



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF INFORMATICS

NÁVRH INFORMAČNÍHO SYSTÉMU

DESIGN OF AN INFORMATION SYSTEM

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MARTIN HORÁK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PETR DYDOWICZ, Ph.D.

BRNO 2015

ZADÁNÍ DIPLOMOVÉ PRÁCE

Horák Martin, Bc.

Informační management (6209T015)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává diplomovou práci s názvem:

Návrh informačního systému

v anglickém jazyce:

Design of an Information System

Pokyny pro vypracování:

Úvod

Vymezení problému a cíle práce

Teoretická východiska práce

Analýza problému a současné situace

Vlastní návrh řešení, přínos práce

Závěr

Seznam použité literatury

Seznam odborné literatury:

BASL, J. a R. BLAŽÍČEK. Podnikové informační systémy. Podnik v informační společnosti. 1. vyd. Praha: Grada, 2008. 283 s. ISBN 978-80-247-2279-5.

MOLNÁR, Z. Automatizované informační systémy. 1. vyd. Praha: Strojní fakulta ČVUT, 2000. 126 s. ISBN 80-01-02269-2.

MOLNÁR, Z. Efektivnost informačních systémů. 1. vyd. Praha: Grada Publishing, 2000. 142 s. ISBN 80-7169-410-X.

ŘEPA, V. Analýza a návrh informačních systémů. 1. vyd. Praha: Ekopress, 1999. 403 s. ISBN 80-86119-13-0.

SODOMKA, P. a H. KLČOVÁ. Informační systémy v podnikové praxi. 2. aktualiz. a rozš. vyd. Brno: Computer Press, 2010. 501 s. ISBN 978-80-251-2878-7.

Vedoucí diplomové práce: Ing. Petr Dydowicz, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2014/2015.

L.S.

doc. RNDr. Bedřich Půža, CSc.
Ředitel ústavu

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
Děkan fakulty

V Brně, dne 28.2.2015

Abstrakt

Cílem této diplomové práce je návrh informačního systému, zaměřujícího se na správu koncertů a vystoupení hudebního tělesa. Informační systém je navržen ve spolupráci se společností Limeta Apps s.r.o.

Abstract

The aim of the master thesis is design of an information system for concert management of music ensembles. The information system is developed in cooperation with LimetaApps s.r.o.

Klíčová slova

Návrh informačního systému, Informační systém, PHP, MySQL, DF diagramy, ER diagramy, databáze

Key words

Design of an Information systém, Information systém, PHP, MySQL, DF diagrams, ER diagrams, database

Bibliografická citace práce

HORÁK, M. *Návrh informačního systému*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2015. 75 s. Vedoucí diplomové práce Ing. Petr Dydowicz, Ph.D.

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně.

Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 25. května 2015

.....

podpis

Poděkování

Tímto bych chtěl poděkovat všem, kteří mi poskytli cenné informace, rady a doporučení. Zejména vedoucímu mé práce Ing. Petru Dydowiczovi, Ph.D. a mému oponentovi Ing. Lukáši Horákovi.

Obsah

ÚVOD	10
CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ.....	11
1 TEORETICKÁ VÝCHODISKA PRÁCE	13
1.1 Informace a data	13
1.2 Informační systém	13
1.2.1 Účel informačního systému.....	15
1.2.2 Způsoby implementace informačního systému.....	16
1.2.3 Rozvoj existujícího řešení	17
1.2.4 Vývoj informačního systému na míru.....	17
1.2.5 Koupě existujícího informačního systém.....	17
1.3 Databáze	18
1.3.1 Historie databází.....	18
1.3.2 Databázové modely	19
1.3.3 Relační databáze.....	20
1.3.4 Omezení integrity	21
1.3.5 Vztahy integritních omezení	22
1.3.6 Normální formy.....	24
1.4 SWOT analýza.....	25
1.5 SQL.....	26
1.6 Použité technologie.....	27
1.6.1 HTML.....	27
1.6.2 CSS.....	28
1.6.3 JavaScript	29
1.6.4 PHP.....	31
2 ANALÝZA SOUČASNÉ SITUACE.....	32
2.1 Informační systém složený z několika komponent.....	34

2.1.1	SWOT analýza	36
2.1.2	Zhodnocení systému.....	36
2.2	Informační systém kapely Boršičanka.....	36
2.2.1	SWOT analýza	38
2.2.2	Zhodnocení systému.....	38
2.3	Interní web kapely Lácaranka.....	38
2.3.1	SWOT analýza	42
2.3.2	Zhodnocení systému.....	43
2.4	Celkové zhodnocení současného stavu.....	43
3	VLASTNÍ NÁVRHY ŘEŠENÍ.....	44
3.1	Výběr technologií	44
3.2	Výběr pluginů	45
3.2.1	Javascriptové pluginy	45
3.2.2	PHP pluginy	45
3.3	Struktura informačního systému KAPELis	46
3.4	Bezpečnostní opatření.....	47
3.4.1	Přihlášení uživatele	47
3.4.2	Role	48
3.5	Návrh databáze	50
3.6	DFD diagram	51
3.7	Vývojový diagram	52
3.8	Komponenty informačního systému.....	53
3.8.1	Hlavní menu	53
3.9	Diskuze	55
3.10	Moduly.....	57
3.10.1	Osobní deska	57
3.10.2	Přehled budoucích akcí	58

3.10.3	Detail akce.....	58
3.10.4	Vytváření nové akce / editace stávající akce.....	60
3.10.5	Finance akce.....	61
3.10.6	Mé finance.....	61
3.10.7	Kontakty.....	63
3.10.8	Přehled všech akcí a Celkové finance.....	63
3.11	Ekonomické zhodnocení.....	64
3.11.1	Outsourcingová varianta.....	64
3.11.2	Varianta jednorázové implementace.....	66
3.11.3	Výnosy outsourcingu.....	66
3.11.4	Výnosy jednorázové implementace.....	67
3.12	Celkové zhodnocení.....	67
3.12.1	Silné stránky.....	67
3.12.2	Slabé stránky.....	68
3.12.3	Příležitosti.....	68
3.12.4	Hrozby.....	68
3.12.5	Outsourcing vs. Implementace.....	68
	ZÁVĚR.....	69
	SEZNAM POUŽITÉ LITERATURY.....	70
	SEZNAM OBRÁZKŮ.....	73
	SEZNAM TABULEK.....	75

ÚVOD

Dnes si život bez informačních systémů dokážeme jen stěží představit. Spousta lidí si to možná ani neuvědomuje, ale informační systémy jsou všude kolem nás. Představme si obyčejnou cestu studenta do menzy. Hned na začátku, kdy si student vybírá z nabídky, která svítí na LCD, narážíme na první informační systém. Další na nás čeká u pokladny, kde pokladní zadá do počítače naše jídlo a vybere peníze. Zkrátka s informačními systémy jsme ve styku každou chvíli a ani o tom možná nevíme nebo si to neuvědomujeme.

V současné době je čas stále dražší a i z tohoto důvodu nám informační systémy slouží. Dokážou nám ušetřit nejen finanční zdroje, zpřehlednit údaje. Dovedou nám hlavně ušetřit čas.

V mé práci se chci zaměřit na vývoj nového informačního systému, který by napomohl k řízení hudebního tělesa po manažerské stránce. Měl by však sloužit nejen manažerům, ale i dalším členům, kteří by nemuseli dostávat důležitá data o zkouškách, koncertech mailem nebo prostřednictvím SMS. Tento způsob je možný pouze, pokud je hudební těleso malé a má pár vystoupení za rok. V momentě, kdy je však těleso větší a má několik vystoupení za měsíc se stává situace pro manažera velmi nepřehlednou a potřebuje mít přehled o všech hráčích, kteří na akci vystupují.

Zaměřím se tedy na analýzu současného stavu, zda nějaký dostatečně efektivní systém na řízení hudebního tělesa existuje a pokud ne, pokusím se takový informační systém sestavit. Součástí návrhu bude i zachycení ekonomických hledisek při zavedení systému do běžného fungování.

CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ

I přes nesčetné množství počítačových informačních systémů se stále najdou obory, pro které doposud nebyl vytvořen žádný efektivní počítačový informační systém. Jedná se o obory, které jsou vysoce specifické a mají své vlastní požadavky na provoz. Jedním z těchto oborů je i obor řízení hudebních těles.

Hudební těleso (pod tímto pojmem si představme např. symfonický orchestr, dechový orchestr ale i tělesa menšího charakteru) se pohybuje ve specifickém marketingovém prostředí. Jedním z těchto specifíků je i nedostatek finančních prostředků věnovaných na řízení takového hudebního tělesa. Z tohoto důvodu je v tomto oboru vysoký nedostatek informačních systémů, které by mohly sloužit k usnadnění řízení těchto těles.

Cílem mé práce je tedy vytvořit pokročilý evidenční systém hudebního tělesa. Systém by měl být použitelný jak pro správce (manažery, kapelníky, vedoucí) tak i pro samotné uživatele (hráči, výpomoci apod.). Měl by umět evidovat nejen hudební vystoupení, ale i správu financí, možnost alternování jednotlivých uživatelů, evidovat kontakty na alternující muzikanty a mít schopnost vést diskuzi k jednotlivým hudebním vystoupením.

Systém by tak šetřil čas a finance všem členům hudebního tělesa – odpadlo by neefektivní emailové domlouvání či textování pomocí SMS. Zároveň by byl pod jednou střechou nejen organizační faktor systému, ale i finanční. Pro každého člena by šla spravovat finanční ohodnocení za odehrané vystoupení. Tím by se daly vytvářet měsíční přehledy a ušetřil by se tím i čas finančnímu manažerovi popř. účetnímu.

Začátkem mé diplomové práce se budu soustředit na vysvětlení pojmů souvisejících s informačními a databázovými systémy. K tomu se pokusím vysvětlit základní principy analýz informačních systémů.

Poté provedu analýzu současného stavu, ve které použiji zmíněné metody, které jsem zmínil v předchozí kapitole. Pokusím se vystihnout hlavní kritéria, na základě kterých budu současné systémy hodnotit a u každého systému provedu malé shrnutí.

V následující části diplomové práce se budu věnovat již samotnému návrhu informačního systému, jeho dílčích částí a jeho databáze. Vše budu postupně popisovat.

Součástí bude i ekonomická analýza systému, ve které se pokusím vykalkulovat náklady na vytvoření, ekonomické zhodnocení pro prodej, samotný obchodní model prodeje a požadavky na provoz.

V závěru práce provedu celkové zhodnocení navrhnutého informačního systému.

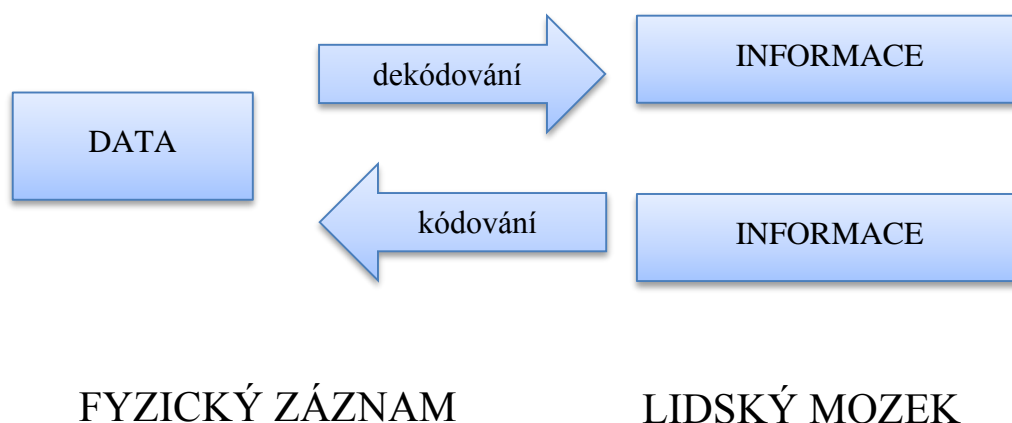
1 TEORETICKÁ VÝCHODISKA PRÁCE

Tato kapitola shrne základní pojmy, které se zabývají problematikou a vývojem informačních systémů.

1.1 Informace a data

„Informace je zpráva (sdělení), která snižuje neurčitost (entropii) příjemce“ (1, s. 10).

Pod pojmem data můžeme rozumět nějaké zaznamenané informace, které jsou schopny přenosu a dalšího zpracování. Pokud nad informací provedeme nějakým způsobem přenos do počítače, dá se prohlásit, že přenesené informace v počítači jsou data. Jestliže tato data dekódujeme zpět, jsme schopni dostat informaci.



Obrázek 1: Data a informace

Upraveno dle (1)

1.2 Informační systém

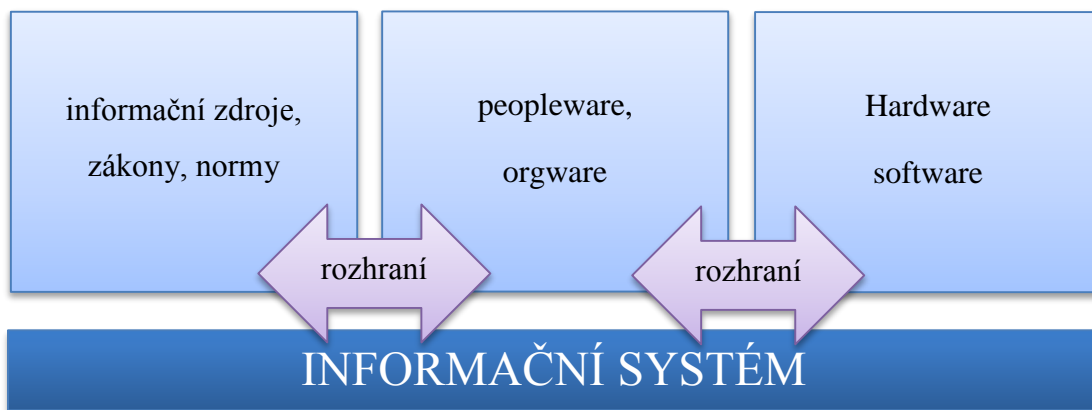
O informačním systému dle všeobecné definice můžeme prohlásit, že tento systém se dynamický a skládá se z různých prvků a vazeb mezi nimi samotnými. Pro lepší představivost si můžeme pomoci představou, že informace jsou v tomto případě vazby a prvky v systému tvoří místa transformací těchto informací (2).

Pojem informační systém však nemusí nutně znamenat souvislost s počítači, tedy hardwarem či softwarem. Naopak. Můžeme si pod tímto pojmem vyložit i zápisník – diář, kalendář nebo kartotéku u lékaře a to proto, že všechny tyto nástroje účelově zpracovávají data.

Dnes by se dalo bez informačních technologií v moderním světě fungovat jen stěží. Dovedu si dokonce tvrdit, že by to znamenalo totální chaos a možná i katastrofu. Informační systémy jsou totiž dnes všude kolem nás.

Podle Tvrdíkové je informační systém složen z několika faktorů (4):

- Hardware (technické prostředky)
- Software (programové prostředky)
- Orgware (organizační prostředky)
- Peopleware (lidské složky)
- Reálný svět (informační zdroje, zákony, normy..)



Obrázek 2: Faktory se vzájemnými vazbami v informačním systému

Upraveno dle (1)

Aby byl informační systém efektivní a účelný, dle Dohnala musí splňovat následující podmínky (3):

- Informacemi pro informační systém jsou pouze ta data, která dokážeme využít a přiřadit jim nějaký význam či smysl.
- Musí umět pracovat s daty (sbírat, uchovávat, editovat, kontrolovat)
- Podpora počítače je žádoucí, ne však nutná

Současně s předchozími podmínkami na systém by měl také splňovat tyto i následující vlastnosti (3):

- Integrovaný
- Otevřený, pružný
- Nezávislý, konzistentní
- Standardizovaný
- Adaptabilní
- Přístupný
- Stabilní, bezpečný
- Stálý (s dlouho životností)
- Ergonomický, jednoduchý
- Účelný

1.2.1 Účel informačního systému

Úloha informačního systému je různá. Liší se míra a úroveň samotného výstupu informačního systému a to už například ve smyslu přesnosti, kvality, komplexnosti nebo jiných měřítek. Různé jsou i míry zatížení dílčích částí v procesech, které se v informačním systému odehrávají. Důležité však je, že zůstává samotný výstup z informačního systému a tím je informace – tedy odpovědi.

V celkovém kontextu tedy informační systém provádí funkci, díky které vzniká výstup, na jehož základě se provádí rozhodování, ze kterého plyne užitek.

Funkce informačního systému, tedy procesy podporující konečný cíl (7):

- získávání informací
- zpracování informací
- uložení informací
- přenos informací
- zpřístupnění informací

Zavést informační systém do podniku však není nutností. Implementace informačního systému se doporučuje pouze v případě, kdy bude mít zavedení pozitivní vliv na zvýšení efektivity práce

lidí, tedy že z něj bude plynout lidem a firmě užitek. Podoba užitku se pro každého uživatele systému liší (8, s. 102)

- vlastníci společnosti – dochází ke zhodnocení vloženého majetku
- manažeři – napomáhá k úspěšnému řízení podniku
- zaměstnanci – vytváří se lepší pracovní prostředí, vyšší společenský status, napomáhá ke zjednodušení a zefektivnění pracovních procesů
- zákazník – produkt s vyšší přidanou hodnotou za příznivou cenu (9, s. 2)

Protože se klade důraz na kvalitativní přístup k užitku, není zde možné přesně vyčíslit užitek přínosu informačnímu systému. Z tohoto důvodu nejde přesně určit, zda bude pro firmu zavedení informačního systému účelné a efektivní. Proto je samotné zhodnocení účelnosti a účinnosti velmi složitý úkol.

1.2.2 Způsoby implementace informačního systému

Z historického hlediska existuje několik způsobů, jak řešit potřebu řízení informačního systému. V osmdesátých letech byly charakteristické přístupem řešení programování vlastních informačních systémů, které byly navrženy ve spolupráci s podnikovými analytiky a programátory. Systém jimi navržen na základě požadavků a potřeb uživatelů.

Pozvolný zvrát nastal počátkem devadesátých let. I když stále v této době existovala ve spoustě podniků preference vlastního řešení vývojem vlastního nebo rozvojem stávajícího informačního systému, postupně se začala probíjet varianta realizace informačních systémů nákupem již hotového, připraveného a snadno nastavitelného informačního systému. Tato metoda řešení podnikových informačních systémů se uchovala dodnes a tvoří většinu dnešních informačních systémů. (6, str. 54–56).

V následujícím odstavci jsou popsány a shrnuty klady a zápory řešení informačních systémů.

1.2.3 Rozvoj existujícího řešení

Toto řešení se může stát efektivním konečným řešením u spousty firem. Firma totiž může maximálně využít existující zdroje a investice. Využije existence stávajícího informačního systému a provede jen jeho inovaci. Tato inovace je i z krátkodobého hlediska lacinější a rychlejší na provedení. Zároveň tím uspokojí okamžité potřeby.

