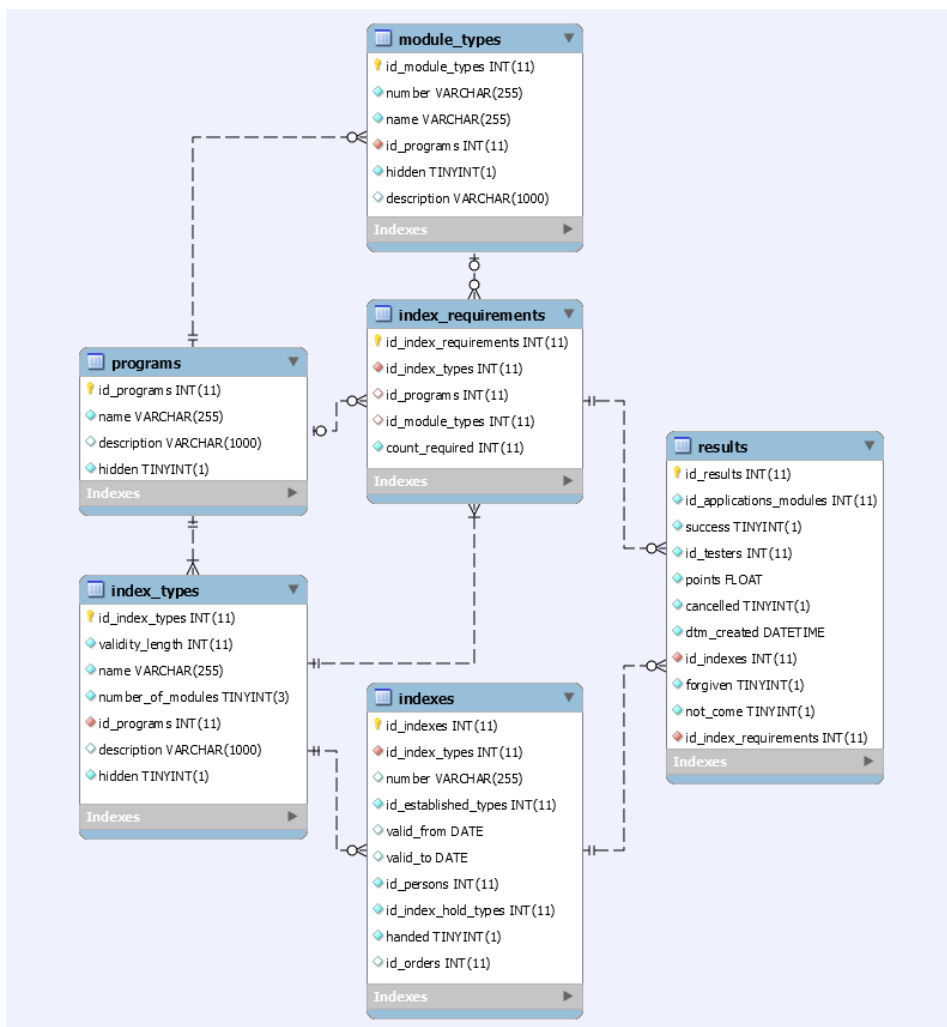
[Přidat moduly](#)

Obr. 3 Obrazovka se správou vypsaného termínu

V levé části obrazovky je možné upravovat details termínu včetně seznamu testů, kterým má chodit na email upozornění v případě nových přihlášek. V pravé části je možné upravovat nabídku modulů pro testování. Odstranit lze pouze moduly, na které není nikdo přihlášen. Pokud student odstraní modul z nabídky je z databáze získána informace o počtu přihlášených studentů a pouze v případě, že je jejich počet nulový, je modul skutečně odstraněn. Vše probíhá opět v transakci, která zaručuje korektní provedení celé operace.

6.2 Indexy a výsledky

Do každého indexu je možné zapisovat pouze úspěšné výsledky, které odpovídají definovaným pravidlům. Struktura databáze musí být uzpůsobena těmto požadavkům a především samotná aplikace musí být schopna číst tyto pravidla a aplikovat na jednotlivé požadavky na vložení výsledku.



Obr. 4 Výřez fyzického datového modelu popisující vztahy mezi indexy a výsledky

Na Obr. 4 je vyobrazena část fyzického datového modelu popisující právě vztahy mezi indexy a výsledky. V tabulce *index_requirements* jsou pro každý index nadefinovaná pravidla, podle kterých je možné vkládat výsledky zkoušek z modulů. Aplikace pracuje se třemi možnostmi.

