

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Informační systém jazykové školy



2015

Vedoucí práce:
Mgr. Arnošt Večerka, Ph.D.

Ing. Martin Křepinský

Studijní obor: Aplikovaná informatika,
kombinovaná forma

Bibliografické údaje

Autor: Ing. Martin Křepinský
Název práce: Informační systém jazykové školy
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2015
Studijní obor: Aplikovaná informatika, kombinovaná forma
Vedoucí práce:
Mgr. Arnošt Ve-
čerka, Ph.D.
Počet stran: 48
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Ing. Martin Křepinský
Title: Information System for Language School
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Pa-
lacký University Olomouc
Year of defense: 2015
Study field: Applied Computer Science, combined form
Supervisor:
Mgr. Arnošt Ve-
čerka, Ph.D.
Page count: 48
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Předmětem bakalářské práce je navrhnout a implementovat informační systém pro jazykovou školu. Informační systém jsem vytvořil ve spolupráci s jazykovou školou Enara s.r.o. v Jihlavě. Práce se skládá z těchto hlavních částí: z analytické části, určení funkcí informačního systému, popisu technického řešení a popis uživatelského rozhraní informačního systému. Analytická část obsahuje analýzu prostředí jazykové školy Enara s.r.o., především procesy a organizační strukturu, a analýzu požadavků na informační systém. Z provedené analýzy vychází určení funkcí informačního systému a návrh technického řešení. V popisu uživatelského rozhraní je uveden způsob ovládání informačního systému. V závěru práce jsou uvedeny možnosti dalšího rozvoje informačního systému jazykové školy.

Synopsis

The goal of the thesis is design and implementation of an information system for language school. I created information system in cooperation with the language school Enara s.r.o in Jihlava. The thesis consists of these main parts: an analytical part, description of a technical solution and description of user interface. The analytical part contains analyse of the language school environment Enara s.r.o., mainly processes and organization structure, and user requirements analysis. On the bases of the analytical part I set up functions of the information system an proposed technical solution. The description of the user interface deals with the manipulation with the information system. At the end I stated possibilities of further development of the information system

Klíčová slova: informační systém; závěrečná práce; jazyková škola;

Keywords: information system; thesis; language school

Děkuji mému vedoucímu Mgr. Arnoštovi Večerkovi, Ph.D. za rady při zpracování bakalářské práce.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

1	Úvod	8
2	Analýza současného stavu	8
2.1	Organizační struktura	9
2.2	Procesy	9
2.2.1	Proces zakázek	9
2.2.2	Proces výuky	10
2.2.3	Proces fakturace	11
2.3	Zaměstnanci	12
2.4	Technologie - Informační systém	14
3	Určení funkcí nového informačního systému	14
3.1	Funkce IS - zakázky	15
3.2	Funkce IS - výuka	15
3.3	Funkce IS - fakturace	16
4	Analýza případů užití IS	16
4.1	Administrativa	17
4.2	Lektoři	17
4.3	Studenti	18
5	Architektura informačního systému	18
6	Technologie	20
7	Technické řešení	21
7.1	Aplikace prezentační vrstvy	21
7.1.1	Webová aplikace Student	21
7.1.2	Webová aplikace Lektor	22
7.1.3	Aplikace Administrativa	22
7.2	Aplikace serverová část	25
7.3	Databáze	26
8	IS- uživatelská část	26
8.1	Webová aplikace Student	26
8.1.1	Modul Kurzy	26
8.1.2	Modul Testy	27
8.2	Webová aplikace Lektor	27
8.2.1	Modul Třídnice	28
8.2.2	Modul testy	30
8.2.3	Modul Testy Kontrola	30
8.2.4	Modul Přehled výuky	30
8.3	Aplikace Administrativa	31
8.3.1	Modul Adresář	31

8.3.2	Modul Zakázky	33
8.3.3	Modul Kurzy	33
8.3.4	Modul Výuka	35
8.3.5	Modul Absence	36
8.3.6	Modul Změny ve výuce	36
8.3.7	Modul Rozvrh	37
8.3.8	Modul Testy	38
8.3.9	Modul Plány	39
8.3.10	Modul Náповěda	39
9	Další možnosti rozvoje systému	39
	Závěr	41
	Conclusions	43
	Bibliografie	45
	A První příloha	46
	B Obsah přiloženého CD/DVD	47
	Seznam zkratk	48

Seznam obrázků

1	Organizační struktura společnosti	9
2	Proces zakázek	10
3	Proces výuky	11
4	Proces fakturace	12
5	Případy užití Administrativa	18
6	Případy užití Lektori	19
7	Případy užití Studenti	19
8	Schéma databáze	26
9	Modul Testy	28
10	Aplikace Lektor - zápis výuky.	29
11	Aplikace Administrativa - Adresář.	32
12	Aplikace Administrativa - Kurzy	34
13	Aplikace Administrativa - Výuka	36
14	Aplikace Administrativa - Rozvrh	37

Seznam tabulek

1	Proces zakázky – popis činností	10
2	Proces výuky – popis činností	11
3	Proces fakturace – popis činností	12
4	RACI - proces zakázek	15
5	RACI - proces výuka	16
6	RACI - fakturace	17

Seznam vět

Seznam zdrojových kódů

1	AS3 - volání funkce na serveru.	21
2	AS3 - zpracování přijatých dat.	21
3	Konstruktor třídy DbService	23
4	volání funkce na serveru	24

1 Úvod

Řízení procesu výuky ve vzdělávacích organizacích mimo formální vzdělávací instituce má řadu specifík, které mají základ především v individuálních požadavcích na výuku, kde obsah a průběh výuky je definován zejména požadavky zákazníka (studenta) a ne formálními vzdělávacími programy. Dalším faktorem, který ovlivňuje řízení procesu výuky v jazykové škole, je rozložení výuky do delšího časového období (většinou roku) než v případě kurzů, ve kterých je výuka organizována v blocích. Tato skutečnost má dopad především na větší počet změn, ke kterým dochází v jazykových kurzech a z toho vyplývajících požadavků na evidenci změn.

Proces výuky v jazykové škole se skládá z řady činností, které zohledňují výše uvedená specifika, a jejich efektivita je ovlivněna kvalitou a dostupností informací potřebných k jejich realizaci. Efektivita realizace těchto aktivit tedy úzce souvisí s informačním systémem, který zajistí dostupnost potřebných informací.

Hlavním cílem mé závěrečné práce je vytvořit informační systém, který poskytne podporu pro řízení procesu vzdělávání v jazykové škole a bude zohledňovat výše uvedená specifika. Při vytváření systému jsem spolupracoval s jazykovou školou Enara s.r.o. se sídlem v Jihlavě.

Cílem je vytvoření programu, který umožní evidenci informací vztahujících se k procesu výuky v jazykové škole a výměnu informací mezi jeho účastníky. Tento informační systém by měl přispět k vyšší efektivitě výuky v jazykové škole.

2 Analýza současného stavu

Hlavní problémy jazykové školy v současnosti se týkají těchto oblastí:

- Studenti nemají přístup k informacím o výuce.
- Není podpora pro průběžné testování studentů.
- Neefektivní řešení změn ve výuce především suplovaných hodin.
- Neexistuje elektronická evidence zakázek.
- Chybí generování rozvrhů učeben a lektorů což negativně ovlivňuje přípravu kurzů a řešení změn ve výuce.

V analýze současného stavu jsem zaměřil na čtyři klíčové oblasti, které mají rozhodující vliv na efektivitu organizace a to:

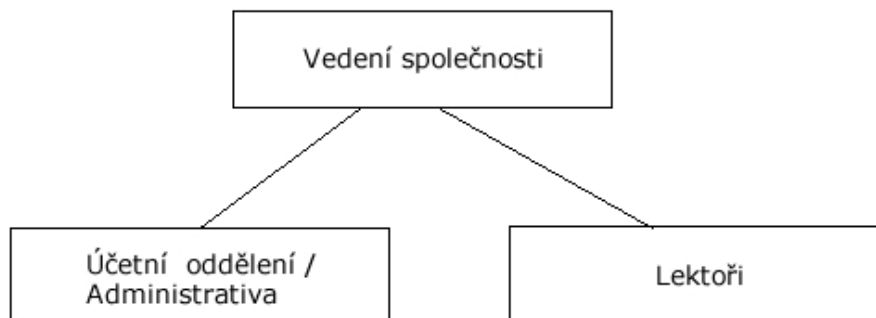
1. Organizační struktura
2. Procesy

3. Lidé

4. Technologie - Informační systémy

2.1 Organizační struktura

Organizační struktura společnosti je liniová. Role jednotlivých organizačních jednotek jsou definovány rozsahem činností a odpovědností za jejich realizaci.



Obrázek 1: Organizační struktura společnosti

Vedení společnosti

Vedení společnosti odpovídá především za stanovení a realizaci strategie, přípravu nabídek, komunikaci se zákazníky, uzavírání zakázek a kontrolní činnost.

Účetní oddělení/ Administrativa

Oddělení odpovídá za kontrolu správnosti účetnictví společnosti. Hlavní činnosti jsou účtování a fakturace. Mezi další činnosti patří především spolupráce na organizaci kurzů, komunikace s lektory, kontrola správnosti podkladů pro fakturaci (především počty odučených hodin) a spolupráce s vedením společnosti na přípravě zakázek.

Lektoři

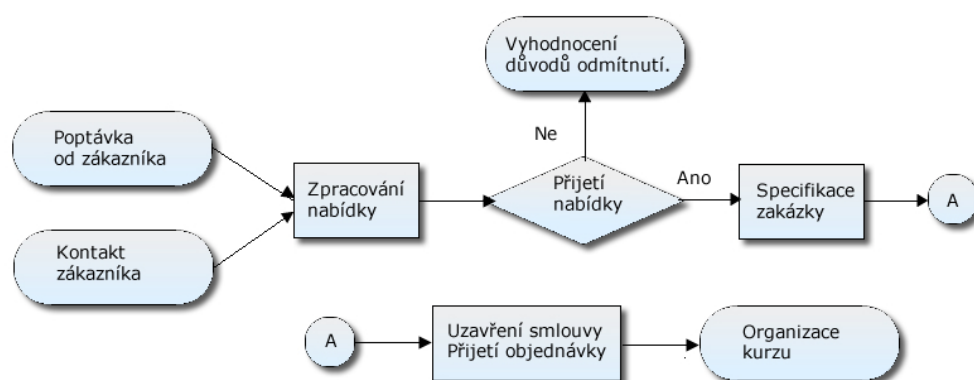
Hlavní činností je výuka v jazykových kurzech. S tím souvisí evidence odučených hodin na formulářích a v stávajícím informačním systému společnosti. Součástí výuky je i testování studentů z dosažených znalostí vyučovaného jazyka.

2.2 Procesy

Mezi hlavní procesy společnosti, které se týkají výuky, patří získávání zakázek, výuka, fakturace, platby.

2.2.1 Proces zakázek

Zakázky zajišťují vedoucí pracovníci společnosti ve spolupráci s pracovníky administrativy a lektory. Činnosti procesu zakázek jsou uvedeny v tabulce 1.



Obrázek 2: Proces zakázek

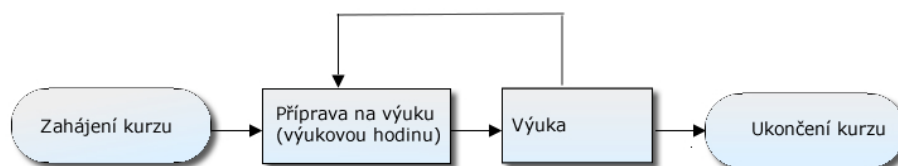
Tabulka 1: Proces zakázky – popis činností

Činnost	Pracovník	Vstup	Výstup	Poznámka*
Kontakt zákazníka	Vedoucí pracovník, lektor	Nabídka kurzů společnosti	Termín předložení nabídky	Propagační materiály
Zpracování nabídky	Vedoucí pracovník, sekretářka	Informace o požadavcích zákazníka	Nabídka	Počítač
Projednání nabídky	Vedoucí pracovník	Nabídka	Přijetí/odmítnutí nabídky	
Specifikace zakázky	Vedoucí pracovník, sekretářka	Nabídka, konkrétní požadavky zákazníka	Návrh smlouvy	Počítač
Uzavření smlouvy	Vedoucí pracovník	Návrh smlouvy	Uzavřená smlouva	Počítač
Organizace kurzu	Vedoucí pracovník, sekretářka	Smlouva se zákazníkem	Rozvrh kuru	Výstupem je kdy, kde a kdo bude kurz učit

* v sloupci poznámka jsou uvedeny použité nástroje, materiály nebo dodatečné informace o činnosti.