Nese sebou však riziko časové trvanlivosti. Do budoucna totiž nemusí odpovídat všem požadavkům a nárokům na systém samotný. Pokud se jedná o starší informační systém, náklady na upgrade mohou být vyšší a možnost koupě nového informačního systému může být výhodnější. Výsledek samotného upgradovaného informačního systému nemusí být vždy kvalitní a zdokonalený systém. (6, str. 54)

1.2.4 Vývoj informačního systému na míru

Řešení formou programování na míru je pro firmu vždy nejefektivnější, bohužel z drtivé většiny nejdražší. Firma sama přesně stanoví, co potřebuje a zkušení analytici a programátoři dle toho vyrobí systém na míru. Firma si může i vývoj řídit a dohlížet na něj.

Jak zde již bylo zmíněno, jedná se o nejdražší variantu, kterou si nemůže dovolit každý podnik. Časově firmu může vývoj zatížit a stále zde existuje hrozba špatné analýzy a z ní vzniklý nekvalitní informační systém, který firmě ve výsledku nepomůže tak, jak firma očekávala. (6, str. 54)

1.2.5 Koupě existujícího informačního systém

V dnešní době existuje nepřeberné množství informačních systémů různého řazení. Z těchto existujících řešení si firma může některý vybrat a zakoupit. Z dlouhodobého hlediska, kdy informační systém je dlouho trvanlivý se jedná o finančně méně náročnou variantu. Spolu s tím přichází i ušetření času při zavedení. Jelikož se jedná o produkt odzkoušený více uživateli a prošel určitým historickým vývojem, je zde pravděpodobné, že se bude jednat i o funkční systém, který se bude i nadále vyvíjet. (6, str. 54)

1.3 Databáze

Pod pojmem databáze si můžeme představit soubor uspořádaných dat, která popisují reálný svět.

Databáze slouží primárně jako nástroj, díky kterému se snadněji a efektivněji extrahují informace. Dalším účelem je i snazší komunikace, respektive distribuce dat a informací.

Po převodu informací reálného světa do dat databáze (strukturalizaci) přináší databáze nesčetné množství operací nad těmito daty a to jak od základních (přehled nad daty), tak i různé selekce, pohledy, procedury a další nástroje. Díky uspořádání se z nepřehledných nestrukturovaných dat stává přehledný systém informací, ze kterého můžeme daleko snadněji čerpat informace a znalosti, které by nám normálně zůstaly skryty a pomocí níž se můžeme lépe zorientovat při rozhodování.

1.3.1 Historie databází

Historie organizování dat sahá až do 18. století, kdy objevitelem této filosofie byl Carl Linnaeus, který vynalezl svým způsobem první metadata v podobě lístků. (10) V této době samozřejmě neexistovaly databáze tak, jako je známe teď. Existovaly formou kartoték, kde nosičem dat byl papír (pomineme-li historicky starší záznamové techniky). S touto metodou se můžeme stále setkat ve formalizované podobě u katalogů knihoven – autorem je Melvil Dewey. (2. polovina 19. století (11))

Dalším mezníkem byl vynález děrných štítků. Tyto štítky předcházely elektronickým datovým nosičům. Pomocí elektromechanických strojů byla data zaznamenávána přímo na štítek. Mezi první zpracování dat na děrných štítcích se považuje sčítání lidu v roce 1980 v USA (12). Muž, jenž stvořil automat, který umožnil toto děrování, se jmenuje Hermanem Hollerithem (13).

Mezi první jazyky pro hromadné zpracování dat se řadí jazyk COBOL (Common Business – Oriented Language). Tento jazyk byl výsledkem práce skupiny Data Systems Languages a to v roce 1960. Tuto skupinu ustavilo tehdejší ministerstvo obrany USA pro standardizaci softwarových aplikací. (13)

O deset let později přichází na svět jedna z prvních relačních databází. Zasloužil se o to Edgar Frank Codd svým článkem, který nazval „A Relational Model of Data for Large Shared Data Banks“ (13). O čtyři roky po té vzniká i první verze dotazovacího jazyka SQL (Structured Query Language) (12).

Poté se rozvoj jazyka připisuje především společnosti IBM a to do doby, než se v roce 1980 dostala na trh první SQL databáze od firmy Oracle (13).

1.3.2 Databázové modely

Relační model se řadí mezi nejrozšířenější modely databází. Nepatří však k jediným. Mezi nejznámější databázové modely se řadí (14)

- Lineární model
- Hierarchický model
- Síťový model
- Relační model
- Objektový model

První jmenovaný model – lineární – představuje množinu souborů záznamů, které jsou uspořádány dle předmětu zájmu. Zároveň platí, že mezi jednotlivými soubory záznamů nejsou žádné vazby. Jako typický příklad si můžeme vybavit kartotéku pacientů u lékaře. (14, s. 21-22)

Dalším modelem je model Hierarchický. Ten je na rozdíl od lineárního modelu obohacen právě o vazby nadřazenosti neboli rodičovské segmenty a vazby podřazenosti neboli dětské segmenty. Pointery – tedy ukazatelé – nám pak znázorňují samotné vazby mezi nimi. Bohužel, právě tito ukazatelé značně omezují možnosti pohledu na samotná data, protože fungují pouze ve směru rodič → potomek, nikoliv však naopak. Přestavme si pacienta (rodič), který trpí určitými nemocemi (potomek). Pointer v tomto případě zamezuje prohlížení na všechny nemoci všech pacientů, povoluje pouze směr pacient → nemoci (14, s. 23).

Mezi další modely se řadí model síťový. Ten je obdobou modelu hierarchického, avšak v tomto modelu jsou odstraněny zmíněné omezující rodičovské vztahy a pointery tak mohou

vést libovolně mezi segmenty. Díky tomu jsou značně rozšířeny možnosti prohlížení dat. (14, s. 23)

Tyto zmíněné modely jsou dnes již méně používané. I když nutno podotknout, že se využívají ještě stále na osobních počítačích. Avšak pro potřeby obchodního průmyslu jsou nevyužitelné. K tomuto účelu se dnes používají spíše následující modely. (14, s. 23)

Za nejpoužívanější model současnosti je považován relační model. Tento model je spojením lineárních modelů, které k sobě navzájem vážou relační klíče. Spojení tohoto typu nejsou konstantní, vznikají jen na dotaz uživatele. Každý z dílčích segmentů obsahuje vlastní unikátní identifikační klíč, přes který se tvoří spojení do dalšího segmentu. Další segment obsahuje stejnou položku, která má stejný datový typ. (14, s. 23)

A nakonec model nejnovější. Tím je považován model objektový. V něm jsou za věty považovány objekty (např. sportovec) a nad těmito objekty jdou definovat metody (např. záznam nového sportovce). Za samotné metody pak považujeme instance objektu (např. osobní údaje o mé osobě). Díky použití unikátních identifikátorů objektů pak můžeme mezi nimi vést vazby jako v síťovém modelu nebo vazby relační vést jako v relačním modelu. (14, s. 24-25)

1.3.3 Relační databáze

Na množinu hodnot uspořádaných do tabulek (řádků a sloupců) nahlíží datový model. Dle pohledu se pak liší i samotná terminologie (14, s. 26)

- Pohled aplikační
- Pohled množinový
- Pohled relační

Dle pohledu aplikačního se jedná o slovíčka – položka, hodnota údaje, věta, struktura věty a tabulka. Z pohledu množinového pak – atribut, doména atributu, entita, atributy entity, množina entit a z pohledu relačního – atribut, hodnota atributu, n-tice relace, schéma relace a relace. (14, s. 26)

Jednotlivé atributy mohou být (14, s. 18):

- Jednoduché (nedělitelné)

- Složené (atribut s více různorodými hodnotami, např. psč + město)
- Vícehodnotové (atribut s více stejnorodými hodnotami, např. více telefonních čísel)
- Odvozené (vypočítané hodnoty na základě jiných, např. rok narození)

1.3.4 Omezení integrity

Abychom mohli zachovat data univerzálně využitelným zdrojem, je nutné dodržovat několik pravidel. Ta jsou nazývána integritní omezení a jsou dělena dle předmětu, na který jsou aplikovány. (14, s. 28)

- Integritní omezení pro entity
 - Integrita doménová
 - Integrita entitní
 - Integrita Referenční
- Integritní omezení pro vztahy entit

Doménová integrita

Tato integrita vyžaduje, aby každá z hodnot v atributu odpovídala parametrům, které jsou definovány nad atributem, tedy datový typ (string, integer, date, time), povinnost zadání hodnoty (ISNULL), jedinečnost, rozsah (počet znaků, velikost čísla) nebo seznam přípustných hodnot, atd. (14, s. 29)

Entitní integrita

Integrita entity se naopak vztahuje na unikátní atribut v rámci relace v n-tici – tedy primární klíč. Aby byla splněna podmínka primárního klíče, musí být záznamy ve sloupci jednoznačné a minimální. Nemohou tedy existovat dvě stejné n-tice a navíc musí být primární klíče soubor atributů, u kterých nelze žádný z nich vypustit, aniž bychom porušili podmínku jednoznačnosti. Primárním klíčem tedy může být: (14, s. 29)

- Jednoduchý, může obsahovat jen jeden atribut
- Složený, může v sobě zahrnovat více než jeden atribut
- Umělý, je tvořen pomocí samostatně definovaného atributu (klasické ID)

Nesmíme také zapomenout na další vlastnost primárního klíče a tou je fakt, že u každé z n-tic musí být atribut vyplněn – tedy nesmí být NULL. Zároveň již zmíněné pravidlo, že musí být v rámci n-tic v kterémkoliv okamžiku identifikovatelný záznam. Pro každý primární klíč také platí, že může mít jeden či více alternativ – tedy kandidátních klíčů. (14, s. 29)

Referenční integrita

Referenční integrita nám omezuje položku na opačném konci vztahu, než je samotný primární klíč – nazýváme jej cizím klíčem. Ten musí plně odpovídat některému z primárních klíčů v tabulce na druhém konci referenční vazby. (14, s. 30)

1.3.5 Vztahy integritních omezení

Princip integritního omezení vztahů mezi entitami spočívá v kardinalitě záznamů v tabulkách. Ty mezi sebou spojují referenční vazby. Vazby se mezi sebou označují poměrem, který odpovídá míře n-tic v relaci. Odpovídají následujícím vztahům: (14, s. 31)

- 1:1
- 1:N
- N:1
- M:N

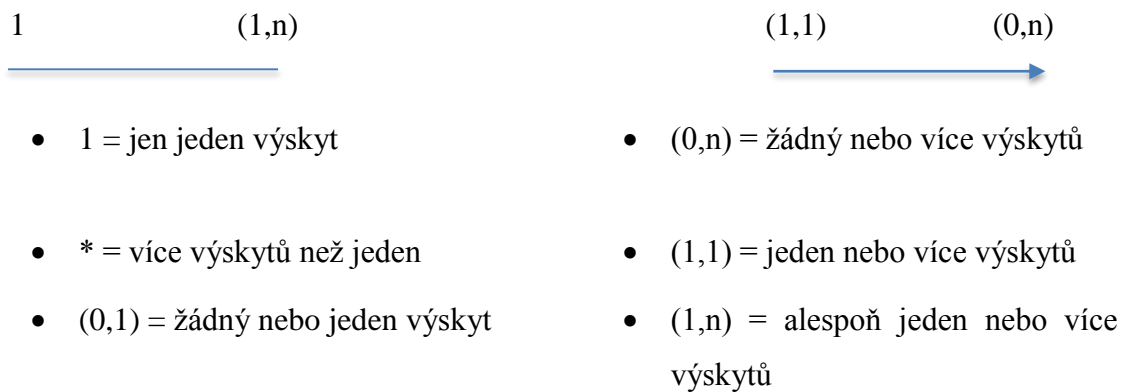
Princip fungování si můžeme uvést na jednoduchém příkladu. Uvažujme, že malé dítě trpí více nemocemi a současně jedno onemocnění může mít více dětí ve školce. Tento případ je ukázkou vztahu M:N. Jelikož nastává problém, že nemůžeme určit více nemocných dětí se stejnou nemocí, aniž bychom vytvářeli více stejných n-tic v tabulce (dítě1 – nemoc1, dítě1 – nemoc2), musíme provést tzv. dekompozici. Ta spočívá v tom, že přidáme mezi tento vztah další relaci – průnikovou entitu. Tato nová relace obsahuje svůj vlastní primární klíč ale mimo něj i cizí klíče obou relací. Po dokončení procesu dekompozice se tedy vazby místo M:N změní na 1:M a N:1. (14, s. 35)

Při značení vazeb se můžeme setkat s několika různými variantami značení. Rozdíly nejsou jen ve značení ale i ve vypovídající schopnosti samotných značek.



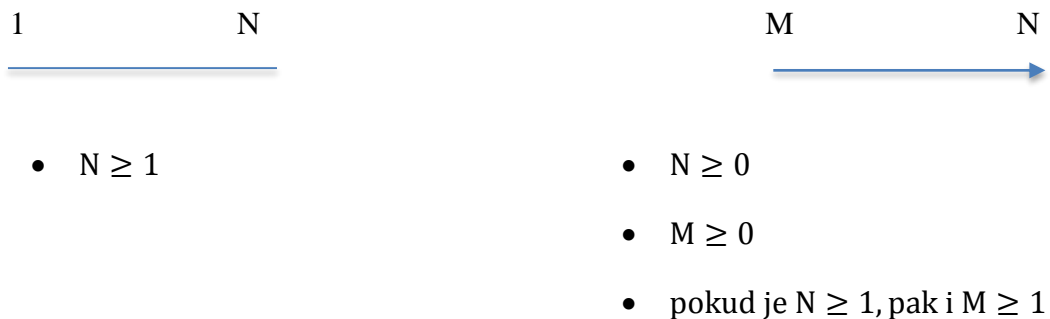
Obrázek 3: Ukázka Bachmanova způsobu značení kardinality vztahů

Zdroj: (15, s. 16)



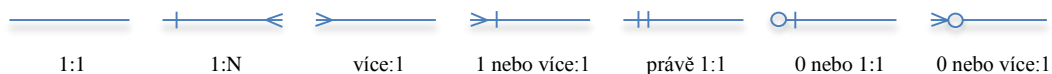
Obrázek 4: Ukázka Martinova způsobu značení kardinality vztahů

Zdroj: (15, s. 17)



Obrázek 5: Ukázka Chenova způsobu značení kardinality vztahů

Zdroj: (15, s. 39)



Obrázek 6: Ukázka inženýrského stylu způsobu značení vztahů

Zdroj: (15, s. 40)

1.3.6 Normální formy

Jedním z hlavních účelů provozu databáze je uchovávat optimálně využitelná data. Aby byla data takto optimálně využitelná, je třeba dodržovat normalizační pravidla – tedy normalizovat data do normalizačních forem (NF) neboli úrovní. (16)

- 0NF
 - V relaci existuje alespoň jeden atribut, který je neprázdný
- 1NF – „atomičnost“
 - Relace splňuje podmínku 0NF a neobsahuje složené nebo vícehodnotové atributy
- 2NF – „funkční závislost“
 - Relace splňuje podmínku 1NF a každý atribut v relaci je funkčně závislý na attributech primárního klíče
- 3NF – „tranzitivní závislost“
 - Relace splňuje podmínku 2NF a všechny neklíčové atributy na sebe nemají navzájem funkčně závislý vztah
- BCNF – „Boyce – Coddova forma“
 - Relace splňuje podmínku 3NF a současně mezi kandidátními klíči neexistuje funkční závislost za podmínek, že relace má alespoň dva cizí klíče, které jsou složené a z části se překrývají
- 4NF – „multizávislost“
 - Relace splňuje podmínku BCNF a funkčními závislostmi z CK jsou zároveň i vícehodnotové závislosti
- 5NF – „cykličnost“
 - Relace splňuje podmínku 4NF a pokud se vyskytuje více cyklicky spojených relací, musí být všechny součástí stejného vektoru hodnot

Obecným faktem je skutečnost, že se spíše vyžaduje optimalizace pouze do třetí normální formy (3NF). Zbylé se považují spíše za nadstandardní opatření. Pokud se však nebudou data optimalizovat alespoň do třetí normální formy, nemá smysl data shromažďovat.

Pro pozdější využití se stávají nepoužitelnými a nedá se s nimi rozumně pracovat a využívat k extrakci souvislostí, které v sobě mohou skrývat.

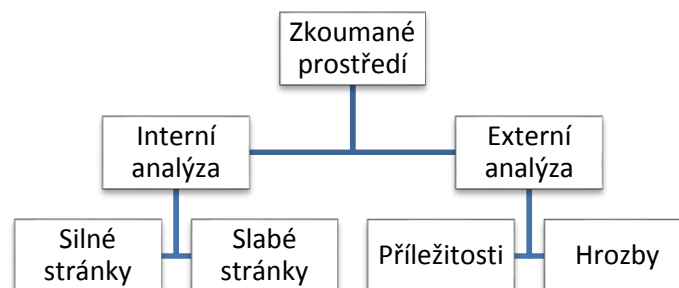
1.4 SWOT analýza

Základem strategické analýzy SWOT je zkoumání několika faktorů, které nám pomáhají například při rozhodování mezi různými formami strategií. Těmito sledovanými faktory jsou: (20)

- silné stránky (Strengths)
- slabé stránky (Weaknesses)
- příležitosti (Opportunities)
- hrozby (Threats)

Tyto čtyři faktory se spadají pod interní a externí analýzu. Nyní něco krátce k interní analýze. Z jednotlivých názvů si můžeme sami odvodit, že v případě silných stránek se jedná o všechny výhody analyzovaného subjektu před konkurencí. To stejné platí i pro slabé stránky. Avšak ty shrnují na rozdíl od silných stránek všechny nevýhody, které subjekt má před konkurencí. Ve SWOT analýze je velmi důležité zachovat objektivitu.

K externí analýze se řadí analýza příležitostí a hrozeb. Zde se zaměřujeme na vnější vlivy prostředí. Můžeme například zmínit různé příležitosti (boom na trhu s určitými produkty), nové technologie nebo nenaplněná poptávka spotřebitelů po určitém zboží. K tomu se ale řadí i hrozby například ze strany substitutů, konkurenční výroba s lepšími technologiemi nebo nižšími náklady na výrobu. Řadí se sem i různé zásahy státu, mezi které můžeme jmenovat různá regulační opatření nebo daňovou zátíženost. (21)



Obrázek 7: Matice SWOT analýzy

Upraveno dle (21)

1.5 SQL

SQL, tedy Structured Query Language je v dnešní době nevíce rozšířeným jazykem relačních databází. Z historického pohledu jazyk vznikl v roce 1986. Tehdy byl přijat jako první standard SQL86. Později, roku 1992 byl přijat standard SQL92 a ten se používá dodnes. (16)

Jedná se o deklarativní programovací jazyk – nelze využít jen jako samostatně stojící kód. Naopak, SQL kód se obvykle používá a vkládá do jiného z procedurálních programovacích jazyků (mimo použití při připojení přímo k SQL serveru). Tento skriptovací jazyk umožňuje nejen správu samotných dat, ale také správu databáze, ve které jsou data uchována. (16)

Můžeme tak jazykem SQL například říct databázi, se kterými daty hodláme pracovat, databázový software pak k těmto datům zajistí přístup. SQL jazyk disponuje širokou škálou příkazů, ty se větví dle předmětu užívání či uživatele na následující: (17, str. 88)

- DML – Data Manipulation Language (příkazy pro manipulaci s daty)
- DDL – Data Definition Language (příkazy pro tvorbu schémat databáze)
- VDL – View Definition Language (tvorba pohledů)
- SDL – Storage Definition Language (způsob ukládání tabulek)

Například díky DML jsme schopni z databáze načíst řádky, měnit je či mazat. Pomocí příkazů DDL zase celé tabulky vytvářet, editovat nebo mazat.