1. Přímo je definován konkrétní modul (*id_module_types*) pro daný typ indexu (*id_index_types*).

2. V atributu *id_programs* je definován program a v *count_required* je definován počet. Význam je, že do indexu lze zapsat definovaný počet modulů daného programu, ale nesmí se jednat o moduly, které jsou definovány prvním pravidlem.
3. Pokud má program a typ modulu hodnotu NULL a je definován počet požadovaných modulů, tak to znamená, že lze do indexu zapsat daný počet libovolných modulů bez ohledu na program, do kterého patří. Nesmí se znovu jednat o moduly, které jsou definovány prvním pravidlem.

Při každém požadavku na vložení výsledku pak probíhá validace, zda je možné do zvoleného indexu výsledek zapsat a až poté je akce provedena. Stejná pravidla se aplikují pro validaci uchazečova nároku na certifikát. Konkrétní definice se nachází v tabulce *certificate_requirements*. Při požadavku na zjištění nároku na určitý certifikát se namapují všechny úspěšné výsledky, které nejsou starší tří let a jsou zapsány do stále platných indexů, na pravidla definována v databázi. Třída *Certificate* se pak postará o kontrolu, že jsou všechny podmínky splněny a výsledky vrátí připravené pro vykreslení.

6.3 Emailová komunikace

Systém na základě některých úkonů potřebuje komunikovat se studentem a testerem i v případě, že právě nepracují s aplikací. To je důvodem pro implementaci řešení pro odesílání elektronické pošty.

K veškeré komunikaci ven ze systému se používá třída *MailService*. Odesílání emailů umožňuje nadstavba Nette frameworku, která je postavena nad funkcí mail, která je nativně součástí PHP.

Díky tomu, že komunikace probíhá přes jedno rozhraní v podobě *MailService* není složité překonfigurovat odesílání emailů a využívat SMTP¹¹ server.

6.4 Dokumentace

Součástí požadavků na implementaci bylo i vytvoření programátorské dokumentace. Ačkoli se jedná o velice vývojáři neoblíbenou činnost, výhody vytvoření dokumentace jsou zřejmé. Nejvhodnější přístup je dokumentovat části kódu přímo při jeho psaní, protože to umožňuje odhalit chyby v návrhu, příkladem je příklad, kdy metoda provádí příliš komplikované operace, které by bylo vhodné spíše rozdělit do více metod.

V práci byly dokumentovány jednotlivé třídy, jejich proměnné a metody. Výsledná dokumentace byla poté vygenerována nástrojem Apigen, ale nebyl by problémem použít jiný nástroj, jako například phpDocumentor. Nejznámější nástroje vy-

¹¹ STMP (Simple Mail Transfer Protocol) - internetový protokol určený pro přenos zpráv elektronické pošty

užívají stejné standardy pro názvy dokumentačních tagů, proto není problémem přechod na jiný software.

```
<?php
/**
 * Třída reprezentující certifikát. Jedná se o potomka třídy Entity.
 */
class Certificate extends Entity{
    /**
     * Metodá vrátí pole certifikátů včetně informace
     * o nároku a namapovaných výsledků na požadavky
     * pro získání certifikátu ve vztahu k danému uchazeči.
     * @param DibiConnection $db připojení k databázi
     * @param int $personId id studenta
     * @return array
     */
    public static function getStatus($db, $personId){...}
    ...
}
```

Na ukázce je zřejmé, jakým způsobem se programátorská dokumentace tvoří. Klíčovým symbolem pro parser je lomítko následované dvěma hvězdičkami. Pro vytvoření bohatší a přehlednější dokumentace je využíváno speciálních tagů, které jsou tvořeny zavináčem a klíčovým slovem. Nejčastěji se tagy používají k definování parametrů, návratových hodnot, označení abstraktních tříd, deprecated tříd a metod, atd.

Overview Namespace **Class**

Class Certificate

Třída reprezentující certifikát.

Entity
└ Certificate

Located at <model/entities/Certificate.php>

Methods summary

public static array	<code>getStatus(DibiConnection \$db, integer \$personId)</code> Metoda vrátí pole certifikátů včetně informace o nároku a namapovaných výsledků na požadavky pro získání certifikátu ve vztahu k danému uchazeči. Parameters <code>\$db</code> připojení k databázi <code>\$personId</code> id studenta Returns array
public static boolean	<code>validate(DibiConnection \$db, integer \$personId, integer \$certificateId)</code> Metoda zvaliduje a vrátí informaci, zda má student nárok na určitý certifikát. Parameters <code>\$db</code> připojení k databázi <code>\$personId</code> id studenta <code>\$certificateId</code> id studentova certifikátu Returns boolean výsledek, zda má daný uchazeč nárok na certifikát

Methods inherited from Entity

`__construct()`, `__get()`, `__isset()`, `__set()`, `getData()`, `logChange()`, `logNew()`

Obr. 5 Ukázka z vygenerované programátorské dokumentace

Na Obr. 5 je vyobrazena ukázka vygenerované dokumentace. Nástroj rozparsuje zdrojový kód aplikace, najde všechny třídy a jejich vlastnosti a přiřadí jim doplňnou dokumentaci. Není nutné generovat dokumentaci pro všechny soubory, stačí zvolit libovolné složky nebo soubory. Do cílové složky pak nástroj vygeneruje jednotlivé soubory obsahující validní html kód, potřebné javascriptové soubory a soubory s kaskádovými styly. Přes soubor index je poté možné zobrazit celou dokumentaci, která podporuje například i vyhledávání.