2.2.2 Proces výuky

Hlavní předmět činnosti společnosti – jazykovou výuku realizují lektori. Z povahy jazykové výuky vyplývá, že tuto činnost v konkrétním kurzu zajišťuje lektor sám v plném rozsahu a v současnosti s minimální spoluprací s ostatními pracovníky společnosti. Spolupráce se týká pouze zajištění výukových materiálů a evidence odučených hodin.



Obrázek 3: Proces výuky

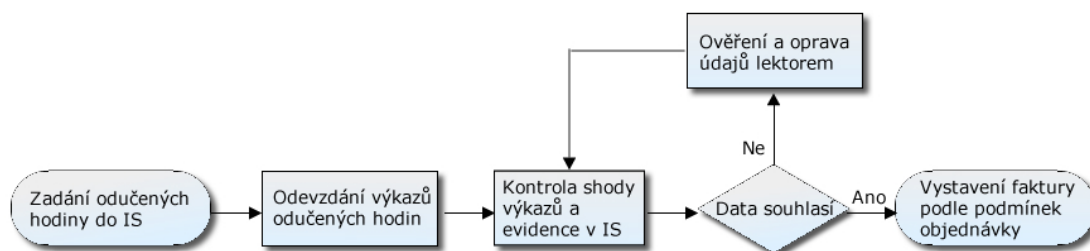
Aktivity v procesu výuky jsou v převážné míře aktivity, které se opakují, i když jejich obsah může být velice různorodý.

Tabulka 2: Proces výuky – popis činností

Činnost	Pracovník	Vstup	Výstup	Poznámka
Zahájení kurzu	Vedoucí pracovník, lektor	Organizace kurzu	Zahájení kurzu	
Příprava na výuku	Lektor	Plán výuky	Materiály na výuku v konkrétní hodině	Výukové materiály, počítač
Výuka	Lektor	Přípravy na hodinu	„Vzdělanější studenti“	Výukové materiály, počítač
Testování	Lektor	Obsah výuky	Vyhodnocení efektivity výuky	Počítač, testy
Ukončení kurzu	Lektor	Test	Zhodnocení kurzu	Počítač

2.2.3 Proces fakturace

Většina výuky se platí zpětně na základě evidence odučených hodin. Faktury jsou vystavovány studentům nebo firmám, které si objednaly jazykovou výuku. Činnost provádí oddělení účetnictví. Fakturace se provádí na začátku měsíce, který následuje po měsíci kdy proběhla výuka, podle smluvních podmínek o zajištění jazykové výuky.



Obrázek 4: Proces fakturace

Tabulka 3: Proces fakturace – popis činností

Činnost	Pracovník	Vstup	Výstup	Poznámka
Zadání odučených hodin do IS,	Lektor	Záznamy o výuce v jednotlivých hodinách	Přehled výuky za dané období	Kontrola 1x za měsíc
Kontrola údajů	Účetní	Přehled výuky	Zkontrolovaná evidence odučených hodin	Evidence v IS a na formulářích, provádí se manuálně
Vystavení a odeslání faktury	Účetní	Evidence odučených hodin	Vystavené a odeslané faktury	Faktury se vystavují v jiném IS než je evidence výuky
Kontrola zaplacení faktury	Účetní	Faktura – termín splatnosti	Likvidace faktury, žádost o provedení platby	Ve výjimečných případech odepsání faktury

2.3 Zaměstnanci

Činnost firmy zajišťují v podstatě tři skupiny pracovníků - management, administrativa a lektori. Znalosti, způsob práce a zkušenosti lektorů jsem zjišťoval pomocí dotazníku (viz. příloha A). Cílem dotazníkového šetření bylo zjistit informace o úrovni spolupráce mezi jazykovou školou a lektory, a informace o způsobu práce lektorů s informacemi a se stávajícím informačním systémem. Projektu in-

formačního systému JS se týkaly především otázky 2, 5, 11 a 12. Na dotazník odpovědělo celkem 15 lektorů.

Z dotazníkového šetření vyplynulo že:

Otázka č. 2. V jakém rozsahu využíváte internet, počítač ve výuce?
54 % lektorů využívá internet ve výuce pravidelně a 46 % lektorů občas.

Otázka č. 5 Dostáváte včasné a úplné informace potřebné pro vaši práci?

54 % lektorů si myslí, že má dostatečné informace, 13 % lektorů uvedlo, že spíše ne a 33 % lektorů uvedlo, že nemá dostatečné informace pro výuku. V poznámce na tuto otázku většina těch, kteří odpověděli, že mají nedostatečné informace, uvedli, že nedostatky se týkají především průběhu kurzu například nedostatečné informace o počtu odučených hodin.

Otázka č. 11 Jaké jsou, podle vás, hlavní nedostatky současného informačního systému jazykové školy?

Otázka č. 12. Jakým způsobem byste doplnili, změnili současný informační systém jazykové školy?

V otázce č. 11 a 12 téměř všichni lektori (90 %) uvedli, že se stávajícím informačním systémem jsou spokojeni, a že k němu nemají připomínky. Tento výsledek je částečně v rozporu s vyhodnocením otázky č. 5, kde jen 54 % lektorů uvedlo, že má dostačující informace.

Znalosti, způsob práce a zkušenosti pracovníků společnosti jsem zjišťoval prostřednictvím rozhovoru.

Rozhovory byly zaměřeny především na způsob práce s informacemi týkající se výuky. Z rozhovorů vyplynuly následující závěry o stávajícím IS.

1. Chybí elektronické vedení zakázek.
2. Neexistuje část informačního systému pro studenty, kde by studenti měli k dispozici informace o výuce.
3. Chybí možnost elektronického testování studentů.
4. Ve stávajícím IS není možnost evidovat plánovaný obsah výuky.
5. Komplikované vytváření přehledů o výuce.
6. Zcela chybí možnost evidence obsahu výuky.
7. Nemožnost zadávání změn výuky – suplované hodiny.

8. Chybí možnost generování rozvrhů.

Celkově lze říci, že jak lektoři, tak pracovníci jazykové školy jsou zvyklí pracovat s informačním systémem a mají k tomu i potřebné dovednosti. Z analýzy dále vyplynulo, že lektoři vidí roli IS pouze v zadávání odučených hodin a považují to za dostačující. Na druhé straně administrativní pracovníci vidí ve stávajícím systému řadu nedostatků, které brání dalšímu rozvoji.

2.4 Technologie - Informační systém

V současné době společnost využívá dva programy pro potřeby administrativy a řízení společnosti - Informační systém pro evidenci kurzů a výuky a systém pro vedení účetnictví. Informační systém pro evidenci kurzů obsahuje pouze funkce pro evidenci kontaktů, kurzů a výuky.

Dalším charakteristickým znakem současného způsobu práce ve společnosti je, že veškeré procesy jsou do značné míry založeny na papírových podkladech (evidence výuky, faktury, informace o zakázkách, lektorech, kontaktech) atd., přičemž řada těchto dat neexistuje v elektronické podobě. Tento stav velice snižuje efektivitu práce s daty a jejich sdílení.

Podobná je i situace v oblasti výuky. Společnost nemá systém pro sdílení výukových materiálů a pro testování studentů.

3 Určení funkcí nového informačního systému

Určení funkcí informačního systému je založeno na analýze současného stavu, především na analýze procesů a pracovníků společnosti.

Charakter organizační struktury určuje především rozdělení pravomocí, zodpovědnosti a činností v organizaci. Z tohoto rozdělení následně vyplývá, jaké jsou informační potřeby pracovníků vykonávající jednotlivé činnosti. Tyto informační potřeby potom určují požadavky na informační systém společnosti.

Rozdělení činností v organizaci vyplývá z popisu procesů v organizaci, které bylo uvedeno v předcházející části. Při analýze informačních potřeb jsem se zaměřil na čtyři základní procesy ve společnosti - proces zakázek, proces výuky, proces fakturace a proces plateb.

K analýze rozdělení pravomocí a zodpovědností jsem použil matici RACI, která umožňuje zobrazit odpovědnost a pravomoci jednotlivých pracovníků při realizaci jednotlivých činností procesu.

- R - pravomoc k provádění činnosti a zodpovědnost za její provedení
- A - kontrola

- C - konzultování před provedením případně v průběhu činnosti
- I - Informování

3.1 Funkce IS - zakázky

Z analýzy procesu zakázek vyplynulo, že realizace činností v procesu zakázek spočívá především na vedení společnosti s malým zapojením ostatních pracovníků společnosti. Jako hlavní důvod se jeví to, že pracovníci v oddělení administrativy nemají potřebné informace nutné k tomu, aby mohli být ve větším rozsahu zapojeni do realizace jednotlivých aktivit procesu zakázek.

Tabulka 4: RACI - proces zakázek

Činnost	Vedení	Lektoři	Administrativa
Zpracování návrhu zakázky	R/A		
Projednání návrhu zakázky	R/A		
Specifikace zakázky	R/A		C
Uzavření smlouvy	R/A		I
Organizace kurzu	R/A	C	I/R
Sledování plnění zakázky	R/A	C/I	R

Požadavek na informační systém 1. Z analýzy procesu vyplynulo, že realizace činností „Zpracování návrhu zakázky“ a „Organizace kurzů“ by mohla být delegována do oddělení administrativy. Aby bylo možné delegovat činnosti spojené s přípravou zakázky je nutné, aby informační systém obsahoval informace o připravovaných a uzavřených zakázkách, a bylo je tak možné sdílet mezi pracovníky společnosti.

Požadavek na informační systém 2. Vzhledem k tomu, že existuje vazba mezi procesem zakázek a procesem výuky, je také nutné podpořit tuto vazbu v IS. Aby byla zajištěna tato provázanost, je třeba umožnit v IS přiřadit zakázky jednotlivým kurzům, aby bylo možné zajistit sledování plnění zakázky. Současně je nutné informace potřebné pro lektory zapsat ke kurzům a ty zpřístupnit lektorům.

Požadavek na informační systém 3. Pro činnost organizace kurzů je potom nutné aby informační systém obsahoval informace o rozvrhu lektorů.

3.2 Funkce IS - výuka

Z analýzy procesu výuky vyplývalo, že naprostá většina pravomocí a odpovědností za jednotlivé činnosti se týká lektorů. To je fakt, který odpovídá jednotlivým

činnostem. Během analýzy se ale ukázalo, že úroveň konzultací a sdílení informací o průběhu výuky v jednotlivých kurzech je na velice nízké úrovni. Příčinou je neexistující organizační struktura pro sdílení informací a neexistující podpora ve stávajícím informačním systému společnosti.

Tabulka 5: RACI - proces výuka

Činnost	Vedení	Lektoři	Administrativa
Zahájení kurzu	A	R	I
Příprava na výuku		R/A	
Výuka	I	R/A	
Ukončení kurzu	R/A	I	I

Požadavek na informační systém 1. V informačním systému realizovat funkce, které umožní sledovat a analyzovat aktivity v kurzech a vyhodnocovat efektivitu výuky v kurzech. Což Konkrétně to znamená evidovat podrobné informace o aktivitách během výuky a zavést online testování studentů.

Požadavek na informační systém 2. Za většinu aktivit procesu zodpovídají lektoři. Vzhledem k tomu, že ale lektoři v kurzu realizují pouze výuky je vhodné, aby v informačním systému existovala možnost zobrazit dodatečné informace o kurzu. Jedná se o informace týkající se zakázky při zahájení kurzu nebo v průběhu realizace zakázky.

3.3 Funkce IS - fakturace

Z analýzy obou procesu fakturace vyplynulo, že za realizaci většiny činností odpovídá oddělení administrativy/účetnictví. Lektoři odpovídají jenom za dodání podkladů pro fakturaci. Hlavním problémem celého procesu je, že je z velké části založen na papírové evidenci informací vztahující se k určení částky faktury. To se týká především informací o zakázkách a evidence změn ve výuce hodin (suplované hodiny).