MySQL

Databázový systém MySQL byl vytvořena švédskou firmou MySQL AB. Tento databázový systém je považován za průkopníka dvojího licencování a to proto, že je jednak k dispozici zdarma – s licencí GPL (General Public License - všeobecná veřejná licence), tak i s komerční – placenou verzí. Tento systém je multiplatformní a jak lze odhadnout již z názvu systému, komunikace s databází probíhá za pomoci jazyka SQL.

Právě díky jednoduché možnosti implementace, výkonu a faktu, že se jedná o licenci zdarma vlastní MySQL vysoký podíl v používaných databázových systémech dnešní doby. Velmi častou a oblíbenou kombinací MySQL je Linux, PHP a Apache - můžeme se s nimi například často setkat jako základní vybavení každého webového serveru.

Již od počátku bylo MySQL stavěno především za účelem vysokého výkonu a rychlosti, bohužel i za cenu některých zjednodušení. Kvůli tomu má například jednoduché způsoby

zálohování. Do nedávna dokonce nebyly podporovány trigger, pohledy nebo uložené procedury. Tato zdokonalení jsou doplňována teprve v posledních letech, kdy začaly běžným uživatelům – programátorům webových stránek – chybět (18, 19)

1.6 Použité technologie

1.6.1 HTML

Hypertext Markup Language – HTML je kód sloužící ke zpracování dat a to předem daným způsobem. Díky němu můžeme stručně a jasně vymezit a vyjádřit, co potřebujeme zobrazit uživateli v prohlížeči. Volně řečeno se jedná o jazyk, který slouží k popisu, co má webová stránka obsahovat. V souladu s technologickým vývojem a různými požadavky a potřebami producentů softwaru a jeho uživatelů se vyvíjí i jazyk HTML již řadu let. Protože existuje mnoho firem, které se zabývají vývojem vlastních webových prohlížečů, je třeba upřesnit pravidla chování a zobrazování tak, aby za použití různých prohlížečů zobrazení stránek vypadalo stejně. Aby tato pravidla fungovala, vznikla W3C (WWW Consortium). Ta schvaluje konečnou podobu pravidel pro zobrazování webových stránek. (23, str. 15-16)

Na nepoužívanější prohlížeče se můžeme podívat v následujícím krátkém přehledu. Data jsou platná za únor 2015: (25)



Obrázek 8: Statistika jednotlivých prohlížečů za únor 2015

Upraveno dle (25)

Je obecným faktem, že internet patří mezi nejmocnější média. Díky tomu dovede ovlivnit velké masy lidí a samotná role internetu je pro každého z nás čím dál více důležitější.

I proto je důležité zdokonalovat stávající koncepty a vyvíjet nová specifika. V této oblasti má právě W3C primární úlohu. Odborníci z W3C na těchto zdokonalení a nových konceptech pracují. Dnes je dokončena nejnovější specifikace – technologie HTML 5.0, která v sobě zahrnuje všechny funkce, které po ni byly požadovány. (24)

Dnes již W3C pracuje na nových prvcích, které by měly být zahrnuty do verze HTML 5.1. Jedná se zlepšení zpracování videa, vylepšené formuláře (počítá se s funkcí autocomplete či kontrolou pravopisu), zlepšené zpracování obrázků a mnoho dalších funkcí. W3C také zveřejnilo specifikace Canvas 2D Level 2, které by mělo přinést podporu zobrazení vysokého rozlišení na displejích, které toto vysoké rozlišení umožní. Původní odhady dokončení byly naplánovány na rok 2016. W3C nadále řeší problém se standardizovaným videokodekem pro HTML5. Uvažovalo se o H.264 nebo VP8 od společnosti Google, nakonec však z těchto možností sešlo. Analytici říkají, že dokončení standardu HTML5 je sice skvělá zpráva, ještě je však před W3C dlouhá cesta, než se dořeší některé problémy (např. při zobrazování na mobilních zařízeních), které sebou inovace HTML5 přinesla. (24)

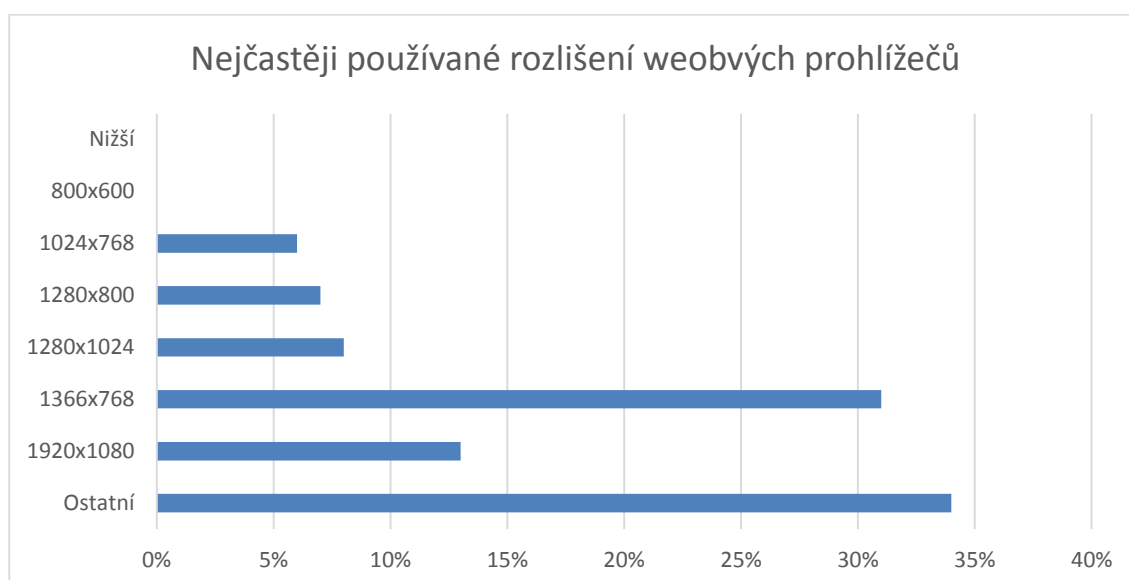
1.6.2 CSS

CSS (Cascade Style Sheets) neboli kaskádové styly nabízejí programátorům webových stránek široké možnosti, jak přizpůsobit vzhled webových stránek k jejich představám. Díky nim lze výkonně, efektivně a flexibilně formátovat obsah HTML stránek. V předchozí kapitole jsme popisovali jazyk HTML, který je primárně určen pro popis struktury dokumentu. Za použití správné kombinace CSS a HTML můžeme docílit oddělené prezentační formy od struktury. Nejen, že se tím dosáhne zjednodušení tvorby a správy webového dokumentu,lepší se tím i jeho přístupnost a přehlednost. (19, str. 269–270).

Pomocí CSS můžeme definovat různé fonty, barvy, pozadí, styly písma, tabulek a hlavně rozměrové prvky stránek. Nároky na zobrazení jsou hlavně pro dnešní dobu vysoké. S tím souvisí i plochy, na které se webové stránky vykreslují. Ty jsou různé, záleží na možnostech uživatelského displeje a jeho rozlišení.

Za zmínku tedy určitě stojí uvést i přehled nepoužívanějších rozlišení zobrazení, která se v dnešní době používají. Může nám to prozradit, jakým směrem se při vývoji webového informačního systému ubírat a na co se zaměřit. Jakýkoliv smysl totiž postrádá vyvíjet webovou aplikaci pro rozlišení, které je pro dnešní dobu zastaralé. Nutno však podotknout, že správná

webová aplikace či obyčejný web by měl umět pracovat se všemi možnými rozlišeními. Tento způsob programování webových stránek či aplikací se nazývá „responsive design“ neboli responzivní design. Díky němu jsme schopni zobrazit stránky jak na velkém rozlišení (např. FULL HD - 1920x1200px), tak i na malém rozlišení, kterým disponují například mobilní telefony. Aby byla usnadněna aplikace responzivního designu, existuje dnes řada tzv. pluginů, která nám aplikaci umožní. Jedná se např. o plugin Bootstrap či Foundation. Když tyto pluginy přidáme na svůj web a používáme je při jeho tvorbě, přizpůsobení všemožným uživatelským rozlišením tak máme několikanásobně usnadněné. Tyto pluginy se nevztahují jen na přizpůsobení se, obsahují v sobě i základní nastýlované prvky, které můžeme pouhým přidáním class k prvku v HTML nastýlovat. Data jsou platná k Lednu 2014: (25)



Obrázek 9: Nejčastěji používaná uživatelská rozlišení – Leden 2014

Zdroj: (24)

Nutno zmínit, že mezi ostatní rozlišení patří především rozlišení 1440x900 které tvoří 7% a větší rozlišení, než je 2560x1440, které k mému překvapení tvoří 10%.

1.6.3 JavaScript

JavaScriptové skripty se dají definovat jako krátké programy, které jsou umístěné na webové stránce a doplňují možnosti jazyka. Pomáhají řešit některé problémy, které jsou dodnes

v HTML zcela otevřené. Tento jazyk byl vyvinutý společností Netscape. Tyto skripty – mnohdy skoro programy, které jsou napsané v JavaScriptu jsou naprosto nezávislé na platformě počítače, kde se zobrazují. Vše záleží na prohlížeči. Pokud tedy prohlížeč dané skripty podporuje, je možné skripty spustit na všech dostupných operačních systémech. Rozdíl oproti PHP skriptům je ten, že JavaScript se provádí na straně uživatelského prohlížeče, nikoliv na serveru. Díky tomu nemusí server zbytečně zatěžovat a ten se může soustředit na své vlastní výpočty dat.

Nevýhodou zůstává fakt, že je možné JavaScript manuálně vypnout a tím se dá narušit samotná funkčnost stránek. Ale tento krok je čistě na uživateli a záleží čistě na něm, jestli chce funkční webovou prezentaci či nikoliv. Při programování je však dobré na tuto skutečnost pamatovat a pokusit se stránky naprogramovat tak, aby byly funkční i při vypnuté podpoře JavaScriptu. (27, str. 15-16)

Jak jsem již zmiňoval, programovací kód se provádí na straně uživatele – v prohlížeči. Zdrojový text se tedy píše rovnou do HTML nebo do samostatného souboru, na který se v HTML odkazuje. Po načtení stránky se řádek po řádku vykonávají příkazy napsaného skriptu. Dle (27, str. 17) patří mezi nejzákladnější vlastnosti JavaScriptu:

- Možnost reagovat na různé události (pohyby myši či stisku klávesy)
- Možnost provádět akce s uživatelem zadanými daty
- Možnost vytvářet interaktivní formuláře
- Možnost měnit obsah stránky v různých závislostech (čas, datum, období)
- Možnost uchovávání uživatelských dat (cookies)
- Možnost vytvářet dynamické stránky

K dalším plusům patří mimo jiné používání tzv. pluginů. Díky nim si můžeme ušetřit několik desítek hodin času při vytváření webových stránek. Mezi neznámější pluginy dnes patří bezesporu JQuery knihovna. Ta nám umožňuje používat JavaScript jednodušším způsobem a práce s knihovnou je velmi efektivní. Mezi další patří např. Less, Bootstrap, Foundation a tisíce jiných různých knihoven.

1.6.4 PHP

PHP je programovací jazyk, který úzce souvisí s vytvářením informačních systémů. „*PHP je hypertextový procesor, který na serveru interpretuje stránky HTML s vlastními příkazy před jejich odesláním ke klientovi (obvykle je jím webový prohlížeč)*“ (22, str. 15).

Ve výsledku to znamená, že tento programovací jazyk umí vkládat vlastní skripty do HTML stránek. Na tom však není nic zvláštního, stejným způsobem se používá například JavaScript. Na rozdíl od něj je ale jazyk PHP interpretován na serveru, nikoliv na straně uživatelského prohlížeče, jak je tomu u JavaScriptu. Při provádění kódu serverové stránky přináší především tyto výhody: (22, str. 16):

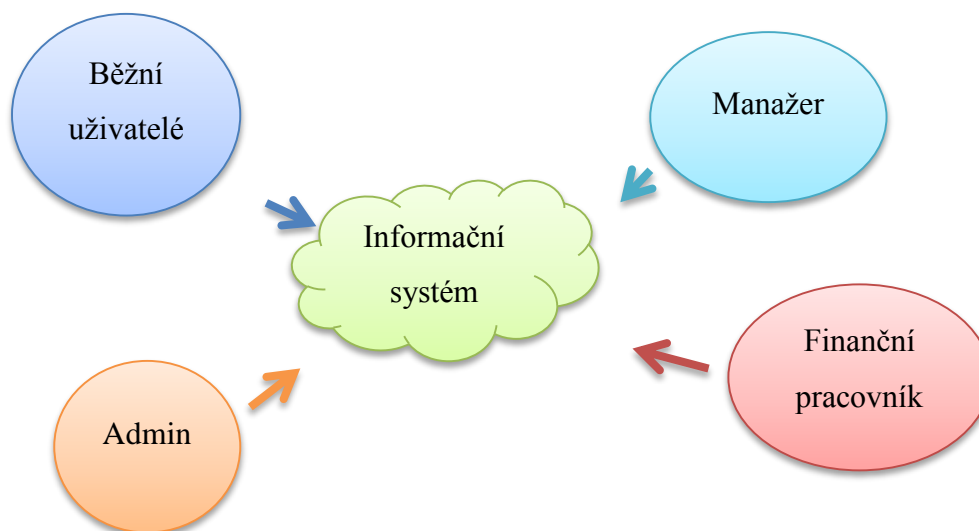
- ✓ Větší bezpečnost – při potřebě interakci s dalšími aplikacemi na serveru není třeba přednášet data po internetu, data mohou být vyměněna mezi aplikacemi přímo na samotném serveru
- ✓ Snížené nároky na hardware nebo software klienta – samotným výstupem PHP skriptu jsou obvykle čisté HTML. To znamená, že klient nemá žádné potíže, o vše se postará běžný prohlížeč. Klient tedy není zatěžován prováděním skriptů či rozsáhlých výpočtů.
- ✓ Nižší nároky na konektivitu – protože se skripty, které jsme vložili do kódu, interpretují na serveru a ve výstupu se již neobjeví, je objem přenesených dat nižší. Skripty určené pro klienta naopak ano.
- ✓ Ochrana zdrojových textů programů – jelikož interpretace kódu probíhá na straně serveru, klient se tak nemůže dostat ke zdrojovým textům. Naopak při použití klientských webových technologií jsou zdrojové texty přímo zaslány uživateli.

2 ANALÝZA SOUČASNÉ SITUACE

Již v úvodu jsem zmínil, že drtivá většina hudebních těles nevyužívá žádný efektivní elektronický informační systém pro řízení a management samotného tělesa. Když už se najde nějaká světlá výjimka, používají kombinaci několika nástrojů ke komunikaci, správě financí, správě členů, kontaktů a kalendářů.

Jako aktivní člen několika hudebních orchestrů, ve kterých se podílím nejen jako hrající člen orchestru, ale i jako řídicí člen, získal jsem zkušenost, díky které dovedu určit aspekty důležité pro řízení samotného hudebního tělesa. Tyto aspekty nyní využiji k hodnocení ostatních systémů, které se nyní využívají v některých orchestrech.

Hned na úvod je potřeba vymezit, kteří uživatelé mají systém používat. Vzhledem k tomu, že se jedná o informační systém pro hudební těleso, je jasné, že převážná většina uživatelů budou samotní hudebníci. Nicméně aby měl informační systém smysl a pomohl při vedení tělesa, měl by sloužit i ředitelům, manažerům či kapelníkům samotného hudebního tělesa. V momentě, kdy jsou již v systému obsaženy všechny informace o hráčích a vystoupeních, je vhodné těchto informací využít a připojit i výpočet a evidenci mezd za samotná vystoupení.



Obrázek 10: Uživatelské schéma

zdroj: vlastní zpracování

Prvním a stěžejním aspektem, od kterého se odvíjí celý evidenční systém, je samotná evidence vystoupení kapely. Jedná se o jakousi agendu, dalo by se říct i kalendář, ve kterém se evidují všechna vystoupení. Měl by umožňovat i náhled na detail vystoupení, ve kterém by se měly evidovat potřebné detaily o samotném vystoupení, jako je například čas, datum, místo, oblečení, poznámka k vystoupení a samotné detaily pro management – interní odhad nákladu, cena ve smlouvě, skutečný náklad na vystoupení, zisk, kontaktní osoba a kontakty na ni samotnou.

Tabulka 1: Příklad agendy

Zdroj: vlastní zpracování

Datum	Místo	Typ	Od	Do	oblečení	poznámka
12. 12. 2006	Olomouc	vánoční koncert	19:00	21:00	sako	zvuková zkouška v 18h
14. 12. 2006	Brno	vánoční koncert	17:00	19:00	sako	sraz na Benzině v 16h
12. 1. 2007	Ostrava	ples	20:00	03:00	sako	generálka v 19h

Dalším aspektem je umožnit každému uživateli, aby si za sebe zvolil alternujícího hráče. V širších orchestrech při větším vytížení se nemůže kmenový hráč zúčastnit všech vystoupení. Najde si proto za sebe náhradu. Aby nemusel složitě vše domlouvat s vedením, v systému za sebe pouze napíše alternujícího hráče.

Tabulka 2: Přehledná soupiska hráčů daného vystoupení

Zdroj: vlastní zpracování

Nástroj	Hráč
Pozoun 1b	Miroslav Voráček
Tromba 2b	Jiří Plachý
Housle	Dalibor Ostřížek

Diskuze ke každému vystoupení je dalším z aspektů. Při menším obsazení orchestru se vede diskuze například o dopravě na dané vystoupení nebo o jiných libovolných záležitostech, týkajících se vystoupení. Proto by se mělo jednat o strukturovanou diskuzi, kde by se dalo vést více témat a navzájem na sebe reagovat.

V případě, že za sebe člen hledá náhradu, je možné využít knihu kontaktů, kde může sám vložit nebo pouze hledat potenciálního alternujícího hudebníka.

Také by se velmi hodilo vyřešit stránku financí. Za každé vystoupení získá orchestr honorář, který se dělí mezi členy orchestru. Proto je potřeba zajistit evidenci právě těchto financí.

Tabulka 3: Příklad evidence financí

Zdroj: vlastní zpracování

Jméno	nástroj	km	cestovné	hodin	sazba	celkem
Vlastimil Krátký	1. Housle	50	500	8	100	1300

Tyto zmíněné faktory pokládám za kritéria, pomocí kterých budu hodnotit stávající informační systémy, které se v současné době využívá několik různých hudebních seskupení.