7 Závěr

Práce se zabývala implementací webové aplikace pro podporu vybraných administrativních procesů střediska ECDL. V první fázi bylo nutné seznámit se se samotným konceptem European Computer Driving Licence známém po celém světě. Následovaly první konzultace s manažerem střediska a seznámení se s jednotlivými administrativními procesy. Část práce byla také věnována analýze fungování ostatních středisek, především pak tomu, zda mají některá z nich implementováno vlastní řešení pro podporu administrativních procesů.

Na základě prvotního návrhu požadavků byl pomocí analýzy požadavků (requirements analysis) vytvořen konceptuální datový model, který byl iterativně na základě dalších konzultací upravován až do finální podoby. Součástí dodaných požadavků byl i hrubý návrh nejdůležitějších obrazovek aplikace, na základě kterého byl vytvořen statický návrh vzhledu aplikace s využitím Bootstrapu. Před samotnou implementací proběhla volba technologií vhodných pro tento typ aplikace. Dále se práce věnuje samotné implementaci, doplněné o tvorbu programátorské dokumentace. V průběhu implementace byl kladen důraz na testování funkcionality a bezpečnosti celé aplikace.

Výsledná aplikace má nahradit původní zcela neautomatizované řešení. Přístup k ECDL testování na Mendelově univerzitě v Brně by se tak měl výrazně zjednodušit. Vzhledem k absenci podobných řešení u ostatních středisek, se dá považovat implementace tohoto systému jako důležitá konkurenční výhoda, ale také jako příležitost. Pokud by se provoz systému v praxi dostatečně osvědčil, bylo by možné uvažovat o rozšíření systému mezi další testovací střediska. Fungování všech středisek musí být stejné, protože střediska musí dodržovat stejná pravidla vynucována národním, ale i nadnárodním správcem konceptu.

Věřím, že umožnění registrace komukoli, kdo má přístup k internetu může zvýšit zájem o testování a tím pádem i zisky. Implementace systému pak může mít za následek větší spokojenost uchazečů s poskytovanými službami. Složitost registrace a přihlašování na termíny mohla zcela jistě některé potenciální uchazeče odradit.

Díky aplikaci se zjednoduší správa testování nejen studentům, ale hlavně pracovníkům střediska ECDL. Velká část nároků je přenesena na aplikaci. Tester může vypisovat termíny, sledovat seznam objednávek a stránku s úkoly a o zbytek se postará aplikace. Pokud tester obdrží žádost o vystavení certifikátu, může si být jist, že na něj má student skutečně nárok, protože kontrolu za něj provádí aplikace. Systém testera také upozorní na dosud nezadané výsledky testování. Vše vede ke zjednodušení, zrychlení práce a vyšší spokojenosti uchazečů.

Při implementaci byl kladen důraz na možnost přidání další funkcionality, která nebyla součástí požadavků na tuto práci. Bylo by tedy možné rozšířit aplikaci o možnost přidávat nové typy indexů a certifikátů nebo přímo celého programu. S tím souvisí možnost definovat požadavky na získání certifikátu a předpisy pro nové indexy.

Vzhledem k využití Bootstrapu by bylo možné upravit aplikaci pro zobrazení na mobilních zařízeních, což by umožnilo ještě více rozšířit dostupnost vytvořené aplikace.

Největší příležitostí je možnost propojení aplikace s národním systémem WASET pomocí aplikačního rozhraní (API). V současné době však žádné rozhraní není k dispozici i přes to, že implementace takového rozhraní by znamenala další snížení nároků na testery a celkové zjednodušení. Spojení systémů by eliminovalo potřebu synchronizovat ručně identifikátory indexů, termínů a testerů. Webová aplikace by také mohla automaticky získávat potřebné údaje v případě, že si uchazeč přenesl index z jiného testovacího střediska a má zájem pokračovat v testování.