Požadavek na informační systém 1. V informačním systému realizovat funkce, které umožní vytvářet přehledy odučených hodin včetně změn ve výuce.

Požadavek na informační systém 2. V informačním systému realizovat evidenci zakázek.

4 Analýza případů užití IS

Podle způsobu práce s informačním systémem se uživatelé systému jazykové školy dělí do tří skupin:

Tabulka 6: RACI - fakturace

Činnost	Vedení	Lektoři	Administrativa
Zadání odučených hodin do IS	I	R	A
Odevzdání výkazů odučených hodin	I	R	A
Ověření správnosti údajů	A		R
Vystavení faktury	C/I		R/A
Kontrola zaplacení faktury	I		R/A
Žádost o provedení platby v případě nezaplacení	I/C		R/A
Likvidace faktury	A		R

1. administrativa (vedení společnosti, pracovníci administrativy, účetní,
2. lektoři,
3. studenti.

4.1 Administrativa

Případy užití vedení společnosti a pracovníků administrativy je možné rozdělit do těchto skupin:

- evidence - evidence kontaktů, kurzů a zakázek,
- přehledy - přehledy odučených hodin podle kurzů, lektorů, organizací,
- vytváření testů.

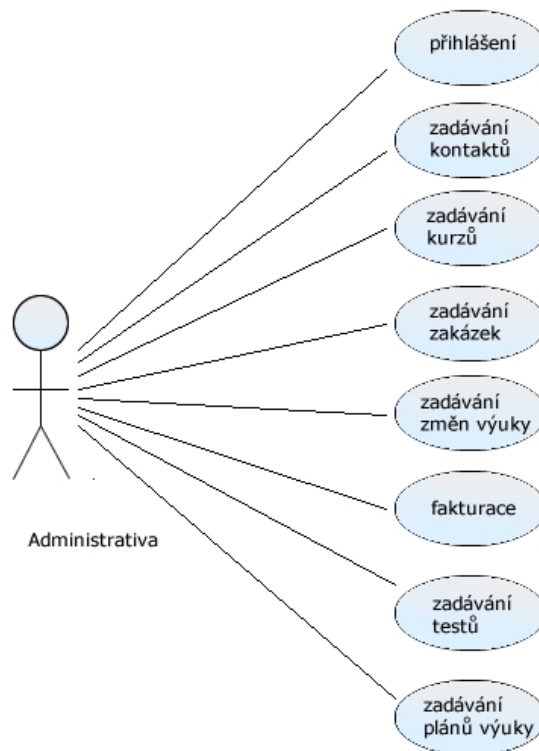
Jednotlivé funkce jsou zobrazeny na obrázku 5. (Případy užití Administrativa)

4.2 Lektoři

Případy užití lektorů je možné rozdělit do těchto skupin:

- evidence výuky – zadávání odučených hodin, obsahu výuky v jednotlivých hodinách, zadávání absence studentů,
- testování – prohlížení testů, přidělování testů studentům, zobrazení výsledků,
- výkazy – vytváření přehledu odučených hodin.

Jednotlivé funkce jsou zobrazeny na obrázku 6. (Případy užití Lektoři)



Obrázek 5: Případy užití Administrativa

4.3 Studenti

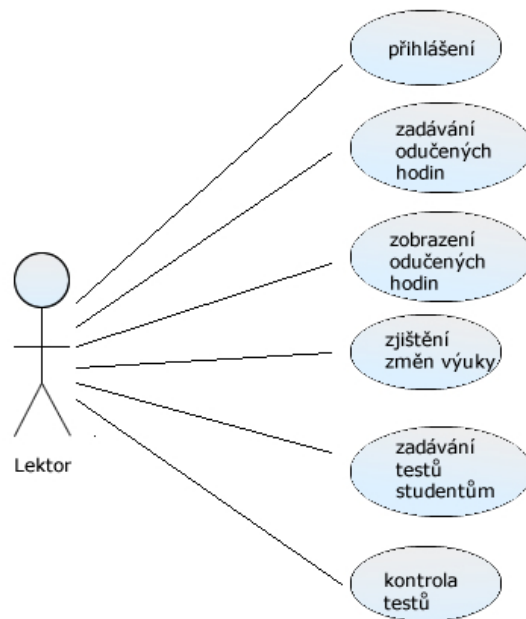
Případy užití studentů je možné rozdělit do těchto skupin:

- Přehled výuky – zobrazení informací o obsahu odučených hodin, omlouvání absence v případě nepřítomnosti
- Testování – on line testování studentů

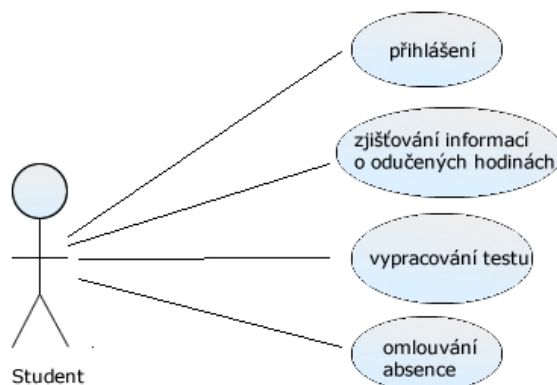
Jednotlivé funkce jsou zobrazeny na obrázku 7. (Případy užití Studenti)

5 Architektura informačního systému

Východiskem pro stanovení struktury informačního systému je analýza informačních potřeb a případů užití informačního systému. Z této analýzy vyplývá, že k informačnímu systému by mělo mít přístup velké množství uživatelů ze strany studentů a lektorů, a pouze omezené množství pracovníků jazykové školy. Z analýzy dále vyplývá, že společnou vlastností všech funkcí informačního systému je manipulace s daty. Protože se jedná o data sdílená všemi uživateli, tak jediné



Obrázek 6: Případy užití Lektori



Obrázek 7: Případy užití Studenti

vhodné řešení je umístění dat ve společné databázi, ke které budou mít uživatelé přístup prostřednictvím příslušné aplikace. Vhodným řešením je tedy struktura informačního systému založená na třívrstvé architektuře.

Řešení prezenční a aplikační vrstvy

Vzhledem k tomu, že se funkce informačního systému pro jednotlivé uživatele nepřekrývají, navrhl jsem rozdělit informační systém na aplikace, které budou zajišťovat požadovanou funkcionalitu pro danou skupinu uživatelů. Systém se tedy skládá z aplikace Student, aplikace Lektor a aplikace Administrativa. Aplikace Student a Lektor jsou řešeny jako webové aplikace a aplikace Administrativa jako desktopová aplikace. Toto řešení odpovídá požadavkům jednotlivých skupin uživatelů na přístup k informačnímu systému. Aplikacím prezenční vrstvy odpovídají na straně severu aplikace, které provádějí operace manipulující s daty podle požadavků uživatelů.

Databáze

Základem informačního systému je společná databáze zajišťující práci s daty, především ukládání, výběr a změnu dat.

6 Technologie

Pro tvorbu program jsem použil následující technologie:

Technologie prezenční vrstvy

Webové aplikace pro studenty a lektory jsou vytvořeny s využitím frameworku Flex 4.6 a programovacího jazyka AS3. Pro framework Flex jsem se rozhodl z důvodu kvalitních knihovny komponent pro vytváření webových aplikací. Další výhodou je že aplikace na vytvořené v jazyce AS3 běží v prostředí Flash Player a z programátorského hlediska není nutné řešit rozdíly mezi webovými prohlížeči. Desktopová aplikace pro administrativu je vytvořena s využitím frameworku JavaFX a programovacího jazyka Java. Toto řešení jsem zvolil vzhledem k tomu, že serverová část jsem naprogramoval v jazyce Java a z hlediska integrace je tedy výhodné využít stejný jazyk v prezenční i aplikační vrstvě. Bylo by samozřejmě výhodné vytvořit i webové aplikace v Javě, ale Java nemá ve webových prohlížečích takovou podporu a není tak rozšířena jako aplikace vytvořené pro Flash Player.

Technologie aplikační vrstvy

Aplikace na serveru zajišťující výměnu dat mezi databázemi a aplikacemi prezenční vrstvy jsou vytvořeny na platformě JavaEE technologie EJB3. Pro výměnu dat mezi klientem a serverem je použit framework GraniteDS. Pro přenos dat je použit formát AMF3. AMF3 je binární formát používaný pro komunikaci mezi klientem a serverem.

Technologie datové vrstvy

Pro uložení a manipulaci s daty jsem použil databázi MySQL ve spojení s frameworkem JPA (Java).

7 Technické řešení

Technické řešení informačního systému vychází ze zvolené architektury a technologie.

7.1 Aplikace prezentační vrstvy

7.1.1 Webová aplikace Student

Webová aplikace Student realizuje uživatelské rozhraní informačního systému pro studenty jazykové školy. Aplikace se skládá ze dvou modulů Account a Tests, které obsahují funkce pro práci s informačním systémem. Po načtení aplikace se nastaví parametry spojení se serverem a vytvoří se objekt ro, což je objekt třídy SecureRemoteObject. Objekt ro následně používají funkce modulů pro komunikaci se serverem. Po úspěšném přihlášení studenta se nastaví spojení se serverem a student může využívat funkcionalitu modulů. Moduly implementují interface ModulInterface. Implementace ModuluInterface je nutná pro komunikaci mezi aplikací a jednotlivými moduly. Jednotlivé moduly obsahují funkci init(), která se spustí při načtení modulu. Funkce init() zajistí inicializaci modulu. Ve funkci init() především dojde k vytvoření odkazu na objekt SecureRemoteObject, viz objekt ro. Jednotlivé funkce využívají frameworku GraniteDS., který zajišťuje asynchronní komunikaci mezi klientem a serverem. Při volání serveru je určena i obsluha události přijetí výsledku viz. zdrojový kód 1. funkce onGetData(ResultEvent) .

```
1 public function getData():void{
2     this.ro.destination="DestinationService"
3     this.ro.funkceDS.addEventListener(ResultEvent.RESULT,onGetData)
4     this.ro.funkceDS (parametry);
5 }
```

Zdrojový kód 1: AS3 - volání funkce na serveru.

Po vrácení výsledků volání následně funkce onGetData() zpracuje přijatá data a zajistí jejich prezentaci uživateli zdrojový kód 2. Výsledky volání jsou pře-

```
1 private function onGetData(event:ResultEvent):void{
2     var result:Object;
3     result=event.result;
4     .
5     .
6 }
```

Zdrojový kód 2: AS3 - zpracování přijatých dat.

dávány funkcím aplikace Student jako objekty, kde předávaná data jsou prezentována jako hodnoty vlastností objektu. Toto mapování dat zajišťuje framework GraniteDS s využitím formátu AMF3.

Funkce EJB komponent na serveru výsledky volání vloží do kolekce HashMap, která je webovou aplikací přijata jako objekt. Klíče objektu HashMap javy jsou mapovány na vlastnosti objektu typu Object v AS3, přičemž hodnoty klíčů kolekce HashMap se navážou na hodnoty vlastností objektu typu Object. Tento objekt je přístupný jako hodnota vlastnosti result objektu ResultEvent. Objekt ResultEvent je parametrem funkce, která byla definována jako obsluha události při volání funkce EJB komponenty na serveru.

Modul Account obsahuje funkce pro zobrazení výukových hodin a funkce pro omlouvání absence studentů ve výuce.

Modul Test obsahuje funkce pro načtení testu a vypracování testu, který byl studentovi přiřazen.

7.1.2 Webová aplikace Lektor

Webová aplikace Lektor poskytuje uživatelské rozhraní informačního systému pro lektory jazykové školy. Koncepte aplikace je stejná jako v případě aplikace Student. Na úrovni aplikace je opět vytvořen objekt ro SecureRemoteObject, který zajišťuje komunikaci mezi klientem a serverem. Způsob komunikace je stejný, jak bylo popsáno v popisu aplikace Student. Aplikace Lektor se skládá ze čtyř modulů Lessons, Test, TestResult, Report. Jednotlivé moduly obsahují funkce pro práci s webovou aplikací a pro komunikaci se serverem.