Mimo tato kritéria ale budu hodnotit i složitost ovládání, design a samotné provedení a případnou integraci do již zavedených webových stránek.

Nyní vyberu z již existujících informačních systémů, které jsou v praxi nasazeny a využívány a provedu na nich analýzu dle zmíněných kritérií.

2.1 Informační systém složený z několika komponent

Tento informační systém – dá-li se jej vůbec informační systém nazývat, využívala dříve kapela Hornobojani. Jednalo se o zobrazení HTML stránky, která se skládala z několika iframů – diskuze a kalendáře.

Diskuze zde byla jen jedna a využívalo se v ní komponenty, kterou poskytuje služba blueboard.cz. Tato komponenta byla velice primitivní, nedalo se zakládat více vláken jednoho příspěvku a příspěvek byl tedy uzavřený vztahem rodič – potomek. Neexistovala evidence uživatelů, každý uživatel tak mohl vystupovat pokaždé pod jinou přezdívkou a v případě zneužití systému by bylo jednak velmi složité rozpoznat, že systém někdo zneužívá, ale také velmi složité identifikovat podvodníka, neboť se může lehce vydávat za některého z uživatelů tohoto systému.

Evidence jednotlivých vystoupení se zde prováděla formou událostí v kalendáři. Z tohoto pohledu se už jednalo o pokročilejší využívání, avšak nebylo žádným způsobem propojeno s diskuzí. Datová struktura kalendáře byla také nevyhovující, dalo se sledovat jen datum, místo a poznámka.

Ostatní sledované prvky, jako jsou správa financí, kontakty, správa jednotlivých uživatelských účtů v tomto systému zcela chybí.

K přihlášení do systému sloužilo pouhé zadání hesla, které se zadávalo přímo na webové prezentaci kapely. Z hlediska autentizace se tak nedalo rozlišit, který uživatel zrovna informační systém používá.



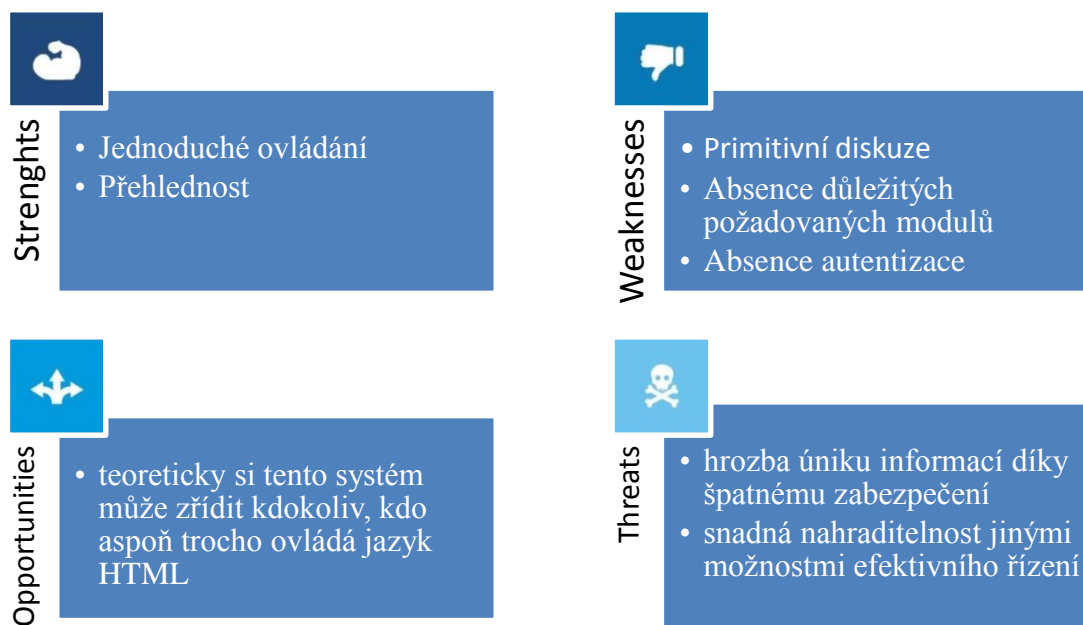
Obrázek 11: Komponenta diskuze od společnosti Blueboard.cz

Zdroj: archiv webového portálu www.hornobojani.com

Z hlediska vzhledu se jedná o velice zastaralý web, nekompaktní a bez jakýchkoliv náznaků responzivního designu. Jedinou vlastností, která se systému nedá upřít je jeho snadné ovládání. To se ale vzhledem k jeho jednoduché struktuře dá očekávat.

Po provedení SWOT analýzy je patrné, že systém není nijak možné v dnešní době rozumně používat. Jeho silné stránky nedovedou ani zdaleka vyvážit jeho nedostatky. O příležitostech samotných se dá pouze polemizovat a hrozby tohoto informačního systému jsou naprosto kritické – je zde obrovské riziko prolomení, zneužití či ztráty dat, které systém spravuje.

2.1.1 SWOT analýza



Obrázek 12: SWOT analýza systému složeného z komponent
zdroj: vlastní

2.1.2 Zhodnocení systému

Tento systém tedy nesplňuje žádné předpoklady pro možnost řízení hudebního tělesa. Je velmi primitivní a přínos pro řízení je opravdu minimální. Jedinou příležitostí může být snad jen to, že si jej může zřídit prakticky kdokoli, kdo zná základy HTML.

2.2 Informační systém kapely Boršičanka

Informační systém pro tuto kapelu byl opět sestaven na zakázku. Tentokrát se již jedná o pokročilejší evidenci vystoupení, kde se sleduje více dat a systém je navíc integrován s webovou prezentací samotného hudebního tělesa.

Celé webové stránky byly naprogramovány v jazyku PHP. Díky použití této technologie tak nebyl problém začlenit interní systém přímo do stránek a provést vzájemnou integraci. Samotný interní systém pak pro evidenci využívá databázi MySQL.

Oproti předchozímu analyzovanému systému zde chybí diskuze. To z prostého důvodu, uživatel tohoto systému je pouze jeden jediný a to je samotný manažer, který si vede evidenci vystoupení.

Jak již bylo zmíněno, evidence vystoupení je zde integrována do webové prezentace a uživatelem zvolené vystoupení se zobrazuje všem návštěvníkům stránek, tedy i hráčům. Ti si tak mohou sami udělat přehled o tom, kdy se mají na dané vystoupení nachystat.

SO 16.8.2014		20,15 - 21 a 22 - 01		Valašsko		koncert + hra k tanci	
Přidat akci	Den Datum	Čas	Místo				
	SO 16.8.2014	13,30 - 15,00	Ostravsko				vypravěčem
	SO 16.8.2014	20,15 - 21 a 22 - 01	Valašsko				hra k tanci
	NE 17.8.2014	15,00 - 18,00	Uherskohl				lidový vypravěč
	SO 23.8.2014	13,00 - 14,15	Přerovsko				přezd do SK
	SO 23.8.2014	19,30 - 21,15	Slovensko				koncert
	SO 8.9.2014	od 14,30 - 18,00	Uherskobro				průvod+koncert
	NE 14.9.2014	09,00 - 10,00	UH				koncert
	SO 4.10.2014	11,15 - 12,30	Zlín				koncert
	PA 17.10.2014	21,30 - 24,00 (01)	ŠVÝCARSKO				soukromá akce
	SO 18.10.2014	17,30 - 19,30	NĚMECKO				koncert
	CT 20.11.2014	18,00 - 21,00	Hodonín				seniři DH
	NE 7.12.2014	14,30 - 16,15	Výkovsko				adventní koncert - DK
	SO 13.12.2014	14,30 - 16,15	Ratíškovice				Adventní koncert - sokol
	NE 14.12.2014	10,00 - 11,00	Uherské Hradiště				adventní koncert - náměstí

Obrázek 13: Interní webový systém kapely Boršičanka

zdroj: www.borsicanka-ak.cz

V evidenci lze sledovat více parametrů, než tomu bylo v předchozím systému. Ke každé akci lze sledovat mimo důležité parametry jako den, datum, čas, i místo, psč, typ události, informace pro veřejnost a přepínač nový záznam a viditelnost pro veřejnost. Ty slouží právě pro zobrazení ve veřejné části webové prezentace, která je dostupná pro každého návštěvníka webových stránek. Lze tedy určit, které události se návštěvníkům zobrazí a popř. zvýraznit značkou nově přidané.

Upravení akce zpět na výpis akcí

Nový záznam

Viditelnost pro veřejnost

Den

Datum (rrrr-mm-dd)

Místo

PSC

Typ

Čas

Info pro veřejnost

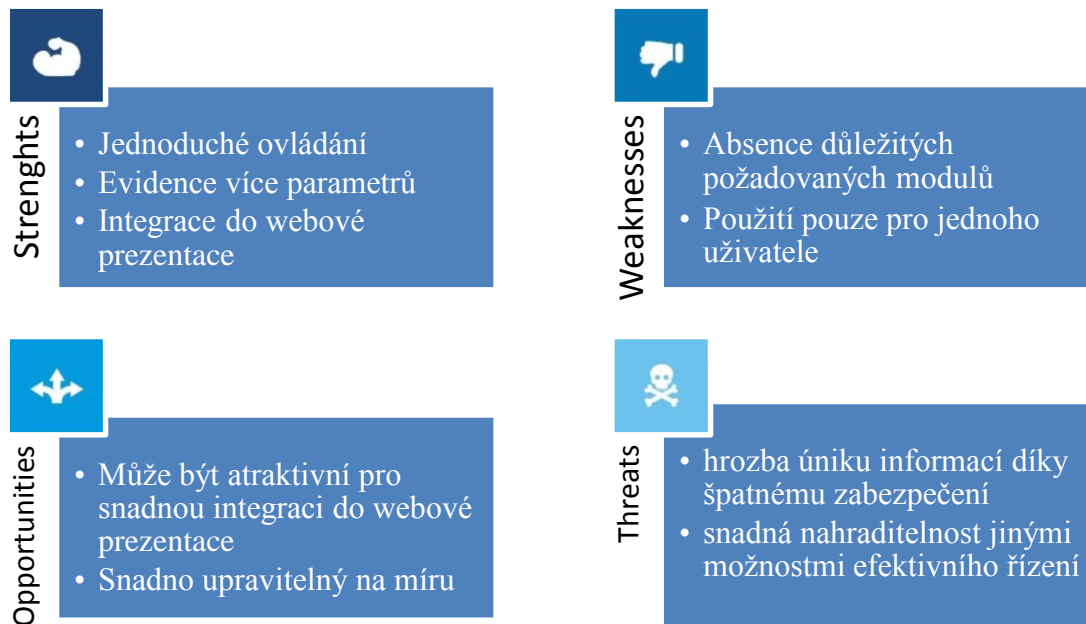
🗑

Obrázek 14: Detail události

zdroj: www.borsicanka-ak.cz

Bohužel, výše zmíněné vlastnosti byly jediné, které se dají přičíst ke kladům tohoto systému.

2.2.1 SWOT analýza



Obrázek 15: SWOT analýza interního systému Boršičanky

zdroj: vlastní

2.2.2 Zhodnocení systému

Dle stanovených kritérií zde chybí všechny ostatní důležité prvky. Systém slouží pouze manažerovi, nelze zde vést žádnou diskuzi, absence kontaktů, absence správy financí, hráčů na samotném vystoupení, absence responzivního designu. Nic z uvedeného zde nenalezneme. Celkový design systému je také velmi prostý a nezajímavý. Proto hodnotím systém také jako nedostačující.

2.3 Interní web kapely Lácaranka

Při analyzování problematiky managementu systémů hudebních těles jsem dospěl k závěru, že tento v řadě třetí analyzovaný informační systém je v současné době nejsložitějším systémem. Byl opět navržen na míru přímo pro kapelu a konečně slouží nejen jako obyčejná evidence, ale i jako komunikační kanál mezi všemi uživateli systému.

Interní web je stejně jako předchozí navržen a naprogramován v PHP s podporou MySQL. V současné době již několikátým rokem běží na webhostingu kapely, která jej denně využívá jako distribuční kanál informací.



Obrázek 16: Logo interního webu kapely

zdroj: interní web kapely Lácaranka

Tento systém jako první z analyzovaných plně využívá autentizaci uživatelů. Existuje tedy číselník uživatelů, kteří se mohou do systému pod svým uživatelským jménem a heslem přihlásit. Aby se zvýšila bezpečnost, stránky jsou umístěny na speciálním místě, o kterém vědí pouze uživatelé.

The image shows a simple login form within a rectangular border. At the top left, the text "Přihlásit" is displayed. Below it, the label "Login" is positioned to the left of a horizontal input field. Further down, the label "Heslo" is positioned to the left of another horizontal input field. At the bottom center of the form, there is a grey button with the text "login" in white.

Obrázek 17: První evidenční systém s autentizací

zdroj: interní web kapely Lácaranka

Jako další klad bych zde zmínil i použití zabezpečení dat použitím rolí uživatele. Je tedy možné přiřadit různá práva různým uživatelům tak, aby mohli různě do systému zasahovat. Tento číselník uživatelů s přiřazenými rolemi lze samozřejmě pohodlně spravovat přímo v systému. Každý uživatel si může spravovat svůj kontakt. V případě nutnosti tak není problém dohledat jakýkoliv kontakt na člena orchestru.

Zároveň se u každého uživatele sledují statistiky příspěvků a jeho poslední aktivita. Ta je pro manažera velmi důležitá. V orchestru totiž mohou hrát i hráči, kteří nejsou počítačově zdatní nebo momentálně nemají přístup k systému. Před každým vystoupením si tak manažer může ověřit, kdy se dotyčný hráč naposledy přihlásil. Pokud manažer rozpozná, že uživatel

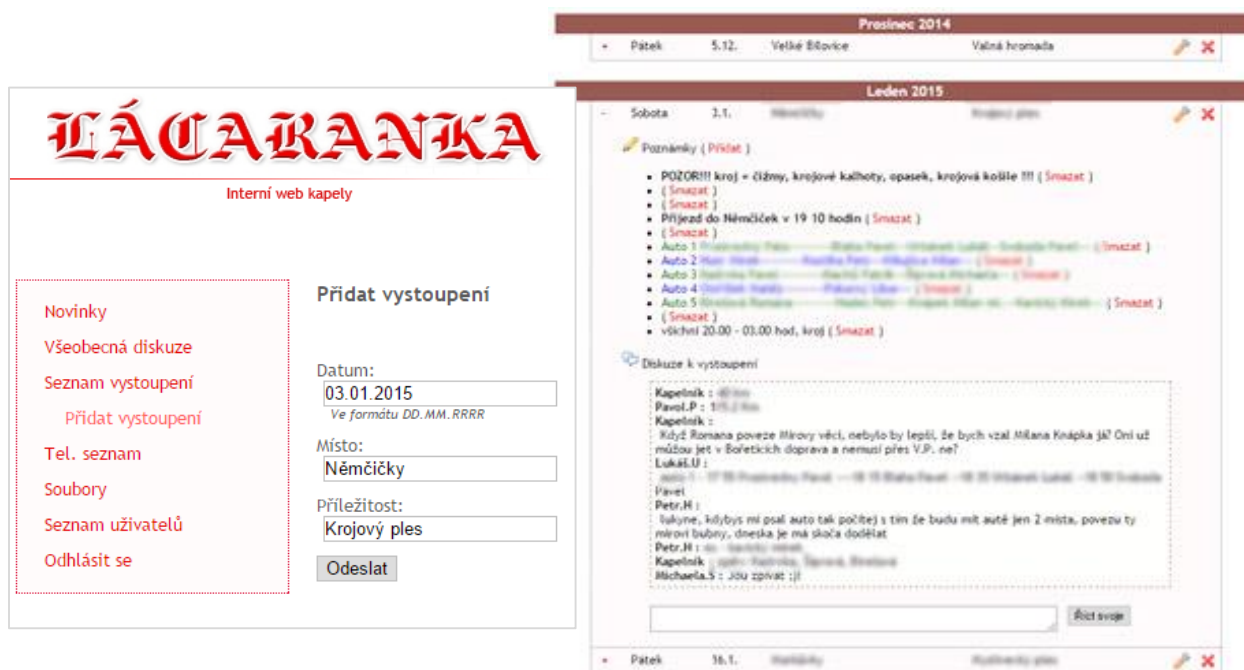
nové informace ohledně vystoupení nečetl, může jej kontaktovat přímo a informace mu sdělit např. telefonicky.

Jméno	Login	Kontakt	Poslední aktivita	Příspěvky na web
David Kadřavý	Nick	nick.pavlik, kadrcav@centrum.cz telefon: 602000000 ID: kadrcav Upravit	02.02.2015 21:55:10	0 akcí 0 poznámek k akcím 8 příspěvků do veřejné diskuze 0 příspěvků do diskuzí pod akcemi uložených souborů
David Kadřavý	Kapelník	nick.pavlik, kadrcav@centrum.cz, tel: 602000000	02.02.2015 18:35:43	10 akcí 12 poznámek k akcím 175 příspěvků do veřejné diskuze 18 příspěvků do diskuzí pod akcemi uložených souborů

Obrázek 18: Správa uživatelů interního webu

zdroj: vlastní

V systému samozřejmě neschází evidence vystoupení. Ta je zde velmi přehledná, uspořádaná do jednotlivých měsíců. Součástí přehledu je i možnost zobrazení podrobností k události. Ty jsou zde vedeny sice nestrukturovanou formou jakési „poznámky“, avšak uživatelům tohle nevadí. Je tedy možné informovat uživatele o různých podrobnostech vystoupení. Nevýhodou tohoto systému tedy zůstává nestrukturovaná data o vystoupení.



Obrázek 19: Evidence vystoupení s podrobnostmi a diskuzí zdroj: vlastní

Poprvé je zde i možnost samostatné diskuze pro každé vystoupení. Při zobrazení detailu každé akce si mohou uživatelé posílat vzkazy, které se nějakým způsobem týkají zvolené události. To považuji za významné plus. Nevýhodou ovšem zůstává fakt, že se jedná o velmi prostou diskuzi, nestrukturovanou bez možnosti vedení více vláken. Neexistuje zde tedy žádná vazba mezi příspěvky. Pokud by systém využívalo více uživatelů, je velmi pravděpodobné, že by se v diskuzi několik příspěvků mohlo ztratit mezi řádky.

Dalším sledovaným aspektem je kontaktní list. Ten v tomto systému také nechybí. Uživatelé si tak mohou sdílet kontakty na jiné hudebníky, které mohou případně potřebovat pro svou vlastní alternaci.