8 Literatura

- BORONCZYK, T. A KOL. *PHP 6, MySQL, APACHE: vytváříme webové aplikace*. Brno: Computer press, 2009. 816 s. ISBN 978-80-251-2767-4.
- BRAMER, MAX. *Web Programming with PHP and MySQL: A Practical Guide*. Springer, 2015.
- CEPIS. *Council of European Professional Informatics Societies - About Us* [online]. Europe: Council of European Professional Informatics Societies, ©2015 [cit. 2015-11-22]. Dostupné z: <http://www.cepis.org/index.jsp?p=636&n=637>
- CSAPO, NANCY. *Certification of computer literacy: The international computer driving license provides standards that let individuals participate in a global digital society*. THE Journal (Technological Horizons In Education), 2002, 30.1: 46.
- ČESKÁ SPOLEČNOST PRO KYBERNETIKU A INFORMATIKU. *Česká společnost pro kybernetiku a informatiku: Obecně o ECDL* [online]. Česká republika: ČSKI [cit. 2015-11-22]. Dostupné z: <http://www.cski.cz/main.php?id=01.05.02.01>
- ECDL - EUROPEAN COMPUTER DRIVING LICENCE - ČZU. *ECDL - European Computer Driving Licence - ČZU* [online]. Praha, ©2001-2011 [cit. 2015-11-25]. Dostupné z: <http://ecdl.czu.cz/>
- ECDL: ECDL CZECH REPUBLIC. *ECDL: O konceptu ECDL* [online]. ČSKI, ©1999-2015 [cit. 2015-11-24]. Dostupné z: http://www.ecdl.cz/o_projektu.php
- ECDL: ECDL CZECH REPUBLIC. *ECDL: Sylaby a moduly* [online]. ČSKI, ©1999-2015 [cit. 2015-11-24]. Dostupné z: <http://www.ecdl.cz/sylaby.php>
- ECDL FOUNDATION. *About Us - ECDL Foundation* [online]. Švédsko: ECDL Foundation, ©1997- [cit. 2015-11-21]. Dostupné z: <http://www.ecdl.org/index.jsp?p=93&n=94>
- EICHORN, J. *PhpDocumentor*. [online]. 2000-2015. [cit. 2015-12-19]. Dostupné z: <http://www.phpdoc.org/>.
- ILOMÄKI, LIISA, ET AL. *What is digital competence*. Linked portal. Brussels: European Schoolnet (EUN), 2011, 1-12.
- LANÍK, PETR. *Webová podpora pro ECDL* [online]. Ostrava, 2014 [cit. 2015-11-25]. Bakalářská práce. Ostravská univerzita v Ostravě, Přírodovědecká fakulta. Vedoucí práce RNDr. Martin Žáček, Ph.D. Dostupné z: <http://theses.cz/id/gm3qpi/>.
- MARK OTTO, JACOB THORNTON, AND BOOTSTRAP CONTRIBUTORS: *Bootstrap - The world's most popular mobile-first and responsive front-end framework* [online]. Mark Otto, Jacob Thornton, and Bootstrap contributors, 2010 [cit. 2015-12-22]. Dostupné z: <http://getbootstrap.com/>
- NETTE FOUNDATION. *Rychlý a pohodlný vývoj webových aplikací v PHP | Nette Framework* [online]. © 2008, 2015 [cit. 2015-11-27]. Dostupné z: <http://nette.org/>.

- PORTÁL ECDL. *Portál ECDL* [online]. [cit. 2015-11-24]. Dostupné z: <http://www.pefka.mendelu.cz/ecdl/>
- REENSKAUG, TRYGVE; COPLIEN, JAMES O. *The DCI architecture: A new vision of object-oriented programming. An article starting a new blog:(14pp)* http://www.artima.com/articles/dci_vision.html, 2009.
- TEOREY, TOBY J. *Database modeling & design*. Morgan Kaufmann, 1999.
- THE APACHE SOFTWARE FOUNDATION. *The Apache HTTP Server Project* [online]. ©1997-2015 [cit. 2015-11-26]. Dostupné z: http://httpd.apache.org/ABOUT_APACHE.html.
- THE PHP GROUP. *PHP: Hypertext Preprocessor* [online]. ©2001-2015, 2015-11-26 [cit. 2015-11-26]. Dostupné z: <http://php.net/>.
- TRESNER, M. *Bezpečnost a testování webových aplikací*. Diplomová práce. MZLU v Brně, 2007.
- VOTRUBA T., HANSLÍK J., NEŠPOR O., NETTE FOUNDATION *ApiGen / Smart and Readable Documentation for your PHP project*. [online]. 2010-2015. [cit. 2015-12-19]. Dostupné z WWW: <<http://apigen.org/>>.
- VRÁNA, J. *PHP triky - Weblog o elegantním programování v PHP pro mírně pokročilé* [online]. ©2005-2015 [cit. 2015-11-26]. Dostupné z: <http://php.vrana.cz/>.
- VRÁNA, J. *1001 tipů a triků pro PHP*. 1. vyd. Brno: Computer Press, 2010. 456 s. ISBN 978-80-251-2940-1.