Funkce Modulů

Modul Lessons obsahuje funkce pro zápis odučených hodin a absence studentů. Modul Test obsahuje funkce pro zobrazení testů a jejich přidělení studentům. Modul TestResult obsahuje funkce pro zobrazení výsledků testů, které studenti vypracovali.

Modul Report obsahuje funkce pro vytvoření přehledu odučených hodin lektora za určité období.

7.1.3 Aplikace Administrativa

Aplikace Administrativa je řešena jako desktopová aplikace vytvořená ve frameworku JavaFx. Stejně jako v případě webových aplikací informačního systému se při přihlášení uživatele nastaví parametry spojení se serverem.

Aplikace se skládá z modulů, které zajišťují vlastní funkcionalitu aplikace. Jednotlivé moduly jsou tvořeny dvojicí FXML souboru a kontroloru. FXML soubory definují vizuální vzhled modulu. V podstatě jsou zde definovány prvky

formuláře, které jsou viditelné uživateli, a s kterými uživatel pracuje. Kontrolor je třída napsaná v jazyce Java, která obsahuje metody obsluhující formulář.

Metody, které komunikují se serverem realizují tuto komunikaci prostřednictvím objektu třídy DbService. Třída DbService je potomkem třídy Task (Task je třída Javy, která obsahuje funkce, které mohou být vykonávány ve vláknech) a je použita jako parametr při vytvoření nového vlákna. Třída DbService využívá pro komunikaci se serverem objekty třídy RemoteService frameworku GraniteDS. Objekt třídy RemoteService se inicializuje s použitím parametrů připojení k serveru, které byly nastaveny při přihlášení uživatele. Objekt RemoteService je vytvořen v konstruktoru třídy DbService, zdrojový kód 3.

```
1
2 public DbService(String service, String des, HashMap hm) {
3
4     // roService je objekty typu RemoteService;
5     this.roService = new RemoteService(MainController.getChannel()
6         , service);
7     this.map = hm;
8     this.dest = des;
9 }
10 }
```

Zdrojový kód 3: Konstruktor třídy DbService

V proměnné service je název EJB3 komponenty na serveru, jejíž metoda má být volána viz metoda call() zdrojový kód 4. Proměnná dest obsahuje název metody a v proměnné hm jsou hodnoty předávaných dat.

Vlastní komunikace se serverem je realizována prostřednictvím spuštění nového vlákna, což umožňuje asynchronní volání serveru. Při volání serveru je vytvořeno nové vlákno, kterému je jako parametr předán objekt třídy DbService. Po vytvoření nového vlákna, je vlákno spuštěno, čímž dojde k zavolání metody call objektu DbService zdrojový kód 4. V metodě call je následně volána metoda newInvocation objektu roService (roService je objekt typu RemoteService)

Výsledky volání serveru jsou potom zpracovány metodou třídy, která vytvořila objekt třídy DbService a spustila vlákno s tímto objektem. Základní součástí zpracování je zobrazení výsledků volání serveru uživateli. Aplikace Administrativa se skládá z těchto modulů main, abContacts, abGroups, abOrders, abTimeTible, absence, groups, contacts, lessons, order, planEditor, subLessons, tests.

Funkce modulů

Modul main (MainController.java, main.fxml) Hlavní okno aplikace, obsahuje metody pro přihlášení a zobrazování ostatních modulů.

```

1
2 public HashMap call() throws Exception {
3     ResponseMessageFuture invoke;
4     invoke = roService.newInvocation(dest, map).addListener(new
5         ResultFaultIssuesResponseListener() {
6         @Override
7         public void onResult(ResultEvent event) {
8             HashMap data = (HashMap) event.getResult();
9             .
10            .
11            .
12        }
13    }
14 }

```

Zdrojový kód 4: volání funkce na serveru

Modul abContacts (AbContactsController.java, abContacts.fxml) obsahuje metody pro zobrazení seznamu kontaktů.

Modul abGroups (AbGroupsController.java, abGroups.fxml) obsahuje metody pro zobrazení seznamu skupin (kurzů).

Modul abOrders (AbOrdersController.java, abOrders.fxml) obsahuje metody pro zobrazení seznamu objednávek.

Modul abTimeTable (AbTimeTableController.java, abTimeTable.fxml) obsahuje metody pro generování rozvrhů učeben a lektorů. Součástí Modul jsou i metody pro zadávání a editaci učeben.

Modul absence (AbsenceController.java, absence.fxml) obsahuje metody pro zobrazení přehledu absence za určité období.

Modul contacts (ContactsController.java, contacts.fxml) obsahuje metody pro zadávání a editaci kontaktů.

Modul groups (GroupsController.java, groups.fxml) obsahuje metody pro zadávání a editaci skupin (kurzů).

Modul order (OrderController.java, order.fxml) obsahuje metody pro zadávání a editaci objednávek.

Modul lessons (LessonsController.java, lessons.fxml) obsahuje metody pro zobrazení přehledu odučených hodin.

Modul planEditor (PlanEditorsController.java, planEditor.fxml) obsahuje zadávání a editaci plánů výuky.

Modul subLessons (SubLessonsController.java, subLessons.fxml) obsahuje metody pro zadávání a editaci změn ve výuce.

Modul tests (TestsController.java, tests.fxml) obsahuje metody pro zobrazení zadávání a editaci testů.

7.2 Aplikace serverová část

Jednotlivým aplikacím realizující uživatelské rozhraní informačního systému odpovídají aplikace na serveru, které manipulují s daty v databázi. Jedná se tedy o tři aplikace Student, Lektor a Administrativa. Jednotlivé aplikace se skládají z EJB komponent. EJB komponenty obsahují metody, které jsou volány z klientské části informačního systému. Úlohou metod je manipulace s daty podle požadavků klientské části. Každá komponenta obsahuje objekt typu EntityManager. EntityManager je třída, která je součástí frameworku JPA. Objekt typu EntityManager spravuje entity, které reprezentují tabulky v relační databázi. Metody komponent tedy manipulují s daty prostřednictvím objektu třídy EntityManager. Získávání dat z databáze je většinou realizováno prostřednictvím dotazů napsaných v jazyce JPQL. Dotazy JPQL jsou definovány pomocí anotací @NamedQuery v jednotlivých entitách. Tyto dotazy jsou potom následně využívány v metodách komponent .

Vlastní komunikaci mezi aplikacemi v klientské části a komponentami na serveru zajišťuje knihovna frameworku GraniteDS.

Aplikace Student obsahuje tyto komponenty UserB a LessonsB. Komponenta UserB je definována jako stavová a obsahuje metody pro přihlášení studenta a testování studenta. Komponenta LessonsB je definována jako bez stavová a obsahuje metody zobrazující data o výuce.

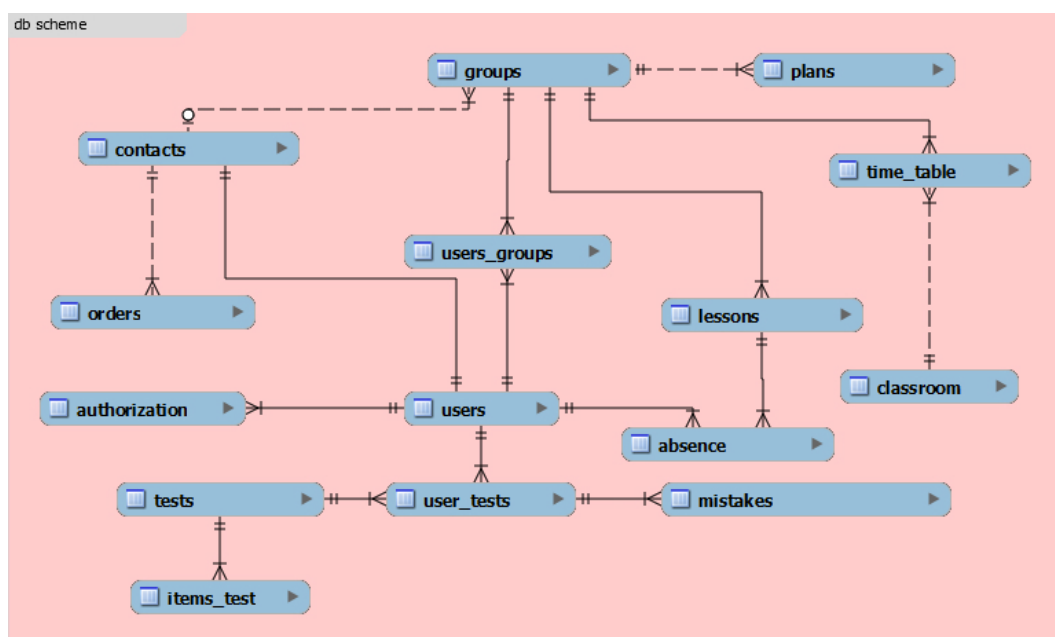
Aplikace Lektor obsahuje tyto komponenty UserB, LessonsB a TestB. Komponenta UserB je definována jako stavová a obsahuje metody pro přihlášení lektora. Komponenta LessonsB je definována jako bez stavová a obsahuje metody realizující zadávání výuky. Komponenta TestB je definována jako bez stavová a obsahuje metody pro testování studentů.

Aplikace Administrativa obsahuje tyto komponenty UserB, ContactsB, OrdersB, GroupsB a TestsB. Komponenta UserB je definována jako stavová a obsahuje metody pro přihlášení pracovníka. Komponenta ContactsB je definována jako bez stavová a obsahuje metody pro editaci a zobrazování kontaktů. Komponenta OrdersB je definována jako bez stavová a obsahuje metody pro editaci a zobrazování objednávek. Komponenta GroupsB je definována jako bez stavová a obsahuje metody pro editaci, zobrazování kurzů, editaci a zobrazování učeben a plánů a metody pro vytváření přehledů odučených hodin a absence. Komponenta TestsB je definována jako bez stavová a obsahuje metody pro edi-

taci a zobrazování testů.

7.3 Databáze

Databáze je vytvořena MySQL. Jednotlivé tabulky jsou mapovány na entity a manipulace s daty je realizována prostřednictvím těchto entit. Knihovna entit je následně sdílena jednotlivými aplikacemi na serveru. Schéma databáze je zobrazeno na obrázku 8.



Obrázek 8: Schéma databáze

8 IS- uživatelská část

8.1 Webová aplikace Student

Prostřednictvím webové aplikace Student mají studenti přístup k informacím o výuce a k online testům, které lektor přidělí studentů ve skupinách, ve kterých vyučuje. Součástí aplikace jsou dva moduly: modul Kurzy a modul Test. Po přihlášení do aplikace se zobrazí tlačítka odkazující na tyto dva moduly a tlačítko Odhlásit pro odhlášení z aplikace.

8.1.1 Modul Kurzy

Modul Kurzy obsahuje funkce zpřístupňující informace o výuce v kurzech, ve kterých je zařazen mezi studenty.

Zobrazení informace o obsahu hodiny

Po přihlášení se studentovi zobrazí seznam kurzů, v kterých je zařazen. Kliknutím na název kurzu se načtou data výuky ve zvoleném kurzu a kliknutím na datum se zobrazí informace o výuce v označeném dni.

Informace o absenci

Před zadáním požadavku na zobrazení dnů absence studenta je nutné označit kurz, pro který se má absence zobrazit. Kliknutím na tlačítko absence se zobrazí data absence studenta v označeném kurzu. Pro jednotlivé dny absence může student zadat informace o typu a důvodu absence.