Seznam položek

Jméno	Nástroj	Kontakt	
Bílková, Andrea		753 489 427 (banda) 514611046	×
Cafourek, Pavel		754 489 428	×
Cíleček, Radek		757 489 470 (banda) 517546104	×
Kaňa, Tonda		756 489 471	×
Koniček, Antonín		757 477 470 (banda) 514611046	×
Kovaříková, Dagmar		754 477 489 (banda) 514611046	×
Kunický, Místostarosta-Kobyří		756 489 489	×
Loveček, Josef		751 489 512 josef.lovecek@banda.cz	×
Otáhal, Josef		448 854 489	×
Rajchman, Jaromír		443 719 489	×
Hanzlíček, Aleš	baryton	448 719 489	×
Otáhal, Milan	baryton	753 503 487	×
Grmolec, Jan	bicí	448 854 489	×
Grmolec, Lukáš	bicí	757 479 489	×
Hájek, Pavel	bicí	757 514 489	×

Obrázek 20: Evidence vystoupení s podrobnostmi a diskuzí

zdroj: vlastní

Dalším sledovaným aspektem je možnost evidence záskoků. Ta se zde dá bohužel provádět pouze příspěvkem v diskuzi. Tím trošku na své kráse použitelnosti systém ztrácí. Avšak nutno podotknout, že tento aspekt nebyl předmětem programování. Na tuto funkci totiž při vývoji nikdo neupozornil.

Po designové stránce hodnotím systém jako poměrně povedený, minimalistický a sladěný s logem a barvou webových stránek. Částečně splňuje i požadavky na responzivní design a rozhodně se řadí mezi přehledné systémy.

2.3.1 SWOT analýza



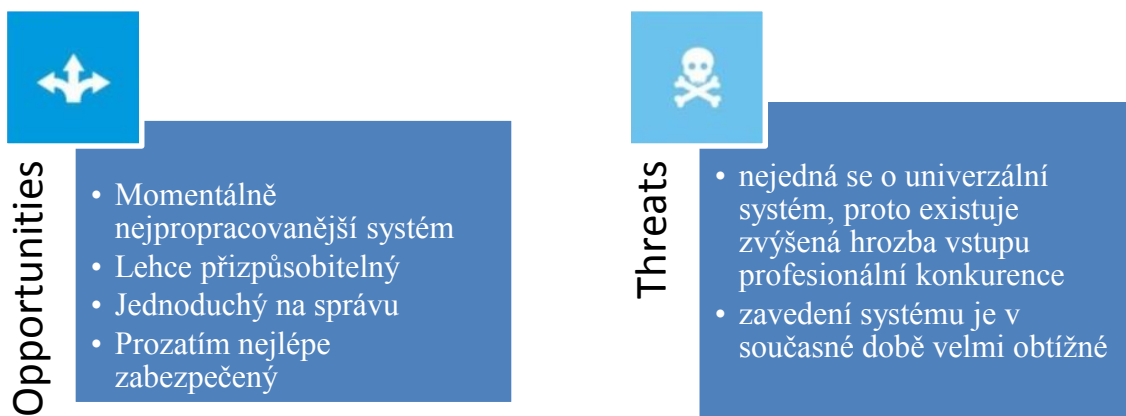
Strenghts

- Jednoduché ovládání
- Autentizace uživatelů
- Separátní diskuze
- Evidence kontaktů
- Sledování aktivity uživatelů



Weaknesses

- Jednovláknové diskuze
- Nedostatek parametrů vystoupení
- Absence seznamu hráčů na vystoupení



Obrázek 21: SWOT analýza interního systému Lácaranky

zdroj: vlastní

2.3.2 Zhodnocení systému

Jelikož jsem nenarazil na efektivnější nástroj řízení hudebního tělesa, považuji tento systém za současně nejpoužitelnější a nejlepší. Podle analýzy dle mnou stanovených aspektů alespoň z poloviny splňuje požadavky, které jsou na systém kladeny. Bohužel zde zůstávají důležité faktory, které implementovány nejsou a nejdou tak použít v obecně širší rovině.

2.4 Celkové zhodnocení současného stavu

Jak jsem již předpokládal v úvodu této práce, momentálně neexistuje žádný efektivní nástroj na řízení hudebního tělesa. Žádný z mnou analyzovaných systémů nesplňoval požadovaná kritéria, které jsem stanovil na základě zkušeností, které jsem nabyl při mém působení v několika různých hudebních útvarech.

Na základě této analýzy tedy mohu přistoupit k návržení univerzálního informačního systému, který by efektivně dokázal řídit chod libovolně velkého hudebního tělesa s různorodým zaměřením.

Při navrhování budu klást důraz na již několikrát zmiňované aspekty a budu se snažit, aby navržený a naprogramovaný systém splňoval všechny požadavky současné doby a trendu.

3 VLASTNÍ NÁVRHY ŘEŠENÍ

Prvním krokem, který před realizací nového informačního systému provedu, je výběr technologií pro projekt.

3.1 Výběr technologií

Prvním předpokladem informačního systému je jeho dostupnost. Jelikož ho mohou využívat desítky uživatelů, nemá smysl systém stavět formou tlustého klienta, kdy by se musely provádět instalace na klientské stanice. Proto bude informační systém naprogramován jako tenký klient a uživatelé se budou moct přihlásit odkudkoliv na světě pomocí webového prohlížeče.

Vzhledem k faktu, že do cílové skupiny spadají i malá hudební tělesa, která nemají dostatečně vysoké finanční zázemí, rozhodl jsem se využít technologií mySQL, PHP, HTML, CSS a Javascriptu. PHP a mySQL především proto, že drtivá většina všech hudebních těles má webovou prezentaci, která běží u poskytovatele na nějakém hostingovém prostředí. To dnes zpravidla zahrnuje i mySQL databáze.

Díky této výhodě nebude velký problém provést implementaci přímo k webové prezentaci hudebního tělesa. Jako příklad uvedu použití: webová prezentace hudebního tělesa je www.posadkova-hudba-olomouc.cz. Jelikož nechci, aby o informačním systému někdo cizí věděl, neuvádím nikde odkaz a každý z členů tělesa ví, že systém nalezne na stránkách www.posadkova-hudba-olomouc.cz/interni-system.

Mimo toto použití ale popřemýšlím nad situací, kdy bych službu nabízel jako outsourcing. Měl bych tedy svůj vlastní server, na kterém bych provozoval doménu a nabízel službu formou pronájmu více hudebním seskupením. Myslím, že tato forma může být pro tato hudební tělesa také zajímavá, neboť by se nejednalo o velkou finanční částku na pořízení systému. Jednalo by se pravděpodobně o nějaký nižší vstupní poplatek do systému a pak by se jen měsíčně platil poplatek za pronájem a provoz systému.

Z obchodního hlediska by bylo zajisté dobré provozovat obě varianty tak, aby si mohl zákazník libovolně vybrat podle toho, co mu bude vyhovovat více.

V následující kapitole se budu věnovat výběru podpůrných knihoven, které využiji při programování systému.

3.2 Výběr pluginů

Jelikož už mám určitou představu o tom, jak by měl informační systém vypadat a co vše by měl obsahovat, provedl jsem rešerši na internetu a vybral několik pluginů, které mi ušetří výrazným způsobem samotné programování nově navrhnutého informačního systému.

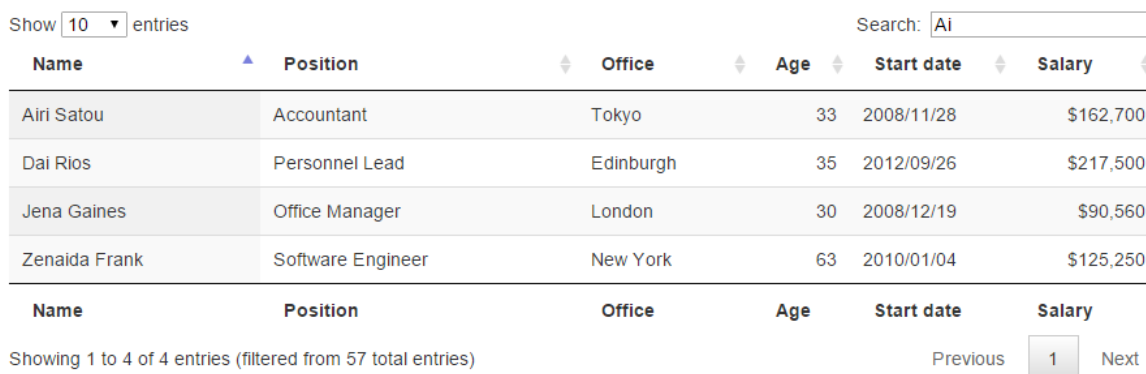
3.2.1 Javascriptové pluginy

Pro responzivní design jsem vybral plugin Foundation. Jedná se o javascriptovou knihovnu, která za pomoci kaskádových stylů a vhodné implementaci v HTML postará o to, aby aplikace vypadala dobře nejen na počítači s použitím různých rozlišení, ale i na mobilních zařízeních a tabletech.

K tomu samozřejmě nesmí chybět další javascriptová knihovna jQuery. Díky ní se nám výrazným způsobem zjednoduší práce s javascriptem samotným.

Jelikož bude informační systém pracovat se spoustou tabulkových záznamů, rozhodl jsem se po delším bádání na internetu použít jQuery plugin s názvem Datatables. Jedná se o javascript, který zpracuje obyčejnou html tabulku a předělá ji za na tabulku s nesčetným množstvím nastavení jako je sortování, stránkování, filtrování atd.

Posledním z pluginů je select2, který budu používat místo klasických HTML selectů.



The screenshot shows a DataTables interface. At the top left, it says 'Show 10 entries'. At the top right, there is a search box containing 'Ai'. Below this is a table with columns: Name, Position, Office, Age, Start date, and Salary. The table contains four rows of data. At the bottom, it says 'Showing 1 to 4 of 4 entries (filtered from 57 total entries)' and has 'Previous', '1', and 'Next' buttons.

Name	Position	Office	Age	Start date	Salary
Airi Satou	Accountant	Tokyo	33	2008/11/28	\$162,700
Dai Rios	Personnel Lead	Edinburgh	35	2012/09/26	\$217,500
Jena Gaines	Office Manager	London	30	2008/12/19	\$90,560
Zenaida Frank	Software Engineer	New York	63	2010/01/04	\$125,250

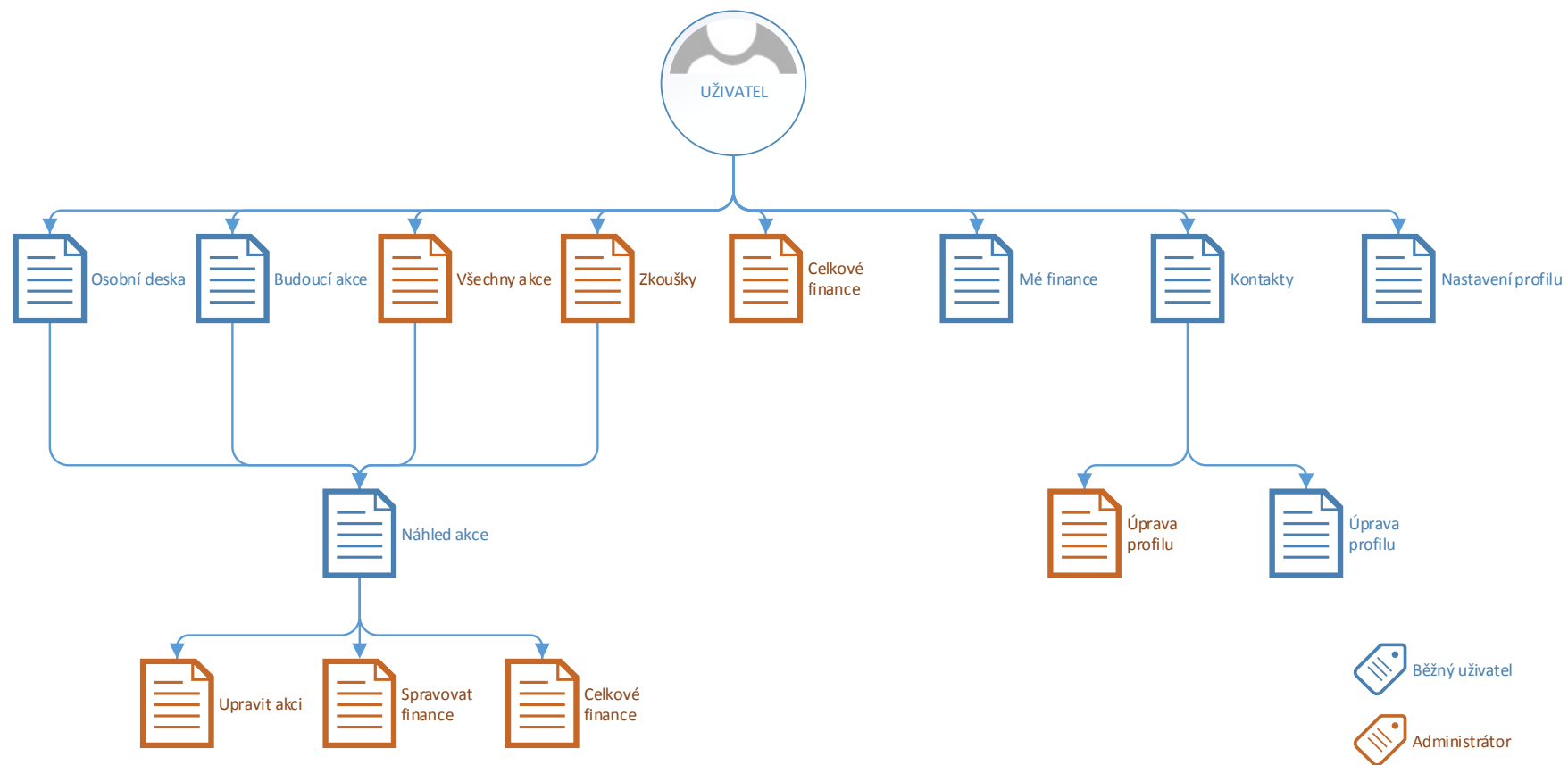
Obrázek 22: Ukázka pluginu DataTables,

zdroj: (28)

3.2.2 PHP pluginy

Další knihovna, která nám ušetří práci, se jmenuje DIBI. Ta zjednodušuje programování SQL. Mimo použití DIBI použiju i univerzální php soubor, ve kterém budu definovat nastavení připojení a základní bezpečnostní funkce a také své dotazy na databázi.

3.3 Struktura informačního systému KAPeLis



Obrázek 23: Struktura informačního systému KAPeLis,

zdroj: vlastní zpracování

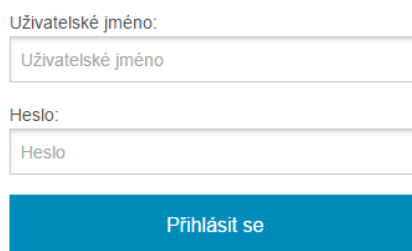
Informační systém jsem pojmenoval KAPELis. Z názvu je patrné, že se jedná o informační systém pro kapely. Návrh struktury systému vychází z analýzy, ve které jsem zachytil momentální potřeby sledovaných informací hudebních těles.

3.4 Bezpečnostní opatření

V této části se budu věnovat bezpečnostnímu opatření navrhovaného informačního systému.

3.4.1 Přihlášení uživatele

Prvním bezpečnostním prvkem informačního systému je zamezení přístupu do systému neoprávněným osobám. Každý uživatel má tedy svůj vlastní účet. Pokud se však nepřihlásí, nesmí se mu zobrazit žádné informace. K tomu slouží bezpečnostní prvek založený na principu SESSIONS.



Uživatelské jméno:

Heslo:

Obrázek 24 : Přihlašovací obrazovka,

Zdroj: vlastní

Každá stránka obsahuje odkaz na univerzální soubor, který mimo jiné ověří, zda je uživatel přihlášený. Pokud tomu tak není, stránka uživatele automaticky přesměruje na přihlašovací stránku systému. Když ale systém zjistí, že uživatel přihlášený je, zobrazí mu potřebné informace. Výhodou tohoto použití je, že se vyhodnocování podmínky provádí na straně serveru, nikoliv použitím skrývání prvků nebo jiných mechanismů, který pokročilý uživatel počítače dokáže prolomit.

Následující kód zajišťuje, že k systému bude mít přístup pouze přihlášený uživatel. Pokud není splněna podmínka přihlášení, uživateli je vykresleno sdělení, že zadal špatně vstupní údaje do systému.

```
$sql = dibi::query("SELECT * FROM hrac WHERE username=' $myusername' AND password=' $mypassword'")->count();
if ($sql==1) {
    session_start();
    header("Cache-control: private");
    $_SESSION["myusername"] = $myusername;
    header("Location: ". $protected);
    exit;
} else {
    echo "Zadali jste špatné jméno nebo heslo.";
}
```

Obrázek 25 : Skript, který ověří uživatelovo přihlášení

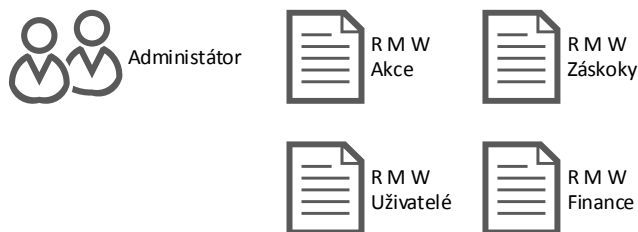
Zdroj: vlastní

Mimo to se i zapíše do proměnné, který uživatel je zrovna přihlášený (jeho ID). Toho budeme využívat především později.

Nyní máme tedy přihlášeného uživatele a můžeme se vrhnout na popis rolí.

3.4.2 Role

Při analýze systému jsem zmínil, že budeme rozlišovat tři druhy uživatelů. Z praxe jsem vypořádal potřebu ještě další role. Všechny role nyní popíšu a uvedu přístup k jednotlivým funkcím.

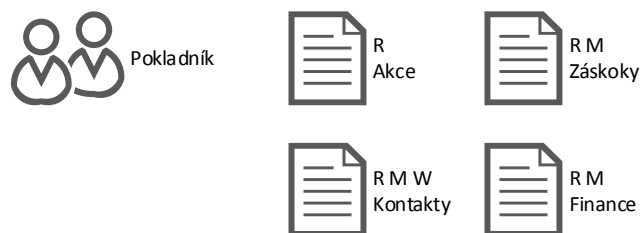


Obrázek 26: Oprávnění uživatelské role Administrátor

Zdroj: vlastní

Administrátorská role patří ke stěžejním rolím systému. Uživatel s touto rolí má přístup k naprosto všem funkcím a informacím, které informační systém poskytuje. Jak můžeme vyčíst

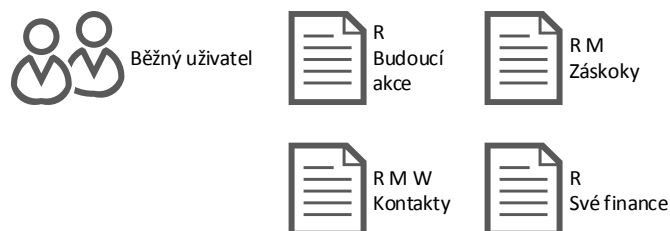
z obrázku, může kompletně spravovat akce, soupisku, uživatele a finance všech osob a akcí. Zpravidla by se mělo jednat o manažera, kapelníka nebo vedoucího daného seskupení.