8.1.2 Modul Testy

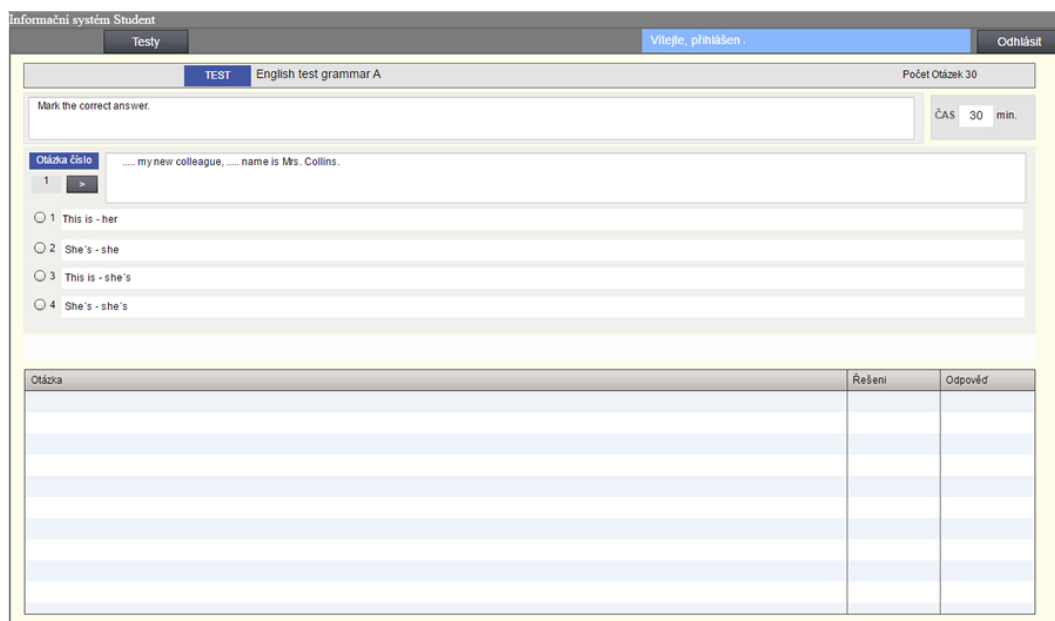
Modul testy umožňuje online testování studentů. Online test, který je studentovi přidělen lektorem, si student zobrazí kliknutím na tlačítko Zobrazit testy. Po kliknutí na tlačítko aplikace načte přidělené testy a zobrazí je v tabulce testů. Přiřazený test může student dělat pouze jednou. To znamená, že po spuštění testu se tento test již nezobrazí. Kliknutím na test v tabulce testů se test načte a zobrazí se okno, kde může student vypracovat označený test (viz. obrázek 8). Test má přiřazen časový limit během, kterého může student odpovídat na testové otázky. Vlastní test se spustí kliknutím na tlačítko Začít test. Po kliknutí na tlačítko se zobrazí první otázka testu. Na jednotlivé otázky student odpovídá volbou jedné ze čtyř možností. Po označení odpovědi na otázku testu se kliknutím na tlačítko další zobrazí další otázka. Správné řešení se zobrazí po zodpovězení všech otázek. Červeně jsou označeny chybně zodpovězené otázky, zeleně správně zodpovězené otázky. Kliknutím na otázku se zobrazí otázka a její řešení.

Vyhodnocení správnosti testu

Po zodpovězení poslední otázky se zobrazí seznam otázek testu s označením chybně zodpovězených otázek. Chybně zodpovězené otázky jsou označeny červeně. Kliknutím na otázku je možné zobrazit otázku a možné odpovědi s označením správné odpovědi. Odpověď studenta je zobrazena v tabulce s otázkami v testu. Číslo v posledním sloupci tabulky označuje číslo odpovědi studenta.

8.2 Webová aplikace Lektor

Webová aplikace Lektor slouží k zadávání odučených hodin a k testování studentů. Po přihlášení do systému se lektorovi zobrazí úvodní obrazovka. Úvodní obrazovka obsahuje v horní části seznam modulů a tlačítko k odhlášení ze systému. Aplikace se skládá z těchto modulů: Třídnice, Testy, Testy kontrola a Přehled výuky.



Obrázek 9: Modul Testy

8.2.1 Modul Třídnice

Po kliknutí na odkaz třídnice se zobrazí úvodní okno modulu, které je rozděleno na dvě části. Horní část obsahuje tabulku se seznamem kurzů, ve kterých lektor aktuálně vyučuje. V dolní části okna je tabulka pro zobrazení suplovaných hodin.

Zápis odučených hodin

Okno pro zápis odučené hodiny v daném kurzu se zobrazí po kliknutí na tlačítko zápis v tabulce obsahující seznam kurzů lektora. Po kliknutí na tlačítko se načte seznam již odučených hodin a tento seznam jako datum výuky se zobrazí v tabulce v levé části okna pro zápis odučené hodiny (viz. obrázek 9). Současně s načtením již odučených hodin se načte i seznam studentů v daném kurzu. Formulář pro zápis odučené hodiny je rozdělen do třech částí. Horní část obsahuje pole pro vlastní zápis hodiny v daném dni, prostřední část obsahuje pole pro zápis aktivit hodiny a spodní část obsahuje tabulku se seznamem studentů pro zápis absence. Při vlastním zápisu hodiny lektor zapíše cíl hodiny, označí datum výuky, označí délku hodiny a hodinu uloží kliknutím na tlačítko uložit hodinu. Po kliknutí na tlačítko se provede zápis hodiny a zpřístupní se tlačítko pro zápis aktivit hodiny.

Aktivity v hodině

Aktivity hodiny jsou rozděleny do několika typů gramatika, slovíčka, konverzace, poslech, čtení, psaní, odborné téma, úkol a poznámka k hodině. Lektor zapíše do textového pole Popis – aktivit a označí její typ. Vlastní zápis se provede kliknutím na tlačítko Zapsat aktivitu. Zapsaná aktivita se zobrazí v tabulce aktivit.

Lektor | Třídnice | Testy | Testy Kontrola | Přehled výuky

Kurzy | Nová hodina | Uložít hodinu | Vymazat hodinu | Plán | Kurz info

Kurz beta kurz aj1 Číslo kurzu: 18 Číslo hodiny

Výuka

- 09/10/14
- 27/10/14
- 02/11/14
- 01/04/15

Celkem hodin: 4

Cíl Hodiny

Cíl Hodiny:

Datum výuky:

Čas výuky: 60 90

Aktivty hodiny

Typ Aktivty: Gramatika Slovička Konverzace Poslech Čtení Psaní Odborné téma Úkol Poznámka

Popis:

Aktivita	Typ	Vymazat

Studenti	Email	Mobil	Absence	X	Info
Karel Černý	Cerny@seznam.cz	789797979			
Bohuslava Havlíčková	slavka.havlickova@wikia7.cz				
Alena Bela	Bela@seznam.cz	77441122			
Pavla Slavíková	slavikova@bosch.cz				
Jana Nováková	jnovakova@bosch.cz				
Důvod Absence					

Obrázek 10: Aplikace Lektor - zápis výuky.

Informaci o uložené aktivitě je možné změnit zobrazením aktivity, změnou popisu aktivity a jejím uložení. Aktivita se zobrazí kliknutím na aktivitu. Uloženou aktivitu je možné vymazat kliknutím na tlačítko X v tabulce aktivit.

Absence studentů

Absenci studenta v dané hodině je možné zapsat kliknutím na jméno studenta v tabulce se jmény studentů. Zapsanou absenci je možné zrušit kliknutím na tlačítko X v tabulce se seznamem chybějících studentů. Informaci o absenci konkrétního studenta je možné zobrazit kliknutím na tlačítko I. Modul nápověda obsahuje informace o funkcích aplikace. Po kliknutí na tlačítko Nápověda se zobrazí okno s nápovědou.

Suplované hodiny

Zápis hodin v kurzech, ve kterých lektor normálně nevyučuje, a kde měl výuku za jiného lektora se provádí v dolní části úvodního okna modulu třídnice. Prvním krokem zápisu suplované hodiny je zobrazení seznamu suplovaných hodiny od určitého data. Seznam suplovaných hodin se zobrazí označením data, od kterého se má seznam vytvořit a kliknutím na tlačítko Suplované hodiny. Seznam suplovaných hodin se zobrazí v tabulce suplovaných hodin. V této tabulce je uveden čas výuky suplované hodiny a u každé hodiny jsou v tabulce dvě tlačítka. Tlačítko I pro zobrazení informací o suplované hodině a tlačítko Z pro zápis suplované hodiny. Po kliknutí na zápis suplované hodiny se zobrazí okno pro zápis suplované hodiny. Vlastní zápis suplované hodiny se provede stejným způsobem jako zápis hodiny v kurzech lektora.

8.2.2 Modul testy

V modulu testy je možné si prohlédnout vytvořené testy, které je možné zadat studentům ve zvolené skupině. Po zobrazení modulu se načte seznam kurzů, ve kterých lektor vyučuje. Tento seznam se zobrazí v tabulce v levé části okna modulu testy. Kliknutím na název kurzu se zobrazí seznam studentů v daném kurzu a současně se označí skupina, které je možné test přiřadit.

Zadání testu

Zadání testu se skládá ze dvou kroků: první krok je výběr testu, druhý krok je přiřazení testu. V prvním kroku lektor určí jazyk a úroveň testů a kliknutím na tlačítko Testy se zobrazí seznam testů, které odpovídají zadaným parametrům, v tabulce testů. Kliknutím na test lektor test vybere. Označený test je možné si prohlédnout a zadat vybrané skupině. Vlastní zadání testu se provede kliknutím na tlačítko Zadat test skupině. Před zadáním je nutné označit skupinu, které má být test zadán. Čas, který mají studenti na vypracování testu, je možné ponechat nebo změnit v poli Čas na vypracování testu.

8.2.3 Modul Testy Kontrola

V modulu Testy Kontrola si lektor může zobrazit testy, které studenti vypracovali. Po načtení modulu se načte seznam skupiny. Kliknutím na název skupiny v tabulce se zobrazí seznam studentů ve skupině. Kliknutím na jméno studenta se testy, které student vypracoval zobrazí v tabulce Testy. V případě, že student test zpracoval, se zobrazí čas, kdy test student vypracoval. Přiřazený test je možné studentům individuálně odebrat kliknutím na tlačítko X.

Zobrazení výsledků testu.

Informace o vypracování testu se zobrazí kliknutím na tlačítko zobrazit. Po kliknutí na tlačítko se zobrazí souhrnné informace o testu a seznam chybně zpracovaných odpovědí. Chybně zodpovězené otázky je možné prohlížet postupně, nebo kliknutím na otázku v tabulce chybně zodpovězených otázek. U každé chybně zodpovězené otázky se zobrazí zadání otázky a možné odpovědi. Správná odpověď je označena zeleně, chybná odpověď studenta je označena červeně.

8.2.4 Modul Přehled výuky

Přehled výuky slouží jako informace pro lektora o počtu odučených hodiny v jednotlivých kurzech za zvolené období.

Vytvoření přehledu.

V okně přehledu označených hodin lektor označí období, za které chce zobrazit přehled odučených hodiny a kliknutím na tlačítko Přehled výuky se vytvoří seznam kurzů, ve kterých v označeném období proběhla výuka. U každého kurzu

je uveden počet hodin v daném období.

8.3 Aplikace Administrativa

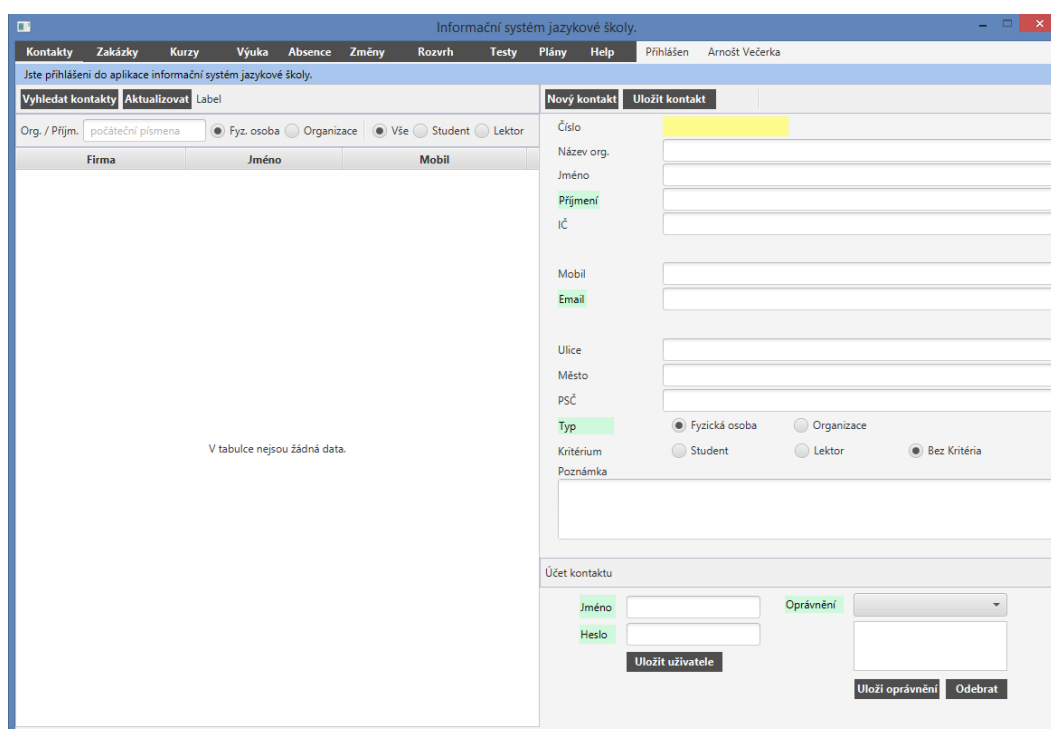
Administrativní část informačního systému slouží k evidenci dat souvisejících s výukou v jazykových kurzech. Vlastní systém se skládá z modulů (adresář, kurzy, zakázky, výuka, testy atd.), které obsahují funkce informačního systému. Zobrazení jednotlivých modulů se provádí přes hlavní menu, které je umístěno v horní části aplikace. Po spuštění programu se zobrazí přihlašovací okno, do kterého uživatel zadá své přihlašovací údaje (jméno a heslo). Po kladném ověření přístupových údajů se zobrazí úvodní panel hlavní menu. Hlavní menu obsahuje tlačítka s názvy jednotlivých modulů. Po stisknutí příslušného tlačítka se načte příslušný modul a uživatel může pracovat s funkcemi příslušného modulu. V jednotlivých modulech zadává uživatel hodnoty, na základě kterých se potom vykonávají jednotlivé funkce. Hodnoty ve formulářích se zadávají do polí, které jsou podle barvy rozděleny na bílé a žluté pole. Do bílých textových polí lze hodnoty zapisovat přímo, do žlutě označených polí hodnoty přímo zapisovat nelze. Hodnoty se do těchto polí vkládají výběrem a následným načtením hodnot ze seznamů. Názvy polí, které jsou označeny zeleně je třeba povinně vyplnit před zadáním požadavku na zpracování hodnot formuláře.

8.3.1 Modul Adresář

Formulář adresáře je rozdělen na dvě části (viz. obrázek 11). Levá část obsahuje tabulku se seznamem kontaktů a pravá část obsahuje formulář pro zobrazení a zadávání údajů kontaktů.

Seznam kontaktů

Panel se seznamem kontaktů se skládá z nástrojové lišty s tlačítky, z filtru pro zadávání zadání kritérií, podle kterých se vytvoří seznam kontaktů, a z tabulky zobrazující seznam kontaktů odpovídající zadaným kritériím. Kliknutím na položku v tabulce kontaktů se data kontaktů zobrazí ve formuláři v pravé části Adresáře. Nástrojová lišta seznamu obsahuje tlačítka pro zobrazení seznamu kontaktů a tlačítko pro aktualizaci seznamu kontaktů. Aktualizaci seznamu je třeba provést manuálně v případě, že jsou do databáze přidány nové kontakty a uživatel chce tyto kontakty zobrazit v tabulce obsahující seznam kontaktů. V této tabulce se nové kontakty zobrazí po stisku tlačítka aktualizace v případě, že vlastnosti kontaktu odpovídají nastavení filtru pro vyhledávání kontaktů. Filtr pro zadání kritérií pro vyhledání kontaktů obsahuje možnosti pro vyhledání podle příjmení nebo názvu organizace, typu kontaktu – fyzická osoba nebo organizace. V případě, že se jedná o fyzické osoby, tak je možnost zadat kritérium Lektor pro vyhledání lektorů nebo kritérium Student pro vyhledání studentů. Výchozí nastavení je Vše. Seznam kontaktů se zobrazí v tabulce, ve které je uveden název organizace, jméno kontaktů a číslo mobilního telefonu.



Obrázek 11: Aplikace Administrativa - Adresář.

Zadávání kontaktů

Panel pro zadávání nových kontaktů je v pravé části modulu Adresář. Pravá část obsahuje nástrojovou lištu, formulář pro zadávání kontaktu a část pro vytváření uživatelských účtů kontaktů. Nástrojová lišta obsahuje dvě tlačítka: Nový kontakt a Uložit. Tlačítko Nový kontakt vymaže data v polích formuláře a tlačítko Uložit uloží nový kontakt do databáze, případně uloží změny dat existujícího kontaktu. Formulář obsahuje pole pro zadávání dat kontaktu. Povinné položky pro zadání kontaktu jsou označeny zeleně. U každého kontaktu je třeba uvést typ kontaktu, zda se jedná o fyzickou osobu nebo organizaci, email kontaktu a příjmení v případě fyzické osoby, případně název organizace. Nelze zadat dva kontakty se stejným emailem.

Uživatelské účty

Pro kontakty typu Lektor a Student, případně pro další uživatele administrativního systému je třeba vytvořit uživatelský účet. Uživatelský účet se vytvoří zadáním jména a hesla uživatele. Do systému nelze vložit uživatele s existujícím jménem. Uživatelský účet lze vytvořit pouze pro kontakt typu fyzická osoba, ne pro kontakt typu organizace. V případě, že má uživatel vytvořen uživatelský účet, je mu možné přiřadit oprávnění pro práci s jednotlivými částmi informačního systému, což jsou oprávnění pro práci s webovými aplikacemi Lektor a Student, případně oprávnění pro práci s programem pro administraci.

8.3.2 Modul Zakázky

Formulář zakázek je rozdělen na dvě části. Levá část obsahuje panel pro zobrazení seznamu zakázek a pravá část obsahuje pole pro zobrazení a zadávání údajů zakázek.

Seznam zakázek

Panel pro zobrazení seznamu zakázek se skládá z nástrojové lišty s tlačítky, z filtru pro zadání kritérií, podle kterých se vytvoří seznam kontaktů, a z tabulky zobrazující seznam zakázek odpovídající zadaným kritériím. Kliknutím na položku v tabulce kontaktů se data kontaktů zobrazí ve formuláři v pravé části modulu Zakázek.

Nástrojová lišta seznamu obsahuje tlačítka pro zobrazení seznamu zakázek, tlačítka pro aktualizaci seznamu zakázek (jehož funkčnost je stejná jako v případě seznamu Adresáře) a tlačítka pro otevření seznamu kontaktů. V případě, že uživatel chce vytvořit seznam zakázek pro určitý kontakt, vybere kontakt v seznamu kontaktů a vloží kontakt do formuláře. V případě, že uživatel zná číslo kontaktu, je možné zadat číslo kontaktu přímo do formuláře kritérií, na základě kterých se vytváří seznam zakázek, který se zobrazí v tabulce.

Filtr pro zadání kritérií pro vyhledání kontaktů obsahuje možnosti pro vyhledání podle názvu zakázky, kontaktu, zahájení zakázky a stavu zakázky – zda se jedná o neukončenou nebo ukončenou zakázku. Zadaný kontakt ve filtru je možné zrušit kliknutím na tlačítko /X/ vedle políčka s číslem kontaktu. Zakázku se seznamu v tabulce zakázek je možné zobrazit kliknutím na název zakázky. Po kliknutí na položku zakázky se data zakázky zobrazí v pravé části modulu zakázek.

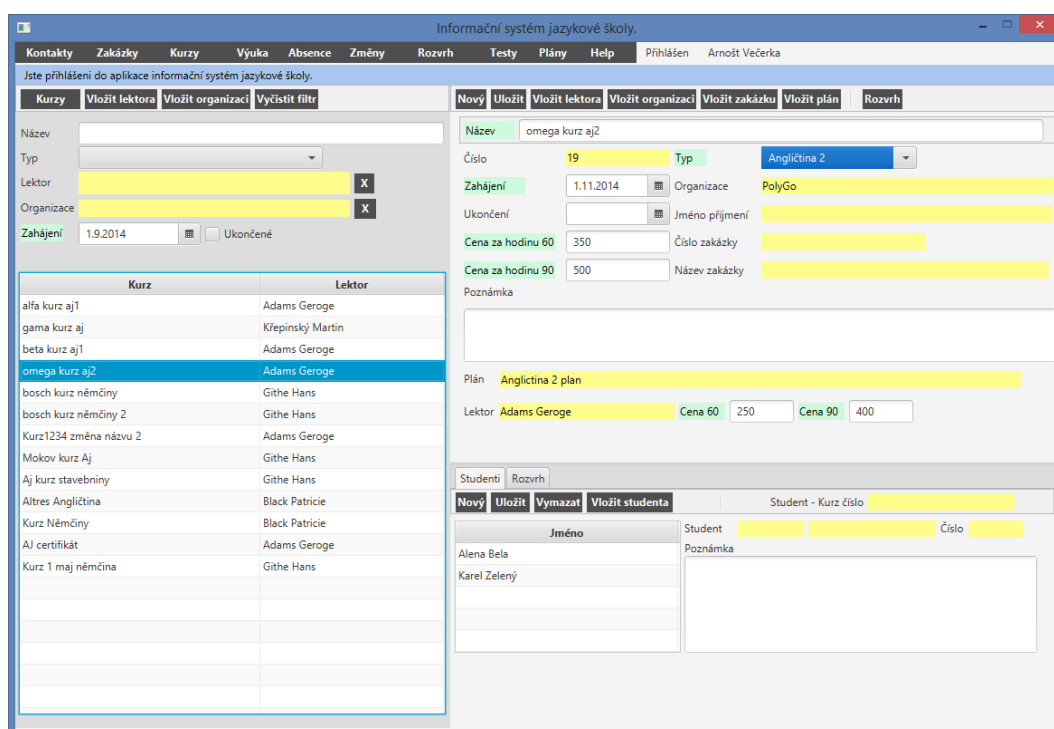
Zadávání zakázek

Formulář zadávání zakázek obsahuje pole pro zadání názvu zakázky, datum začátku zakázky a popis zakázky. Zakázku je možná přiřadit určitému kontaktu (tlačítko vložit kontakt) a v případě ukončení zakázky je možné označit zakázku jako ukončenou. Odebrání kontaktu je možné provést tlačítkem /X/.

8.3.3 Modul Kurzy

Seznam kurzů podle zadaných kritérií se zobrazí kliknutím na tlačítko Kurzy (viz. obrázek 12). Formulář kurzů je rozdělen na dvě části. Levá část obsahuje tabulku se seznamem kurzů a pravá část obsahuje pole pro zobrazení a zadávání kurzů.

Nástrojová lišta dále obsahuje tlačítka Aktualizovat, Vložit lektora, Vložit organizaci a Vyčistit filtr. Tlačítka Vložit lektora a Vložit organizaci otevrou seznam kontaktů, kde je možné vybrat organizaci nebo lektora a vložit do filtru.



Obrázek 12: Aplikace Administrativa - Kurzy

Vymazání hodnot filtru je možné provést tlačítkem Vyčistit filtr. Filtr pro zadání kritérií pro vyhledání kurzů obsahuje možnosti pro vyhledání podle názvu kurzu, typu kurzu, lektora, který kurzy vede, podle organizace, která si kurzy objednala, a také podle data zahájení kurzu. Zadání data kurzu, od kterého se kurzy mají vyhledat, je povinnou položkou filtru pro výběr kurzů. Je možné zobrazit i kurzy ukončené k určitému datu. Datum ukončení kurzu je možné zadat po označení zaškrtnovacího políčka Ukončené. Zadané hodnoty lektora a organizace ve filtru je možné zrušit tlačítky /X/. Po kliknutí na položku v tabulce kurzů se data kurzu zobrazí v pravé části modulu Kurzy.

Zadávání kurzů.

Pole pro zadávání nových kurzů je v pravé části formuláře Kurzy. Pravá část obsahuje nástrojovou lištu a formulář pro zadávání kurzů. Nástrojová lišta obsahuje tlačítka: Nový, Uložit, Vložit lektora, Vložit Organizaci, Vložit zakázku, Vložit plán a Rozvrh. Tlačítko Nový vymaže data v polích formuláře a tlačítko Uložit uloží nový kurz do databáze, případně uloží změny dat existujícího kurzu. Formulář obsahuje pole pro zadání názvu kurzu, typu kurzu, zahájení a ukončení kurzu, cenu účtovanou za hodinu kurzu. V pravém sloupci jsou pole pro vložení kontaktu, který si kurz objednal. V dolní části formuláře jsou pole pro vložení poznámky o kurzu, plánu a lektorovi, který kurz povede. U lektora je pole pro vložení ceny, která se bude platit lektorovi za vedení kurzu. U každého kurzu je třeba uvést název kurzu a datum zahájení kurzu.