Obrázek 27: Oprávnění uživatelské role Pokladník

Zdroj: vlastní

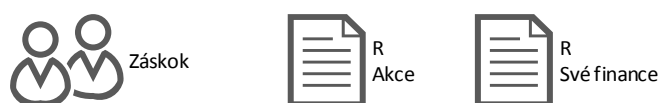
Jelikož pokladníkem může a nemusí být aktivní člen sboru, jsou mu přidělena práva ke čtení akcí, nikoliv však k její editaci či samotnému vytváření. Může také editovat záskoky nebo spravovat kontakty pro případ shánění záskoku za sebe samotného. Má však přístup k výplatám za jednotlivé akce. Díky tomu může uživatelům zapisovat jejich finanční ohodnocení za hudební produkci.



Obrázek 28: Oprávnění uživatelské role Běžný uživatel

Zdroj: vlastní

Běžný uživatel má logicky pouze omezená práva. Má možnost nahlížet pouze do budoucích akcí, může za sebe vybírat záskoky, spravovat kontakty záskoků a číst pouze své finance, které mu byly vyplaceny nebo mají být vyplaceny.

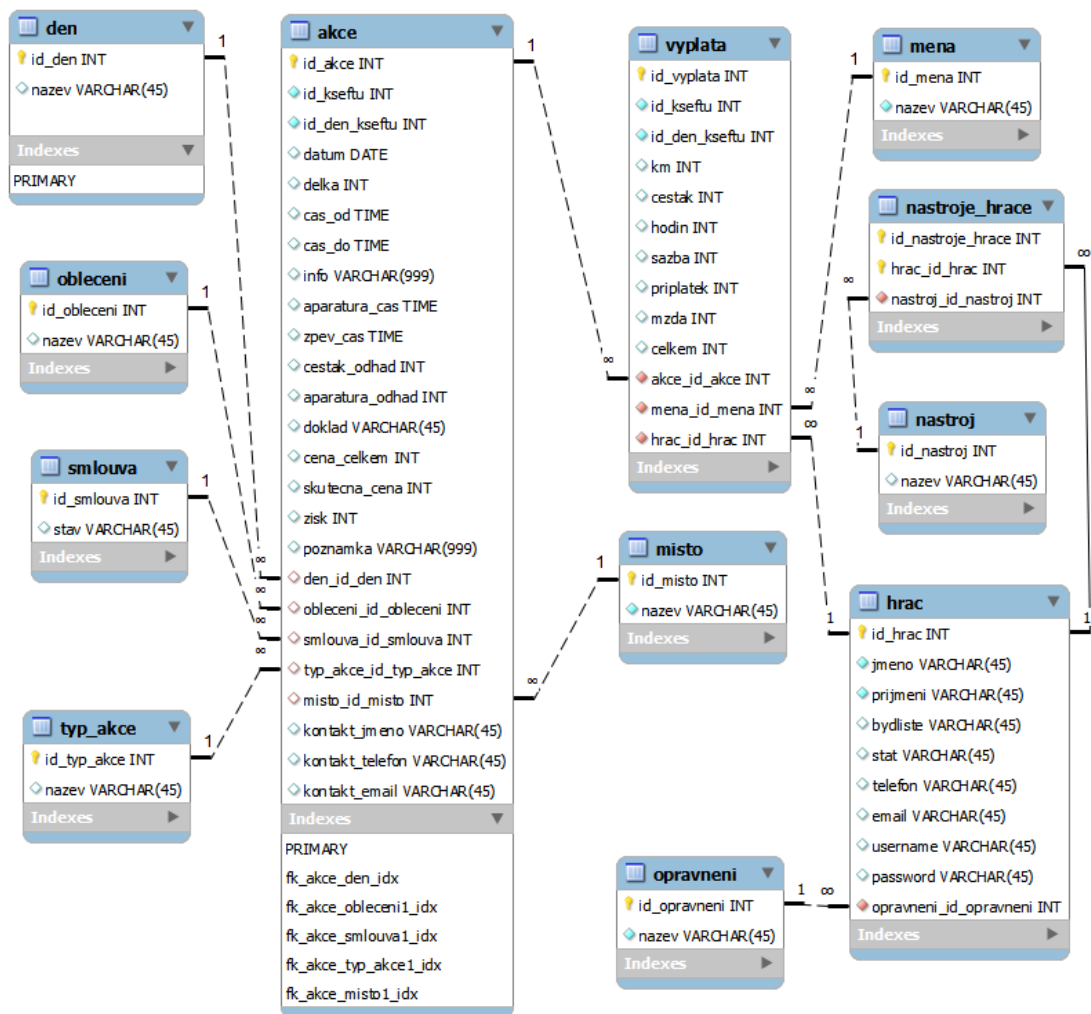


Obrázek 29: Oprávnění uživatelské role Záskok

Zdroj: vlastní

3.5 Návrh databáze

Vzhledem k tomu, že máme už jasno, jak má systém vypadat a jak má být navržený, můžeme se vrhnout na samotný návrh databáze. Návrh je zpracovaný v programu MySQL Workbranch. Výhodou tohoto programu je, že nám na základě návrhu přímo vygeneruje SQL kód databáze.

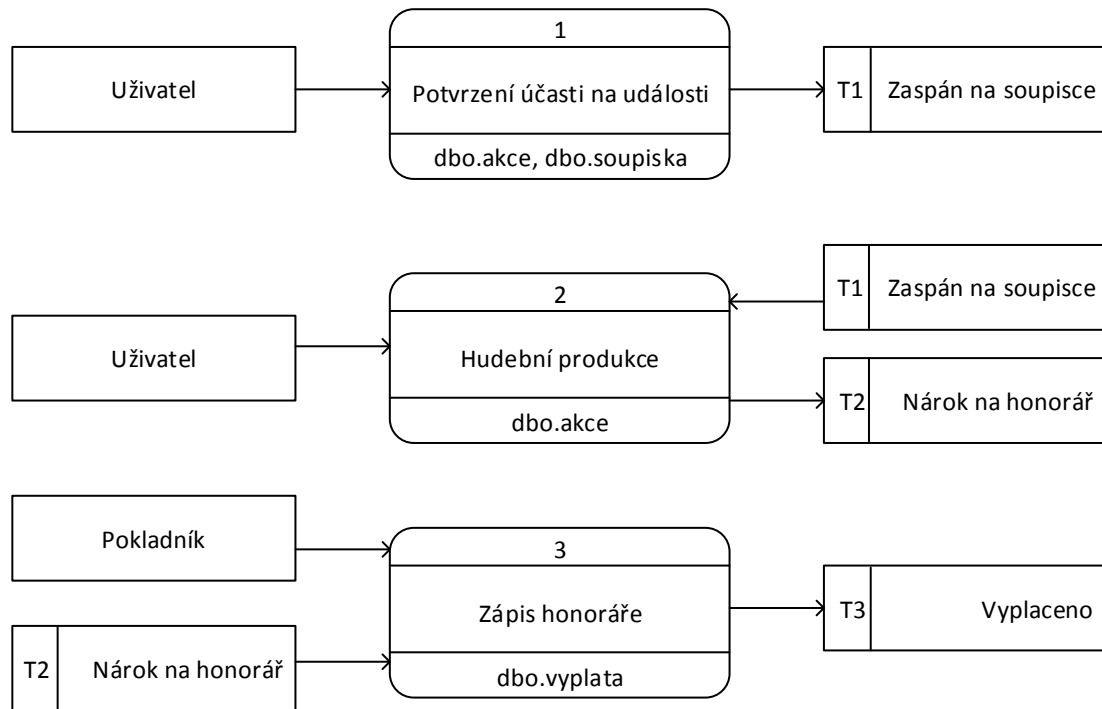


Obrázek 30: Návrh databáze informačního systému

Zdroj: vlastní

3.6 DFD diagram

Jak jsem již zmínil, různé role mají různý přístup k datům a různé povolení s nimi nakládat. Pohyb dat v navrhovaném informačním systému je znázorněn v následujícím modelu:



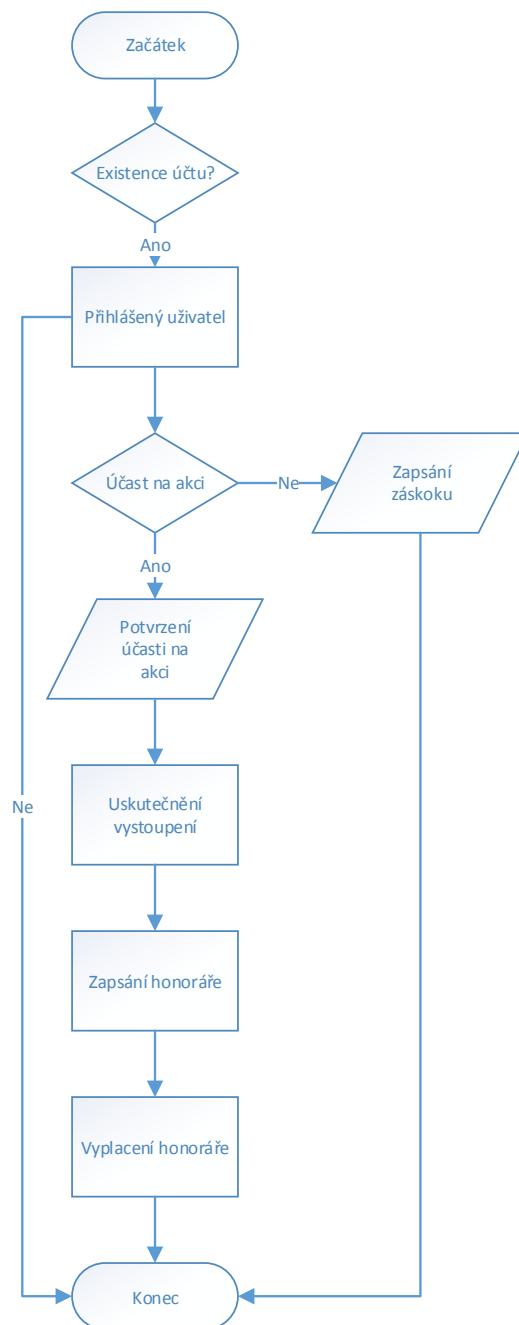
Obrázek 31: DFD diagram

Zdroj: vlastní zpracování

Z DFD diagramu můžeme jasně vidět, jak postupují data při běžném procesu vykonání hudební produkce na nějaké události. Uživatel potvrdí svoji účast a tím pádem je s ním počítáno, že se události zúčastní. Po té, co hudební produkce proběhne, vzniká nárok na honorář za odehranou akci. Tento honorář pak pokladník zapíše do systému. Samotný uživatel pak vidí, jaká částka je nachystaná na vyplacení. Je zde i možnost označení stavu „vyplaceno“, kdy si pokladník může značit, komu byly peníze za událost již vyplaceny a komu ne.

3.7 Vývojový diagram

V následujícím vývojovém diagramu zachycuji běžný postup uživatele IS. Po přihlášení buď potvrdí svoji účast na hudební produkci, nebo za sebe zapíše záskok. Pokud se události zúčastní, je mu následně zapsán a vyplacen honorář.



Obrázek 32: Vývojový diagram

Zdroj: vlastní zpracování

3.8 Komponenty informačního systému

3.8.1 Hlavní menu

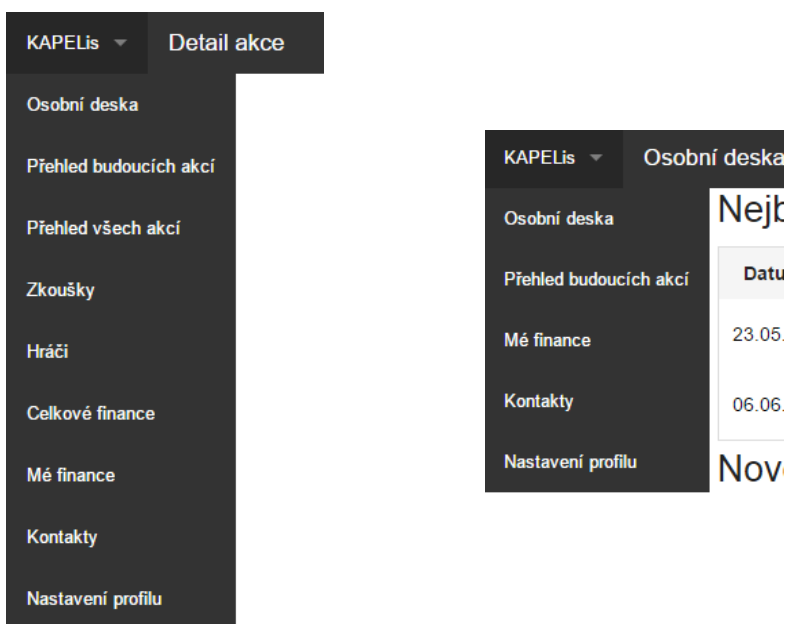
Hlavním ovládacím prvkem systému je navigace. Ta je umístěna v záhlaví systému a je zakotvena tak, že i když se uživatel pohybuje na stránkách níže, záhlaví je vždy k dispozici. Po designové části se jedná o minimalistický styl, který zajišťuje plugin Foundation. Byla zde použita komponenta „top-bar“.



Obrázek 33: Navigace informačního systému

Zdroj: vlastní

Menu se skládá z několika částí. Tou první je samotná navigace v systému, která se při najetí myši rozevře a nabídne tak uživateli kompletní nabídku.



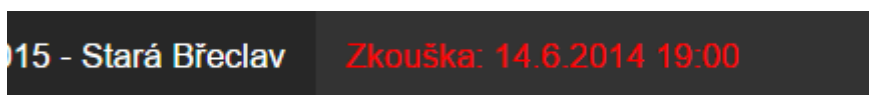
Obrázek 34: Hlavní navigační panel administrátora a běžného uživatele

Zdroj: vlastní

Druhá část uživatele informuje o tom, v jaké části systému se momentálně nachází. Další část navigace slouží k zobrazení aktuálně přihlášeného uživatele a jeho role a zároveň slouží jako tlačítko k přesměrování na uživatelský profil. Poslední tlačítko slouží k odhlášení se ze systému, o kterém jsem se již v rámci této kapitoly zmiňoval.

Mimo tyto statické části navigace slouží pak navigace dynamická. Ty se zobrazují v závislostech na události a místa, kde se uživatel v rámci systému nachází.

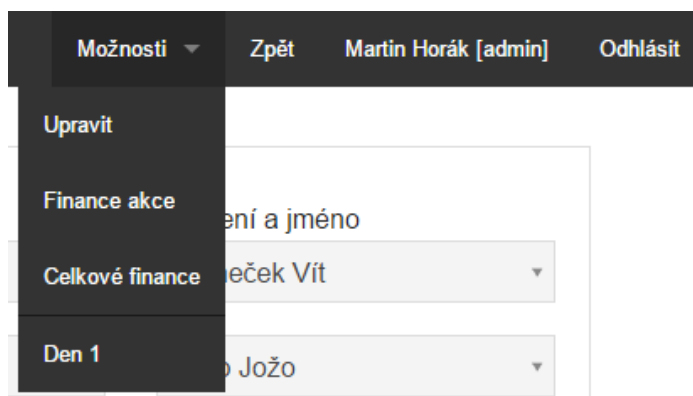
Jednou z nich je část notifikací. Ta slouží k upozornění uživatelů na blížící se zkoušku. Tato notifikace se zobrazí pouze, pokud ne naplánovaná zkouška a informuje uživatele o tom, kdy a v který čas se bude konat.



Obrázek 35: Jednoduchá notifikace o blížící se zkoušce

Zdroj: vlastní

Druhou dynamickou navigací je navigace Možnosti a zpět. Navigace možnosti se zobrazí vždy v momentě, kdy jsou k dispozici různé akce v rámci systému. Například když administrátor zobrazí náhled události a chce ji editovat, upravit finance nebo se jen přepnout na jiný den v rámci nekolikadenní události.



Obrázek 36: Navigace informačního systému „Možnosti“

Zdroj: vlastní

Abychom nastavili různá menu pro různé role, provádíme kontrolu oprávnění na serveru. Proto jsem vytvořil funkci, která vrací id oprávnění aktuálně přihlášeného uživatele. Pomocí ní pak filtruji, co se má v nabídce zobrazit. V následující ukázce je ukázka, jak se provádí filtrování menu na základě oprávnění.

```
<a href='#'>KAPELis</a>
<ul class='dropdown'>
  <li><a href='http://$serverName/is/deska/'>Osobní deska</a></li>
";
  if (getUserPermission() != 2) {
    echo "
  <li><a href='http://$serverName/is/akce/budouci.php'>Přehled budoucích akcí</a></li>
  "; }
  if ( (getUserPermission() == 1) || (getUserPermission() == 4)) {
    echo "
  <li><a href='http://$serverName/is/akce/'>Přehled všech akcí</a></li>
  <li><a href='http://$serverName/is/zkousky/'>Zkoušky</a></li>
  "; }
  if ( (getUserPermission() == 1)) {
    echo "
  <li><a href='http://$serverName/is/hraci/'>Hráči</a></li>
  <li><a href='http://$serverName/is/finance/celkove.php'>Celkové finance</a></li>
  "; }
    echo "
  <li><a href='http://$serverName/is/finance/'>Mé finance</a></li>
  <li><a href='http://$serverName/is/hraci/'>Kontakty</a></li>
  <li><a href='http://$serverName/is/nastaveni/'>Nastavení profilu</a></li>
</ul>
```

Obrázek 37: Nastavení oprávnění v menu

Zdroj: vlastní

3.9 Diskuze

Jelikož je mimo jiné hlavním účelem nového informačního systému ulehčit komunikaci a vytvořit tak nový komunikační kanál, který by byl společný pro všechny, je nutné zahrnout do systému jakýsi modul diskuze. Ta bude dostupná pro každou akci zvlášť a k tomu jedna hlavní diskuze, která bude dostupná na hlavním panelu.