Zadání studentů do kurzu

Po založení nového kurzu se zpřístupní část formuláře pro zadávání studentů kurzů a rozvrhu kurzu. Panel pro zadávání studentů obsahuje nástrojovou lištu s tlačítky Nový, Uložit, Vymazat a Vložit studenta. Tlačítko Nový vymaže pole formuláře, tlačítko Uložit uloží nového studenta, tlačítko Vymazat odebere studenta z kurzu a tlačítko Vložit studenta otevře okno s kontakty, kde je možné vyhledat studenta a vložit jej do formuláře. Vlastní formulář obsahuje tabulku se jmény studentů a pole pro vložení studenta. K jednotlivým studentům je možné vložit poznámku. Tlačítko Rozvrh zobrazí panel Rozvrhu, kde je možné si zobrazit rozvrh lektorů nebo rozvrh učeben.

Zadání času a místa výuky

Kliknutím na záložka Rozvrh se zobrazí část pro zadávání hodin rozvrhu. Panel pro zadávání rozvrhu kurzu obsahuje seznamy pro zadání dne, hodiny výuky a místa výuky. Nástrojová lišta obsahuje tlačítka Nový pro vyčištění polí formuláře a tlačítka pro uložení hodiny výuky a tlačítko pro odstranění hodiny výuky.

8.3.4 Modul Výuka

Formulář Výuka obsahuje funkce pro zobrazení přehledů odučených hodin (viz. obrázek 13). Panel modulu Výuka se skládá z nástrojové lišty, filtru tabulky pro zobrazení hodin odučených v kurzu a tabulky pro zobrazení seznamu kurzů souhrnných dat odučených hodin ve zvoleném období.

Povinným parametrem pro zobrazení seznamu odučených hodin je nastavení data, od kterého se má vytvořit přehled odučených hodin. Dalšími volitelnými parametry pro vytvoření přehledu je datum, do kterého proběhla výuka, lektor a organizace. Po zadání parametrů, podle kterých se má zobrazit přehled odučených hodin, a kliknutím na tlačítko výuka, se vytvoří seznam kurzů, ve kterých proběhla výuka v zadaném období. V přehledu jsou zahrnuty hodiny, které byly zadané do systému, a které odpovídají zadaným parametrům. Vytvořený přehled se zobrazí v tabulce, kde u každého zobrazeného kurzu je uveden název kurzu, organizace, lektor kurzu, počet odučených hodin, výnosy a náklady týkající se odučených hodin. Na základě nákladů a výnosů je spočítán hrubý zisk u jednotlivých kurzů. Nad tabulkou jsou pole, ve kterých se zobrazí souhrnné údaje za všechny kurzy v seznamu.

Seznam odučených hodin v jednotlivých kurzech je možno zobrazit kliknutím na položku v tabulce se seznamem kurzů. Přehled odučených hodin v označeném kurzu se zobrazí v horní tabulce. Detailní informace o odučené hodině se zobrazí kliknutím na položku v tabulce odučených hodin. Toto kliknutí otevře panel, ve kterém se zobrazí informace o obsahu výuky v označené hodině.

Informační systém jazykové školy

Kontakty Zakázky Kurzy Výuka Absence Změny Rozvrh Testy Plány Help Přihlášen Arnošt Večerka

Jste přihlášení do aplikace informační systém jazykové školy.

Výuka Zadat lektora Zadat firmu

Filtr OD 29.9.2014 DO Lektor Firma X X

Kurz: Kurz1234 změna názvu 2 Firma: PolyGo Lektor: Adams Geroge

Datum	Cel hodiny
17/11/2014	3 hodina kurzu
24/11/2014	4 hodina kurz 1234
01/04/2015	1 hodina v dubnu
08/04/2015	Hodina 4 dubna
15/04/2015	Hodina 15.4 gramatika

Výuka

Celkem Příjmy 7900 Celkem Náklady 5800 Celkem Zisk 2100 Celkem Hodin 17

Číslo kurzu	Název kurzu	Počet	Výnosy	Náklady	Profit
18	beta kurz aj1	4	2000	1600	400
19	omega kurz aj2	2	850	650	200
20	bosch kurz němčiny	2	1000	800	200
22	Kurz1234 změna názvu 2	6	2550	1950	600
24	Mokov kurz Aj	1	500	400	100
26	Altres Angličtina	1	500	400	100
28	AJ certifikát	1	500	0	500

Obrázek 13: Aplikace Administrativa - Výuka

8.3.5 Modul Absence

Formulář Absence umožňuje zobrazit seznam studentů a jejich absenci v zadaném období. Přehled absence se vytvoří kliknutím na tlačítko Absence. V tabulce přehledu absence se zobrazí seznam studentů a počet zameškaných hodin. Kliknutím na položku v tabulce se zobrazí informace o zameškaných hodinách konkrétního studenta.

8.3.6 Modul Změny ve výuce

Formulář Změny obsahuje funkce pro zadávání suplovaných hodin v kurzech v případě, že lektor kurzu nemůže zajistit výuku v dané hodině. Je možné na tuto hodinu zadat náhradního lektora, který zajistí výuku. Panel modulu změny obsahuje nástrojovou lištu, formulář pro zadávání změn výuky a tabulky pro zobrazení seznamu suplovaných hodin. Nástrojová lišta obsahuje tlačítka Nová změna, Uložit, Vymazat, Zadat lektora a Rozvrh.

Zadání změny

Pro zadání změny je třeba zadat kurz, čas a místo výuky, datum změny a lektora, který zajistí výuku v suplované hodině, u lektora je nutné uvést cenu lektora. Zadání kurzu, času a místa výuky se provede prostřednictvím tlačítka Rozvrh. Kliknutím na tlačítko Rozvrh se zobrazí rozvrh, kde je možné označit suplovanou hodinu a kliknutím na tlačítko OK ji vložit do formuláře pro zadávání suplova-

ných hodin.

Zobrazení suplovaných hodin

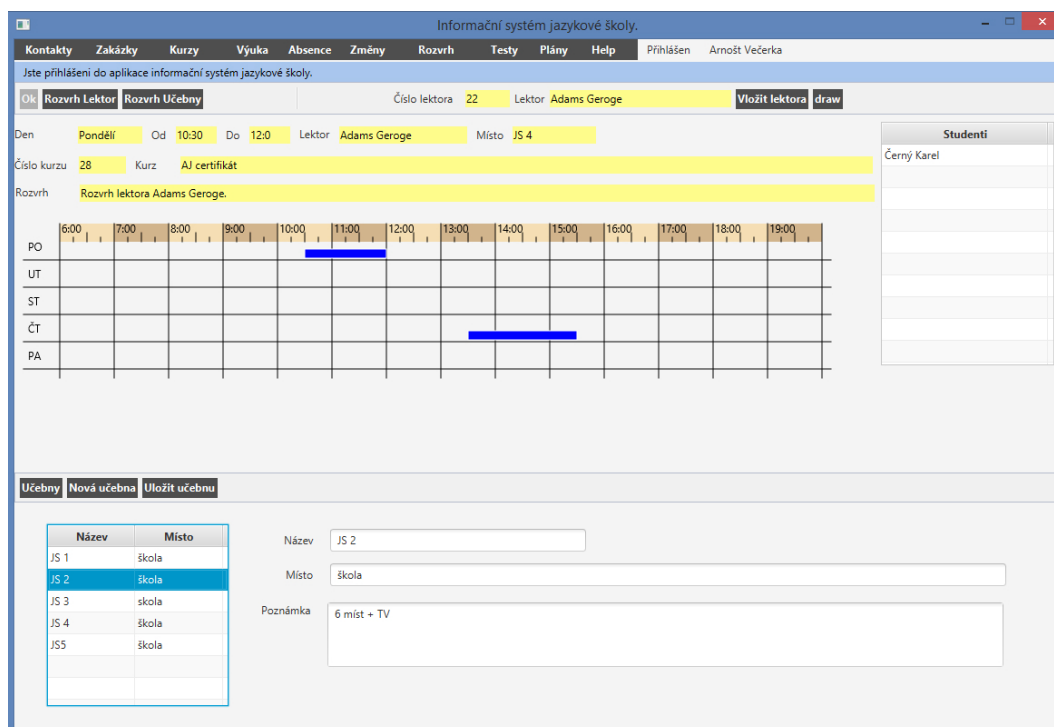
Přehled suplovaných hodin se zobrazí kliknutím na tlačítko suplované hodiny. Pro vytvoření přehledu je nutné označit datum, od kterého se má zobrazit seznam odučených hodin.

Odstranění změny

Suplovanou hodinu je možné odstranit označením suplované hodiny a kliknutím na tlačítko Vymazat. Označení suplované hodiny se provede kliknutím na suplovanou hodinu v tabulce suplovaných hodin. Suplovanou hodinu je možné vymazat pouze tehdy, jestliže nebyla zapsána lektorem. V případě, že hodina byla již odučena a lektor ji zapsal, tak není možná suplovanou hodinu odstranit.

8.3.7 Modul Rozvrh

Formulář Rozvrh obsahuje funkce pro zobrazení rozvrhu lektorů a učeben (viz. obrázek 14). Současně je součástí modulu Rozvrh funkce pro zadávání nových učeben.



Obrázek 14: Aplikace Administrativa - Rozvrh

Zobrazení rozvrhu lektora

Kliknutím na tlačítko Vložit lektora se zobrazí seznam kontaktů, kde se označí

lektor a přes tlačítko OK se vloží do modulu Rozvrh. Kliknutím na tlačítko Rozvrh lektora se pro aktuálního lektora vygeneruje rozvrh. Do rozvrhu jsou zahrnuty všechny kurzy, které mají přiřazeného označeného lektora, a které nemají vyplněný datum ukončení.

Zobrazení rozvrhu učebny

Určení učebny pro zobrazení jejího rozvrhu se provede v dolní části modulu, kde je tabulka se seznamem učeben. Učebny se v tabulce zobrazí kliknutím na tlačítko Učebny. V tabulce se označí učebna kliknutím na řádek s názvem učebny. Je-li učebna označena, je možné pro ni vygenerovat rozvrh kliknutím na tlačítko Rozvrh učebny.

Zobrazení informací o výukové hodině v rozvrhu.

Kliknutím na položku v tabulce rozvrhu se zobrazí informace o výukové hodině. To znamená, že se zobrazí název kurzu, čas výuky a lektor, který daný kurz vede. V tabulce studentů se zobrazí seznam studentů, kteří v daném kurzu studují.

Zadání nové učebny

Novou učebnu je možné zadat v dolní části modulu Rozvrh. V případě, že je vybraná učebna v seznamu učeben, je nutné nejprve zrušit výběr kliknutím na tlačítko Nová učebna a následně vyplnit informace o učebně. Nová učebna se vytvoří kliknutím na tlačítko Uložit učebnu. Kliknutím na tlačítko Uložit učebnu se uloží data o nové učebně, případně se aktualizují data učebny, která byla již zadána dříve.

8.3.8 Modul Testy

Formulář Testy obsahuje funkce pro vytváření testů a pro zobrazení a prohlížení již vytvořených testů. Vlastní přidělování testů studentům a jejich kontrolu provádí lektori přes webové rozhraní lektora. Testy, které je možné vytvořit, jsou úlohy s výběrem odpovědi. Test je tvořen sadou otázek, kde ke každé otázce jsou přiřazeny čtyři možnosti. Při zadávání testu jsou zadány otázky, možné odpovědi a jedna z možných odpovědí je označena jako správná. Test může mít libovolné množství otázek.

Vytvoření nového testu.

Nový test je možné zadávat přímo v případě, že nejsou zobrazena data existujícího testu. V případě, že je zobrazen existující test, je nutné kliknutím na tlačítko Nový test vymazat data existujícího testu. Před uložením nového testu je nutné označit jazyk testu, úroveň testu, zadat název testu a vyplnit zadání testu. Volitelně je možné zadat čas testu, po který budou studenti moci odpovídat na otázky testu. Po uložení nového testu je možné zadávat otázky testu.

Otázky testu

Zadání nové otázky v případě, že je zobrazena otázka testu je nutné kliknout na

tlačítko Nová otázka. Kliknutím na tlačítko Nová otázka se vymažou pole pro zadání otázky a pole pro zadání možných odpovědí. Ruční vymazání, případně přepis odpovědí existující otázky bez kliknutí na tlačítko Nová otázka a následné uložení, přepíše existující otázku a nedojde k vytvoření nové otázky. Po vyplnění polí otázky se otázka uloží kliknutím na tlačítko Uložit otázku. Prohlížení otázek testu je možné tlačítky Další otázky a Předcházející otázka.

8.3.9 Modul Plány

Formulář Plány umožňuje zadat základní informace o obsahu výuky v kurzech. Takto vytvořené plány je možné přiřadit jednotlivým kurzům.

Zobrazení seznamu plánů

Seznam již vytvořených plánů se zobrazí kliknutím na tlačítko Plány. Konkrétní plán se zobrazí kliknutím na název plánů v tabulce, ve které je zobrazen seznam plánů.

Vytvoření plánu

V případě, že je označen plán v tabulce obsahující seznam plánů, je nutné kliknout na tlačítko Nový plán. V případě, že není označen plán v tabulce plánů, je možné zadávat plán přímo. Po vyplnění názvu a obsahu plánu se nový plán uloží kliknutím na tlačítko Uložit plán. Stejným způsobem je možné uložit i změny v již existujícím plánu.

8.3.10 Modul Náповěda

Náповěda obsahuje informace o funkcích aplikace. Po kliknutí na tlačítko Náповěda se zobrazí okno s náповědou.

9 Další možnosti rozvoje systému

Vytvořený informační systém ve velké míře podporuje proces výuky a související procesy v jazykové škole. Vzhledem k velkému rozsahu činností jazykové školy nejsou pokryty všechny činnosti jazykové školy, například oblast spolupráce jazykové školy a firemních zákazníků. Stejně tak i řadu funkcí informačního systému je možné dále rozvíjet. Například oblast testování, kde je možné realizovat funkce umožňující vytváření komplexnějších testů jazykových znalostí. Další možností rozvoje informačního systému je vytvoření modulu pro customizaci informačního systému. Tento modul by mohl obsahovat funkce pro zadávání typu testů, typu kurzů nebo středisek.

Velké možnosti dalšího rozvoje jsou v oblasti analýz. Modul analýz by mohl poskytnout funkce na podrobné analýzy průběhu výuky a její efektivity, analýzy spokojenosti studentů s výukou a informace o vztahu obsahu a způsobu výuky s

dosahovaným pokrokem ve znalostech studentů.

Jednou z nejdůležitějších oblastí činnosti je ekonomická stránka činnosti jazykové školy. V současnosti je tato oblast řešena prostřednictvím účetního informačního systému, a co se týče účetních dokladů, tak tomu bude i nadále. Účetní systém neobsahuje funkce na podrobnější vyhodnocování rentability výuky v jednotlivých kurzech, kde hlavním důvodem jsou chybějící data. Účetní systém obsahuje především agregovaná data na úrovni faktur týkající se výuky. Tato data, neobsahují podrobnější informace o nákladech a výnosech spojených s jednotlivými kurzy a i kdyby obsahovaly, tak zde nejsou nástroje na vyhodnocování těchto dat. Řešení této oblasti by mohly poskytnout funkce informačního systému jazykové školy zaměřené na evidenci a zpracování ekonomických dat.

Závěr

V analytické části práce jsem popsal analýzu prostředí společnosti jazykové školy Enara s.r.o. V analýze jsem se hlavně zaměřil na procesy probíhající ve společnosti, především proces zakázek, výuky a fakturace. Součástí analýzy byl i výzkum názorů pracovníků společnosti týkající se informačních potřeb a výměny informací mezi pracovníky společnosti. Analýza prostředí společnosti byla východiskem pro určení funkcí informačního systému a případů užití informačního systému.

Ve třetí a čtvrté části jsem popsal architekturu informačního systému a použité technologie. Pro informační systém jazykové školy jsem zvolil tří vrstvou architekturu, kde prezentační vrstva je řešena jako samostatná aplikace pro jednotlivé případy užití. Pro skupiny uživatelů studenti a lektori jsem se rozhodl vytvořit webové aplikace v programovacím jazyce AS3 Framework Flex 4.6, a aplikaci určenou pro administrativní pracovníky jsem naprogramoval v jazyce Java Framework JavaFx. Aplikační vrstvu na serveru jsem vytvořil v jazyce Java s využitím technologie EJB3. Databáze je vytvořena v databázovém systému MySql.

V páté části jsem popsal technické řešení informačního systému. Základem technického řešení je Framework GraniteDs a Framework JPA. Framework GraniteDS realizuje přenos dat mezi prezentační vrstvou a aplikační vrstvou Framework JPA umožňuje objektově relační mapování dat v databázi na entity, se kterými pracují EJB komponenty. Dále je v této části popsána struktura jednotlivých aplikací a funkce jednotlivých modulů.

Šestá část obsahuje popis funkcí informačního systému z pohledu uživatele. Základem informačního systému je evidence a zobrazování dat týkající se výuky v jazykových kurzech. Administrativní část informačního systému slouží především pro zadávání jazykových kurzů a souvisejících dat, jako jsou především zakázky a osobní údaje studentů, a vytváření přehledů odučených hodin. Další důležitou součástí je modul pro vytváření testů. Webová aplikace pro lektory slouží pro zadávání odučených hodin a testování studentů. Webová aplikace pro studenty umožňuje studentům jazykových kurzů zobrazit si informace o výuce, což má význam především při absenci studentů ve výuce, a testování znalostí studentů. Testování studentů slouží nástroj poskytující zpětnou vazbu studentům a lektorům o úrovni zvládnutí probírané látky.

V poslední části Další rozvoj informačního systému jsem na závěr popsal možnosti dalšího vývoje, kde největší možnosti a především přínos vidím ve vývoji, který by byl zaměřen na analytickou oblast spojenou s výukou, a evidenci a vyhodnocování ekonomických dat spojených s výukou.

Na závěr lze říci, že vytvořený informační systém poměrně komplexně pokrývá agendu jazykové školy týkající se výuky a přispívá k vyšší efektivitě činnosti jazykové školy. Je pravda, že tato oblast je velice široká a je zde velká příležitost pro další rozvoj informačního systému především v oblasti zpracování a prezentace dat o výuce, což by přispělo k vyšší efektivitě řízení jazykové školy Enara s.r.o a k lepší nabídce služeb pro studenty.

Conclusions

In the analytical part of my thesis, I described the internal environment of language school Enara s.r.o. The analytical part was focused on internal company processes, mainly on processes order, teaching and billing process. As a part of the analysis there was also an employee survey, regarding information needs and information exchange among company employees. The analysis was starting point for a definition of functions of the information system and use cases of the information system.

In the fifth and the sixth part I described an architecture of the information system and technologies that I used. For the information system I chose three tier architecture. The presentation tier is designed as a single application for each use-case of the information system. For users groups students and language teachers I created web applications in programmatic language AS3 and framework Flex 4.6. For users in administration I created desktop application using framework JavaFx. The application tier on the server, I created using technology EJB3. Databases is created in database system MySql.

In the sevenths part, I described the technical solution of the information system. The basis of the solution is framework GraniteDs and framework JPA. Framework GraniteDs realizes data transfer between the presentation tier and the application tier. Framework JPA enables mapping of data in database to entities which are used by EJB components. In this part I also described the structure of the applications and the functions of each module.

The eighth part contains description of information system functions from the user point of view. The basic part of the information system is entering and presentation of data regarding lessons in languages courses. Administrative part of the information system serves mainly for language courses entry and for entry of related data as orders and students personal data. The other important parts are module for lessons reporting and module for test creation.

Web application for students enables students to present information about lessons, that is particularly important when a student misses a lesson, and for testing of students' knowledge. Students' testing serves as a tool to provide feedback to students and lectors how they mastered course content.

In the final part Further development of the information system, I described the possibilities of further development, where I see the biggest contribution in development focused on analytical part related to lessons and to data analysis regarding economic data related to courses.

At the end it is possible to say that the information system covers the

language school agenda related to teaching and contributes to better performance efficiency of the language school Enara s.r.o. It is true that this area is very broad and further development of the information system can contribute to better efficiency of language school management and to better services for students.

Bibliografie

- [1] DI PISA, Filippo, *Beginning Java and Flex*, migrating Java, Spring, Hibernate, and Maven developers to Adobe Flex, New York: Distributed by Springer-Verlag New York, c2009, xvii, 425 p. Expert's voice in Web development. ISBN 978-1-4302-2386-3.
- [2] HARRINGTON, H a James S HARRINGTON, *Total improvement management: the next generation in performance improvement*, New York: McGraw-Hill, c1995, xviii, 488 p. ISBN 0-07-026770-7.
- [3] KOFLER, Michael, *Mistrovství v MySQL 5*, [kompletní průvodce webového vývoje]. Vyd. 1. Brno: Computer Press, 2007, 805 s. Mistrovství. ISBN 978-80-251-1502-2.
- [4] GONCALVES, Antonio, *Beginning Java EE 6 Platform with GlassFish 3*, from novice to professional. 2nd ed., New ed. Berkeley, CA: Apress, 2010. ISBN 978-1-4302-2889-9.
- [5] ŘEPA, Václav. *Analýza a návrh informačních systémů*, Vyd. 1. Praha: Ekopress, 1999, 403 s. ISBN 80-86119-13-0.
- [6] SCHILDT, Herbert. *Java 7*, výukový kurz. 1. vyd. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2.

A První příloha

Dotazník – lektoři

1. Jaké je vaše vzdělání, Současně uveďte zda máte kromě jazykového a pedagogického vzdělání a odborné vzdělání (případně znalosti) i v jiném oboru.

2. V jakém rozsahu využíváte internet, počítač ve výuce? (V případě, že občas uveďte do závorky kolikrát v průměru na jeden kurz za měsíc, např. (2x) znamená asi na dvě hodiny měsíčně v určitém kurzu.

3. Zvládáte uživatelsky Word a Excel?

4. Jaké webové stránky využíváte pro svou práci?

5. Dostáváte včasné a úplné informace potřebné pro vaši práci (pokud ne, blíže specifikujte).

6. Považujete rady společnosti za přínosné?

7. Zajímáte se aktivity jazykové školy

8. Jaké aktivity děláte pro svůj odborný rozvoj?

9. Měli byste zájem o aktivity zaměřené na vaše vzdělávání organizované jazykovou školou?

10. Co by jste změnilý na způsobu vedení výuky jazykové školy :

11. Jaké jsou, podle vás, hlavní nedostatky současného informačního systému jazykové školy:

12. . Jakým způsobem by jste doplnili, změnilí současný informační systém jazykové školy?

13. Co by jste změnilí na způsobu spolupráce mezi lektorem a jazykovou školou?

14. Uveďte jakékoliv vaše další náměty, připomínky k činnosti jazykové školy, vaší práci cokoliv si myslíte, že by se mělo řešit:

B Obsah přiloženého CD/DVD

bin/

V adresáři bin se nachází soubory webové aplikace Student a Lektor a desktopové aplikace pro administrativu JulieFx, obsahující spustitelná jar soubor. Soubory pro instalaci na server jsou v adresáři server a obsahují WAR soubory pro instalaci na javovský server Glassfish 3.1.2.

doc/

Text bakalářské práce ve formátu PDF, a soubory potřebné pro vygenerování PDF dokumentu textu (v ZIP archivu)-

src/

Adresář src obsahuje zdrojové soubory webové aplikace Student a Lektora a zdrojové texty aplikace pro administrativu JulieFx. Zdrojové soubory serverové části informačního systému jsou v adresáři server.

readme.txt

Soubor obsahuje instrukce pro instalaci a spuštění programu INFORMAČNÍHO SYSTÉMU JAZYKOVÉ ŠKOLY, a webová adresa, na které je aplikace nasazena.

Navíc CD/DVD obsahuje:

data/

Adresář obsahuje informace o testovací data použitých pro testování informačního systému.