V rámci diskuze se bude zaznamenávat čas příspěvku, kdo příspěvek napsal popř. na jaký příspěvek je nový příspěvek reakcí. Dále je nutno pamatovat na možnost příspěvek smazat nebo odpovědět. Za tímto účelem jsem tedy stvořil modul diskuze. Přidání příspěvku do diskuze vypadá v PHP kódu následovně.


```

if(IsSet($_POST['pridat_novy_prispevek'])){
    $value = array(
        'id_diskuze' => $_GET['id_akce'],
        'text' => $_POST['text'],
        'time' => date("Y-m-d H:i:s"),
        'id_hrac' => getUserId(),
        'id_reakce_na' => NULL
    );

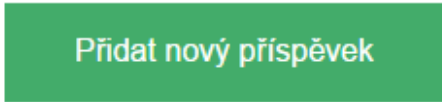
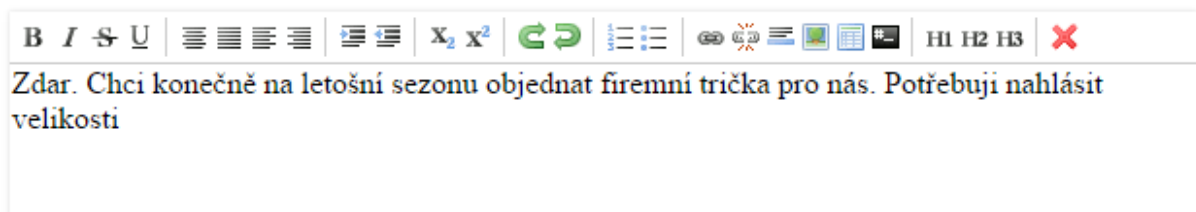
    saveDiskuze(null, $value);
    $idAkce = $_GET['id_akce'];
    $URL="akce_zobraz.php?id_akce=$idAkce";
    header ("Location: $URL");
    exit;
}

```

Obrázek 38: Naprogramovaná diskuze

Zdroj: vlastní

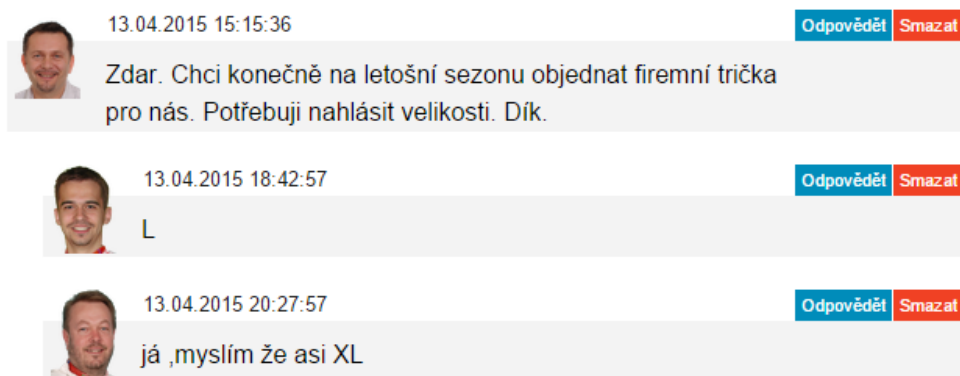
Po uložení dat do databáze se stránka znovu načte a přidáný příspěvek se zobrazí v diskuzi. Mimo to jsem do diskuze zahrnul javascriptový editor wysiwyg, který uživatel zobrazí pohodlný textový editor, do databáze se však uloží výstup z tohoto editoru jako html. Díky tomu je možné uživatelům umožnit základní možnosti nastavení textu, jako jsou různé nadpisy, měnění barev, nové řádky, podtržení atp.



Obrázek 39: Wysiwyg editor pro přidávání příspěvku.

Zdroj: vlastní

Pro zlepšení a zpřehlednění diskuze jsem se rozhodl přidat volbu, zda v diskuzi zobrazovat jména nebo avatary. Myslím, že výsledek s avatary je ještě přehlednější a uživateli dovolí rychleji rozpoznat, kdo je autorem příspěvku v diskuzi.



Obrázek 40: Zobrazení příspěvků v diskuzi.

Zdroj: vlastní

3.10 Moduly

3.10.1 Osobní deska

Osobní deska, jak jsem tento modul nazval, je stránka, která se uživateli zobrazí ihned po přihlášení. Z tohoto důvodu by měla tedy obsahovat jakýsi souhrn informací. Rozhodl jsem se tedy, že jako první a nejdůležitější informací jsou nejbližší události hudebního tělesa. Jako první v pořadí se tedy zobrazí nejbližší akce. Dále pak akce nové, které se označují tagem při vytváření a poslední akce, aby si mohl uživatel snadno najít podrobnosti popř. zobrazit honorář za poslední odehranou akci. Hned pod těmito přehledy se zobrazuje obecná diskuze, kde se mohou domlouvat různé obecné věci, které se netýkají přímo konkrétní události. K tomuto modulu mají přístup všichni uživatelé.

Nejbližší akce										
Datum	Den	Místo	Typ	Oblečení	Od	Do	Info	Aparatura	Zpěv	
23.06.2015	so	Dolní Dunajovice	májová zábava	egyptská čaramura	19:00	23:59	sraz 18:30h	19:00	19:00	
Nové akce										
Datum	Den	Místo	Typ	Oblečení	Od	Do	Info	Aparatura	Zpěv	
16.06.2015	ne	Mikulčice	hody	kroj	14:00	02:00		00:00	00:00	
Poslední akce										
Datum	Den	Místo	Typ	Oblečení	Od	Do	Info	Aparatura	Zpěv	
05.04.2015	ne	Lanžhot	velikonoční zábava	egyptská čaramura	16:00	23:59	sraz 15:30h	16:00	16:00	
14.02.2015	so	Mikulčice	obecní ples	sako	20:00	02:00		20:00	20:00	

Obrázek 41: Přehled na osobní desce.

Zdroj: vlastní

3.10.2 Přehled budoucích akcí

Pro každého uživatele / hráče je nutné, aby měl přehled o budoucích akcích tak, aby si mohl sám rozplánovat svůj rozvrh. K tomu slouží právě tento modul. Ten zobrazuje všechny budoucí akce, které na kterých bude hudební těleso účinkovat. Pokud uživatel zjistí, že se některého vystoupení nemůže zúčastnit, může jednoduše kliknutím na některou z položek zobrazit detail události, ve které za sebe může vybrat zaskakujícího muzikanta.

Přehled budoucích akcí

Zobrazuj záznamů: 50 Najít:

den	Datum	Místo	Typ	Od	Do	Oblečení	Délka	Cena	Zisk	Aparatura	Zpěv
so	2019-05-26	Dolní Dunajovice	májová zábava	19:00	23:59	egyptská čaramura	5	12000	0	19:00	19:00
so	2019-05-26	Tvrdonice	slavnosti	23:00	03:00	egyptská čaramura	4	16000	0	23:00	23:00

Obrázek 42: Přehled budoucích akcí.

Zdroj: vlastní

3.10.3 Detail akce

Tento modul patří dle mého názoru ke stěžejním celého systému. Je to totiž mezník mezi administrací akce a obyčejným zobrazením popř. volbou záskoku. Dělí se na 3 sekce. První z nich je detail samotné události, ve které se sledují požadované informace. Tyto informace vyplňuje administrátor a jsou k dispozici pro všechny uživatele.

Druhou sekcí je samotná soupiska muzikantů, kteří budou na akci vystupovat. Ti jsou při vytváření akce defaultně přiřazeni k akci a každý z uživatelů systému může sám za sebe volit záskok. Z bezpečnostních důvodů toto může člověk činit pouze, dokud nenastane termín události. Poté je tato možnost uživateli automaticky zamítnuta.

Poslední částí detailu akce je diskuze k akci. V ní se muzikanti domlouvají na různých podrobnostech ohledně akce. Např. doprava, oblečení, upozornění na objížďku po cestě, počasí atd.

Obrázek 43: Přehled detailu akce a výběr záskoku

Zdroj: vlastní

Pro výběr záskoku je použita pokročilá javascriptová knihovna Select2, která v sobě obsahuje spoustu nastavení. Její výhodou je mimo jiné i v tom, že umí vyhledávat v nabídce. Pracuje na principu, kdy vezme HTML select, který skryje a jeho obsah přenesou do vlastní komponenty.

K naplnění samotného selectu používám univerzální funkci s konkrétním parametrem.

```
function addSelect($id, $selected, $list, $idColumnName, $valueColumnName) {
    $result = '<select name="' . $id . '>';
    foreach($list as $n => $item) {
        $select = ($item->$idColumnName == $selected) ? ' selected="selected"' : '';
        $result = $result . '<option' . $select . ' value="' . $item->$idColumnName . '>' . $item->$valueColumnName . '</option>';
    }
    $result = $result . '</select>';
    return $result;
}

function listNastroj() {
    return dibi::query("SELECT id_nastroj, nazev FROM nastroj");
}
```

Obrázek 44: Ukázka načítání dat do HTML select komponenty

Zdroj: vlastní

3.10.4 Vytváření nové akce / editace stávající akce

Aby mohl uživatel vytvořit či editovat detaily akce, je nutné mít administrátorské opatření. Pokud ho má, je k dispozici stránka pro vytváření, která zaznamenává důležité údaje. Jedná se o formulář, který sleduje datum a čas události, místo konání, o jakou akci se jedná, v jakém oblečení se hraje atd. Mimo tyto obecné informace ale sleduje i rozpočty na událost – kolik peněz se bude účtovat, kolik akce skutečně stála a kolik na sobě vydělala a také strategické kontakty na pořadatele události. Díky těmto informacím může později vedení vyhodnotit, zda se akce vyplácí nebo mohou srovnávat s předchozími lety. Např. když pořadatel zavolá, že by chtěl kapelu na určitou akci, manažer se může podívat na historii, zda už na dané akci nehráli a pokud ano, hned vidí za kolik a jestli toto vystoupení bylo rentabilní. Na základě těchto informací se pak může rozhodnout, zda nabídku přijmout popř. požadovat navýšení rozpočtu oproti minulým rokům.

Podrobnosti

Nový záznam

Identifikátor akce
432

Den akce [číslo dne]
1

Den
ne

Datum (mm-mm-dd)
2015-04-05

Místo
Lanžhot

Typ akce
veikonoční zábava

Oblečení
egyptská čaramura

Čas od
16:00:00

Čas do
23:59:00

Hodin
8

Čas zpěvů
16:00:00

Čas zvucení
16:00:00

Info
sraz 15:30h

Aparatura odhad
0000

Cesták odhad
0000

Doklad
faktura

Celková cena (dohodnutá)
0000

Skutečná cena
0000

Zisk z akce
0000

Vlastní poznámka
prodlužovalo se do 01:00

Kontaktní osoba
Zdeněk Kulišák

Smlouva
neuvedeno

Telefon
721 459

Email
kulišak.zdenek@atlas.cz

Obrázek 45: Editace / zakládání nové akce

Zdroj: vlastní

3.10.5 Finance akce

Pro pokladníka – správce financí je tato záložka velmi důležitá. Zde totiž bude přiřazovat jednotlivým členům honoráře na základě vnitřních pravidel odměňování. K dispozici má soupisku všech muzikantů, kteří na dané akci účinkovali, a může jim přiřadit honorář. K dispozici má několik různých políček pro upřesnění a pozdější statistiky. Sledují se ujeté kilometry, které se násobí nastavenou hodnotou a přepočítávají se na cestovné. Dále se vyplňuje počet hodin, kdo kolik odehrál a na základě toho se dopočítává násobek hodinové mzdy. Jelikož existují i různé odměny a náhrady, je zde ještě políčko příplatek a následuje celková mzda, která je automaticky dopočítána. Vybírá se i měna a určuje možnost, zdali byla již mzda vplacena.

Nástroj	Jméno	KM	Cest.	Hodin	Sazba	Příplatek	Mzda	Celkem	Měna	Vyplaceno
1. křídlovka	Pavlík Richard	0	0	8	800	200	1000	1000	CZK	<input checked="" type="checkbox"/>
Es klarinet	Žažo Jožo	60	300	8	800	200	1000	1300	CZK	<input type="checkbox"/>
B klarinet	Štvrtecký Vladimír	0	0	8	800	400	1200	1200	CZK	<input checked="" type="checkbox"/>

Obrázek 46: Finance akce

Zdroj: vlastní

Mimo to jsem do systému zakomponoval možnost manuálního dopočítání či doplnění polí. Ty se doplňují a přepočítávají pomocí javascriptových funkcí za použití jQuery. Jakmile uživatel s rolí pokladníka či účetního uloží změny, ostatní uživatelé mají ihned k dispozici přehled o svém honoráři. Ten mají k dispozici v modulu „Mé finance“.

3.10.6 Mé finance

Jak bylo již v předchozí kapitole řečeno, modul Mé finance slouží k přehledu o tom, jaký uživatel kdy dostal honorář včetně jeho detailního rozpočítání. Tento modul se skládá pouze z jedné komponenty a tou je tabulka vylepšená pluginem DataTables. Na této tabulce si můžeme příkladně ukázat, jak takové nastavení tabulky v javascriptu vypadá.

```

var financeUzivate1 = $('#financeUzivate1').dataTable({
  "jQueryUI": true,
  "iDisplayLength": 50,
  "aaSorting": [ [0,'desc'], [4,'asc'] ],
  "fnRowCallback": function( nRow, aData, iDisplayIndex ) {
    /* Append the grade to the default row class name */
    $('#financeUzivate1_wrapper, #financeUzivate1').fadeIn();
  },
  "fnFooterCallback": function ( nRow, aaData, iStart, iEnd, aiDisplay ) {
    var iTotalKm = 0;
    for ( var i=0 ; i<aaData.length ; i++ ) {iTotalKm += aaData[i][5]*1;}
    var iPageKm = 0;
    for ( var i=iStart ; i<iEnd ; i++ ) {iPageKm += aaData[ aiDisplay[i] ][5]*1;}
    var nCells = nRow.getElementsByTagName('th');
    nCells[5].innerHTML = parseInt(iPageKm) +
      ' km z '+ parseInt(iTotalKm);

    var iTotalVypLata = 0;
    for ( var i=0 ; i<aaData.length ; i++ ) {iTotalVypLata += aaData[i][11]*1;}
    var iPageVypLata = 0;
    for ( var i=iStart ; i<iEnd ; i++ ) {iPageVypLata += aaData[ aiDisplay[i] ][11]*1;}
    var nCells = nRow.getElementsByTagName('th');
    nCells[11].innerHTML = parseInt(iPageVypLata) +
      ' Kč z '+ parseInt(iTotalVypLata);

    var iTotalHodiny = 0;
    for ( var i=0 ; i<aaData.length ; i++ ) {iTotalHodiny += aaData[i][7]*1;}
    var iPageHodiny = 0;
    for ( var i=iStart ; i<iEnd ; i++ ) {iPageHodiny += aaData[ aiDisplay[i] ][7]*1;}
    var nCells = nRow.getElementsByTagName('th');
    nCells[7].innerHTML = parseInt(iPageHodiny) +
      ' hod. z '+ parseInt(iTotalHodiny);

    var iTotalMzda = 0;
    for ( var i=0 ; i<aaData.length ; i++ ) {iTotalMzda += aaData[i][10]*1;}
    var iPageMzda = 0;
    for ( var i=iStart ; i<iEnd ; i++ ) {iPageMzda += aaData[ aiDisplay[i] ][10]*1;}
    var nCells = nRow.getElementsByTagName('th');
    nCells[10].innerHTML = parseInt(iPageMzda) +
      ' Kč z '+ parseInt(iTotalMzda);
  }
});

```

Obrázek 47: Implementace jQuery pluginu DataTables

Zdroj: vlastní

V kódu vidíme, že nastavujeme spoustu parametrů tak, abychom si nejen vzhled ale i funkci pluginu přizpůsobili svým potřebám. Pro ukázkou uvedu stručný popis deklarace. Prvním parametrem je vzhled, další implicitně zobrazovaný počet záznamů, ve třetím parametru pak definuji, dle kterých sloupečků chci postupně řadit data. Mimo to jsem se musel vypořádat i s problémem, kdy se zobrazila stránka s klasickou HTML tabulkou a až po čase se přeměnila na tabulku DataTables. Provedl jsem tedy preventivní opatření, kdy tabulku stylem CSS skryji a až po dokončení deklarace tabulky provedu její zobrazení pomocí efektu fadeIn(), který je standardní funkcí jQuery.

Podobně definované tabulky pak používám na všech místech, kde se plugin využívá a to vždy přesně na míru konkrétním potřebám a účelům modulů.

3.10.7 Kontakty

Modul kontakty je ve své podstatě podobný jako předchozí. Zobrazuje pomocí pluginu DataTables tabulku, která obsahuje seznam hráčů. Rozdíl je akorát v tom, že jde na řádek kliknout a uživatel tak může zobrazit detail daného kontaktu. Jelikož se jedná o seznam muzikantů, vypíchl jsem stěžejní údaje o kontaktu přímo do tabulky. Zobrazují zde tedy mimo jména a příjmení i nástroj a telefon s kontaktem. Díky tomu nemusí uživatel zbytečně klikat, aby zjistili podrobnosti. Je zde implementován i filtr, který napomůže rychlejšímu zobrazení kontaktu. Samozřejmě je zde i možnost řadit výsledky v tabulce. Pokud uživatel zjistí, že kontakt na některého z muzikantů je již neplatný nebo neaktuální, kliknutím do řádku tabulky zobrazí detail kontaktu, který může spravovat.

Detail muzikanta

Zpět

Jméno
Martin

Příjmení
Horák

Bydliště
[redacted]

Stát
ČR

Telefon
775317 [redacted]

Email
[redacted]@gmail.com

Uložit změny

Obrázek 48: Detail kontaktu

Zdroj: vlastní

Rozdíl mezi náhledem na kontakt administrátora a běžného uživatele je, že administrátor může spravovat uživatelské jméno, heslo a oprávnění uživatele. Díky tomu nemusí používat nějakou speciální komponentu, ale pouze upravenou současnou komponentu, kde se mu jen zobrazí více polí.

3.10.8 Přehled všech akcí a Celkové finance

Jak již z názvu může být jasno, jedná se o administrátorské moduly, které mohou spravovat veškeré finance a akce s neomezenou historickou platností. Mají tedy k dispozici naprosto všechny historické a budoucí záznamy. Jak již bylo zmíněno, jedná se hlavně o nástroj pro manažery, kteří mají přehled, jaká akce se ve který čas a na jakém místě hrála. Mimo to jsou samozřejmě k dispozici i rozpočty na tyto akce. Slouží ale i k nalezení konkrétních dat o

hráčích. Z přehledu celkových financí lze snadno dohledat, kolik si který muzikant vydělal peněz v určitém časovém horizontu a jiné podrobné statistiky.

3.11 Ekonomické zhodnocení

Ekonomické zhodnocení provedu ve dvou rovinách. První rovinou je poskytování systému formou outsourcingu a druhá rovina formou implementace na hostingový server hudebního tělesa.

3.11.1 Outsourcingová varianta

V momentě, kdy by se informační systém realizovat formou outsourcingu, musíme brát na zřetel nákladnou část této varianty, kterou je pořízení hardwarových prostředků, na kterých informační systém poběží. Mimo samotného nákladu na naprogramování informačního systému musíme přičíst částku na pořízení hardware. Návrh rozpočtu na tuto formu naleznete níže.

Tabulka 4: Odhad jednorázových nákladů na outsourcingovou variantu

Zdroj: vlastní

Druh nákladu	VARIANTA A	VARIANTA B
Vytvoření IS	48 000,-	0,-
Hardware	52 300,-	52 300,-
Software	8 899,-	8 899,-
Kancelář	20 000,-	0,-
Marketing	40 000,-	40 000,-
SUMA	169 199,-	101 199,-

Hned na začátek musím uvést na správnou cestu, proč jsem zakomponoval dvě možné varianty. První varianta, tedy varianta A počítá s tím, že náklady na vytvoření budou proplaceny a také se bude účtovat první nájem kanceláře. Je sice pravdou, že se jedná o fixní náklad, nicméně jsem se rozhodl ho pro ukázkou zahrnout do přehledu pro představu.

Ve druhé variantě však nepočítám s náklady na vytvoření, protože systém jsem stvořil a oživil v rámci diplomové práce a pokud bych chtěl systém provozovat sám, nemám tím pádem žádné náklady. A otázkou je i kancelář. Teoreticky je potřeba jen jednu místnost, kde poběží server, může postačit jedna z místností v domě, kde bydlím. Dalo by se říci, že se tedy jedná o porovnání optimistické a pesimistické varianty – jen záleží na pohledu úhlu, která varianta je optimistická a která pesimistická.

Při kalkulaci mzdy na vytvoření jsem počítal se mzdou 400 Kč/hod a vycházel jsem z předpokladu, že čas na vytvoření takového systému zaber 120 hodin. U ceny, kterou jsem uvedl u hardware a software jsem vycházel z cen uvedených na počítačovém e-shopu firmy CZC.cz. Částka u kanceláře je vyhrazena na vybavení kanceláře.

Celkově se tedy jako jednorázová částka na start provozu systému vyčíslila na cca 170tis. Kč, což není dle mého názoru nijak závratná částka. Bohužel si nejsem jistý, že je to částka, která by se v blízkém časovém horizontu vrátila.

Nyní provedu kalkulaci paušálních provozních nákladů. Do nich zahrnuji náklady na mzdu zaměstnance, nájem kanceláře, spotřebu energií, internetové připojení a náklady na marketing.

Na marketing jsem vyčlenil 40 000,- a to především na zviditelnění produktu. Ke způsobům zviditelnění řadím reklamu na facebooku a jiných portálech, reklamní letáky a brožury, vizitky atd. Pokud by byla reklama účinná, dalo by se uvažovat o navýšení částky.

Tabulka 5: Odhad paušálních nákladů na outsourcingovou variantu

Zdroj: vlastní

Druh nákladu	VARIANTA A	VARIANTA B
Zaměstnanci + odvody státu	280 000,-	140 000,-
Nájem vč. spotřeby energií	55 400,-	0,-
Internetové připojení	7 572,-	7 572,-
SUMA	342 972,-	147 572,-

Opět jsem uvedl dvě varianty. První varianta počítá se náklady na zaměstnance, který je zaměstnán na plný úvazek a provozuje kancelář. Druhá varianta počítá se zaměstnancem pouze na poloviční úvazek a provozovna je jak jsem již v předchozím odstavci zmínil v domě, kde bydlím.

Náklady za kancelář ve variantě A jsou napočítány za kancelář na okraji Brna. Jedná se o malou kancelář (cca 10-15m²). Součástí kalkulace je i náklad na energie.

Cena za internetovou konektivitu je stanovena na základě ceníku UPC, kde se nabízí měsíční poplatek 631 Kč za rychlost 140/20 Mbps. Tarif by měl přinejmenším pro začátek stačit.

Celkově tedy znamená roční poplatek buď téměř 350 tis. Kč nebo 150 tis. Kč. Dle mých osobních předpokladů a odhadů forma outsourcingu má smysl pouze tehdy, pokud se zvolí varianta B a aby se tato varianta vyplatila, musela by být stanovena opravdu dobrá prodejní politika a obchody by musely jít opravdu skvěle. I z tohoto důvodu nejsem přesvědčen, že tato cesta je ta správná.

3.11.2 Varianta jednorázové implementace

Z důvodu omezených finančních prostředků si myslím, že hudební tělesa nebudou mít dostatek financí na to, aby si pronajímala systém měsíčními poplatky. Proto si myslím, že bude pro ně daleko schůdnější varianta, kdy zaplatí jednorázový poplatek za implementaci a tím pro ně vše skončí.

Proto je kalkulace tedy velmi jednoduchá a spočívá pouze v naprogramování informačního systému. Pokud provedu stejnou kalkulaci jako při variantě outsourcingu, vyjde nový informační systém na téměř padesát tisíc korun – pokud si za něj nechám zaplatit. Pokud ne a budu brát na vědomí, že systém je již naprogramovaný, náklady jsou tedy nulové.

3.11.3 Výnosy outsourcingu

Aby byly pokryty jen náklady na provoz u varianty outsourcingu, musel by systém vydělávat nejméně téměř 350 tisíc korun. Bylo by zapotřebí vydělat nejméně 13 tisíc korun měsíčně, aby se zaplatila levnější varianta outsourcingu. Než by se však počet zákazníků vyšplhal na číslo, kdy by bylo schopné tyto náklady pokrýt, trvalo by to několik let. Jelikož se ve sféře hudebních těles pohybují, beru tuto variantu za silně nereálnou.

3.11.4 Výnosy jednorázové implementace

Dle mého názoru je tato varianta nejjednodušší a nejvýnosnější. Provoz a implementaci systému lze provádět při zaměstnání a může se tedy jednat o velmi zajímavý přivýdělek. Systém bych chtěl implementovat za jednorázovou cenu 7 999,- Kč, přičemž by cena obsahovala kompletní instalaci a provedení nastavení. Kdyby se podařilo prodat i jednu licenci za měsíc, znamenalo by to při běžném zaměstnání jistě zajímavý přivýdělek. Jelikož je systém proveden v co nejuniverzálnějším stylu, samotnou implementaci odhaduji na 2-3 hodiny práce včetně nastavení. Ročně by tak systém mohl vydělat několik desítek tisíc korun.

3.12 Celkové zhodnocení

Závěr kapitoly je zpravidla věnován celkovému zhodnocení a ani v mé diplomové práci tomu nebude jinak. Provedu zhodnocení systému pomocí analýzy SWOT.

3.12.1 Silné stránky

K silným stránkám vytvořeného informačního systému KAPELis patří beze sporu přínos ve sjednocení informačního a komunikačního kanálu dohromady. Mimo toto sjednocení vlastní i funkci pro výpočet mezd. Také sleduje strategické informace, které mohou vedoucí souborů, manažeři nebo kapelníci využívat při plánování dalších akcí.

Nelze opomenout ani estetickou stránku systému. KAPELis je navržen v minimalistickém designu s podporou responzivního designu. Uživatelé se tedy mohou bez problémů přihlašovat odkudkoliv ze světa na svých mobilních telefonech či tabletech. Teoreticky by se dal systém využít i při zahraničních cestách, kdy se mohou domlouvat různé srazy apod. přímo v systému.

Další výhodou je možnost externích uživatelů. Pokud si hráč sám za sebe vybere záskok, je možné tomuto záskoku povolit přístup do systému. Ten pak bude mít k dispozici pouze nutně potřebné informace a nehrozí tak únik citlivých dat hudebního tělesa.

3.12.2 Slabé stránky

Ke slabým stránkám patří především závislost na internetu. Se systémem nelze pracovat bez internetového připojení, avšak tato skutečnost je při provozování na straně klienta neodmyslitelným faktem. Jelikož systém je již v ostrém provozu, mohu hrdě prohlásit, že k žádným dalším výrazným slabým stránkám, které by stály za zmínku, jsem nepřišel.

3.12.3 Příležitosti

Máme 21. století a světu vládou technologie. Za velmi výraznou příležitost vidím vytvoření mobilní aplikace. Ta by dokonce mohla potírat slabou stránkou, kterou je nedostupná internetová konektivita. Aplikace by se synchronizovala s databází na serveru, aby mohla být vždy aktuální. Pokud však nebude internetové připojení k dispozici, tak i přes to by mohla uživateli poskytnout poslední zaktualizované informace.

Další příležitost stojí jistě v postavení na trhu. Systém je momentálně nejpropracovanější dostupná aplikace pro správu a management hudebního tělesa. Díky tomu a také dobrému marketingu je možné obsadit trh, dokud není nasycený.

3.12.4 Hrozby

Mezi hlavní a největší hrozby pro informační systém řadím konkurenci a příchod substitučního produktu na trh. Pokud někdo z konkurenčních IT firem dostane podobný nápad, popř. okopíruje nápadu můj a bude mít k dispozici více zdrojů na vybudování lepšího a propracovanějšího informačního systému, může to být pro projekt likvidační.

3.12.5 Outsourcing vs. Implementace

Při srovnání obou variant se přikláním variantě implementace. Člověk ji může provádět při současném zaměstnání, protože není nijak náročná (zabere 2-3 hodiny). Pokud se povede prodat licenci za plánovaných 7 999 Kč, může si tak autor přijít na několik desítek tisíc ročně. V žádném případě nepředpokládám, že by byla klientela tak rozsáhlá, že by jeden člověk nedokázal poptávku uspokojit a prodej systému by se měl stát hlavním předmětem podnikání.

ZÁVĚR

Diplomová práce je soustředěna na analýzu a řešení problematiky informačních systémů na řízení hudebních těles. Na základě analýzy jsem zjistil, že momentálně neexistuje žádný efektivní a přitom jednoduchý informační systém, který by toto řízení uměl.

Na základě stanovených kritérií při analýze jsem pak sestrojil dle mého názoru efektivní systém, který lze použít v široké škále hudebních těles a pomocí kterého se ušetří spoustu času a financí při správě samotného hudebního tělesa. Tento informační systém dokonce již běží v ostrém provozu v patnáctičlenném dechovém orchestru a díky tomu mohu usoudit, že se mi cíl práce povedl.

Při vypracování diplomové práce jsem prohloubil mé znalosti nejen s problematikou informačních systémů na řízení hudebních těles, ale při analýze, návrhu a samotném vypracování jsem ještě více zdokonalil své schopnosti analyticky uvažovat.

Po ekonomické analýze jsem také odkryl možné budoucí profity při prodeji informačního systému. Jelikož se nejedná o informační systém pro firmy, ale většinou pro občanská sdružení a podobné útvary, není možné počítat s vysokými zisky. Nicméně se na druhou stranu nejedná o složitý systém na správu a je možné si díky němu přivydělat k současnému zaměstnání i několik desetitisíců ročně.

SEZNAM POUŽITÉ LITERATURY

- [1] KOCH, Miloš. *Informační systémy a technologie*. Vyd. 3. Brno: Akademické nakladatelství CERM, 2008, 166 s. ISBN 978-80-214-3732-6.
- [2] TVRDÍKOVÁ, Milena. *Zavádění a inovace informačních systémů ve firmách*. 1.vyd. Praha: Grada Publishing, 2000, 110 s. ISBN 80-716-9703-6.
- [3] DOHNAL, Jan a Jan POUR. *Architektury informačních systémů v průmyslových a obchodních podnicích*. 1. vyd. Praha: Ekopress, 1997, 301 s. ISBN 80-861-1902-5.
- [4] TVRDÍKOVÁ, Milena a Jan POUR. *Aplikace moderních informačních technologií v řízení firmy: nástroje ke zvyšování kvality informačních systémů*. 1. vyd. Praha: Grada, 2008, 173 s. ISBN 978-80-247-2728-8.
- [5] MOLNÁR, Zdeněk a Jan POUR. *Efektivnost informačních systémů: nástroje ke zvyšování kvality informačních systémů*. 1.vyd. Praha: Grada Publishing, 2000, 142 s. ISBN 80-716-9410-X.
- [6] BASL, Josef a Jan POUR. *Podnikové informační systémy: podnik v informační společnosti*. 2., výrazně přeprac. a rozš. vyd. Praha: Grada, 2008, 283 s. Management v informační společnosti. ISBN 978-80-247-2279-5.
- [7] KUČEROVÁ, Helena. *Informační systém. Zpracování informací a znalostí* [online]. 2012-03-16 [cit. 2015-03-03]. Dostupné z: <http://info.sks.cz/users/ku/ZIZ/isystem.htm>.
- [8] JAROMÍR, Skřivan. *Databáze a jazyk SQL*. *Interval.cz* [online]. 2000-08-04 [cit. 2013-04-28]. Dostupné z: <http://interval.cz/clanky/databaze-a-jazyk-sql/>.
- [9] BUREŠ, Vladimír. *Znalostní management a proces jeho zavádění: průvodce pro praxi*. 1. vyd. Praha: Grada, 2007, 212 s. ISBN 978-80-247-1978-8.
- [10] LINNEUS, C. *Invented The Index Card*. *ScienceDaily: Your source for the latest research news* [online]. 2009-06-16 [cit. 2015-03-01]. Dostupné z: <http://www.sciencedaily.com/releases/2009/06/090616080137.htm>.
- [11] DEWEY, M. www.aadet.com [online]. c2013 [cit. 2015-03-02]. Dostupné z: http://www.aadet.com/article/Melvil_Dewey
- [12] *Historie. Databáze* [online]. [cit. 2015-03-06]. Dostupné z: <http://www.databaze.chytrak.cz/historie.htm>.

- [13] ŽÁK, Karel. Historie relačních databází. ROOT.CZ [online]. 2001-10-19 [cit. 2015-03-07]. Dostupné z: <http://www.root.cz/clanky/historie-relacnich-databazi/>.
- [14] KOCH, Miloš. *Informační systémy a technologie: průvodce pro praxi*. Vyd. 3. Brno: Akademické nakladatelství CERM, 2008, 166 s. ISBN 978-80-214-3732-6.
- [15] KOCH, Miloš. *Notace relačního modelu*. Brno, 2008 [cit. 2015-03-11]. Učební texty vysokých škol. Vysoké učení technické v Brně. Podnikatelská fakulta.
- [16] RYDVAL, Slávek. *Normální formy*. NaWEBka [online]. 2005-08-07 [cit. 2015-03-12]. Dostupné z: <http://www.rydval.cz/phprs/view.php?cislocclanku=2005123127>.
- [17] MOLNÁR, Zdeněk. *Efektivnost informačních systémů: průvodce pro praxi*. 1.vyd. Praha: Grada Publishing, 2000, 142 s. ISBN 80-716-9410-X.
- [18] PRICE, Jason. *C#: programování databází*. 1.vyd. Praha: Grada, 2005, 623 s. ISBN 80-247-0982-1.
- [19] PROCHÁZKA, David. *PHP 6: začínáme programovat*. 1. vyd. Praha: Grada, 2012, 183 s. Průvodce (Grada). ISBN 978-80-247-3899-4.
- [20] MALLYA, Thaddeus. *Základy strategického řízení a rozhodování: začínáme programovat*. 1. vyd. Praha: Grada, 2007, 246 s. Průvodce (Grada). ISBN 978-80-247-1911-5.
- [21] BLAŽKOVÁ, Martina. *Marketingové řízení a plánování pro malé a střední firmy: začínáme programovat*. 1. vyd. Praha: Grada, 2007, 278 s. Průvodce (Grada). ISBN 978-80-247-1535-3.
- [22] BRÁZA, Jiří. *PHP 5: začínáme programovat*. 1. vyd. Praha: Grada Publishing, 2005, 244 s. Průvodce (Grada). ISBN 80-247-1146-X.
- [23] BROŽA, Petr. *Programování WWW stránek pro úplné začátečníky: začínáme programovat*. Vyd. 1. Praha: Computer Press, 2000, 161 s. Průvodce (Grada). ISBN 80-722-6278-5.
- [24] BERAN, Jiří. W3C: HTML5 je kompletní. *Maxiorel.cz* [online]. 20. 12. 2012 [cit. 2015-03-12]. Dostupné z: <http://www.maxiorel.cz/w3c-html5-je-kompletni>
- [25] W3C. *Browser Statistics*. *w3schools.com* [online]. 11.3.2015 [cit. 2015-03-11]. Dostupné z http://www.w3schools.com/browsers/browsers_stats.asp
- [26] AL], Brian Pfaffenberger .. [et]. *HTML, XHTML, and CSS Bible: začínáme programovat*. 3. Hoboken: John Wiley, 2004, 161 s. Průvodce (Grada). ISBN 07-645-7718-2.

- [27] PÍSEK, Slavoj. *JavaScript: efektní nástroj oživení www stránek*. 1. vyd. Praha: Grada, 2001, 231 s. Průvodce (Grada). ISBN 80-247-0014-X.
- [28] DataTables. *Table plug-in for jQuery* [online]. 2015-04-26 [cit. 2015-04-26]. Dostupné z: <https://www.datatables.net/>

SEZNAM OBRÁZKŮ

Obrázek 1: Data a informace.....	13
Obrázek 2: Faktory se vzájemnými vazbami v informačním systému	14
Obrázek 3: Ukázka Bachmanova způsobu značení kardinality vztahů.....	23
Obrázek 4: Ukázka Martinova způsobu značení kardinality vztahů.....	23
Obrázek 5: Ukázka Chenova způsobu značení kardinality vztahů	23
Obrázek 6: Ukázka inženýrského stylu způsobu značení vztahů.....	23
Obrázek 7: Matice SWOT analýzy	25
Obrázek 8: Statistika jednotlivých prohlížečů za únor 2015.....	27
Obrázek 9: Nejčastěji používaná uživatelská rozlišení – Leden 2014.....	29
Obrázek 10: Uživatelské schéma	32
Obrázek 11: Komponenta diskuze od společnosti Blueboard.cz	35
Obrázek 12: SWOT analýza systému složeného z komponent.....	36
Obrázek 13: Interní webový systém kapely Boršičanka	37
Obrázek 14: Detail události.....	37
Obrázek 15: SWOT analýza interního systému Boršičanky	38
Obrázek 16: Logo interního webu kapely	39
Obrázek 17: První evidenční systém s autentizací	39
Obrázek 18: Správa uživatelů interního webu	40
Obrázek 19: Evidence vystoupení s podrobnostmi a diskuzí.....	41
Obrázek 20: Evidence vystoupení s podrobnostmi a diskuzí.....	42
Obrázek 21: SWOT analýza interního systému Lácaranky	43
Obrázek 22: Ukázka pluginu DataTables,.....	45
Obrázek 23: Struktura informačního systému KAPELis,	46
Obrázek 24 : Přihlašovací obrazovka,.....	47
Obrázek 25 : Skript, který ověří uživatelské přihlášení.....	48
Obrázek 26: Oprávnění uživatelské role Administrátor.....	48
Obrázek 27: Oprávnění uživatelské role Pokladník.....	49
Obrázek 28: Oprávnění uživatelské role Běžný uživatel	49
Obrázek 29: Oprávnění uživatelské role Záskok	49
Obrázek 30: Návrh databáze informačního systému.....	50
Obrázek 31: DFD diagram	51

Obrázek 32: Vývojový diagram	52
Obrázek 33: Navigace informačního systému	53
Obrázek 34: Hlavní navigační panel administrátora a běžného uživatele.....	53
Obrázek 35: Jednoduchá notifikace o blížící se zkoušce	54
Obrázek 36: Navigace informačního systému „Možnosti“	54
Obrázek 37: Nastavení oprávnění v menu	55
Obrázek 38: Naprogramovaná diskuze	56
Obrázek 39: Wysiwyg editor pro přidávání příspěvku.	56
Obrázek 40: Zobrazení příspěvků v diskuzi.....	57
Obrázek 41: Přehled na osobní desce.....	57
Obrázek 42: Přehled budoucích akcí.....	58
Obrázek 43: Přehled detailu akce a výběr záskoku.....	59
Obrázek 44: Ukázka načítání dat do HTML select komponenty	59
Obrázek 45: Editace / zakládání nové akce.....	60
Obrázek 46: Finance akce	61
Obrázek 47: Implementace jQuery pluginu DataTables	62
Obrázek 48: Detail kontaktu	63

SEZNAM TABULEK

Tabulka 1: Příklad agendy.....	33
Tabulka 2: Přehledná soupiska hráčů daného vystoupení.....	33
Tabulka 3: Příklad evidence financí.....	34
Tabulka 4: Odhad jednorázových nákladů na outsourcingovou variantu	64
Tabulka 5: Odhad paušálních nákladů na outsourcingovou variantu	65