

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Portál pro správu hudebních disků a akcí skupiny Story
Music



2015

Vedoucí práce: Mgr. Jiří Zaccal,
Ph.D.

Tomáš Tuček

Studijní obor: Aplikovaná informatika,
prezenční forma

Bibliografické údaje

Autor: Tomáš Tuček
Název práce: Portál pro správu hudebních diskó akcí skupiny Story Music
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2015
Studijní obor: Aplikovaná informatika, prezenční forma
Vedoucí práce: Mgr. Jiří Zaccpal, Ph.D.
Počet stran: 42
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Tomáš Tuček
Title: Portal for Story Music disco events
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2015
Study field: Applied Computer Science, full-time form
Supervisor: Mgr. Jiří Zaccpal, Ph.D.
Page count: 42
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Cílem bakalářské práce je vytvoření webového portálu pro správu hudebních akcí, který obsahuje kalendář, diskusi a fotogalerii. Pro registrovaného návštěvníka umožňuje filtraci volných termínů, rezervaci a editaci termínu akce v kalendáři. Pro administrátora umožňuje filtraci obsazených termínů, správu diskuse a fotogalerií.

Synopsis

The goal of this thesis is to create web portal for disco event management, which includes calendar, discussion and photo gallery. For registred user allows reservation of free date, event management and free date filtering. Also portal provides booked date filtration, discussion and photogalery management for administrator.

Klíčová slova: web; portál; závěrečná práce; asp.net; mvc; kalendář akcí

Keywords: web; portal; thesis; asp.net, mvc; event calendar

Chtěl bych poděkovat vedoucímu práce Mgr. Jiřímu Zaccalovi, Ph.D. za praktické rady a připomínky v průběhu tvorby této bakalářské práce. Dále pak své snoubence Mgr. Vendule Šmoldasové za psychickou podporu, trpělivost a závěrečnou korekci textu.

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

datum odevzdání práce

podpis autora

Obsah

1	Úvod	8
1.1	Zadání bakalářské práce	8
1.2	Analýza	8
1.2.1	Skupina Story Music	8
1.2.2	Současný stav	8
1.2.3	Požadavky skupiny	9
1.2.4	Možnosti řešení	9
1.3	Rozbor existujících aplikací	10
2	Použité nástroje a technologie	12
2.1	Microsoft Visual Studio 2013	12
2.2	ASP.NET	12
2.2.1	WebForms	12
2.2.2	MVC	13
2.3	Microsoft SQL Server	13
2.4	HTML5	14
2.5	CSS3	14
2.6	JavaScript	15
2.7	jQuery	15
2.8	Gimp	15
3	Vývoj aplikace	16
3.1	Posloupnost vývoje	16
3.2	Charakteristika webové aplikace	17
4	Programátorská dokumentace	18
4.1	Adresářová struktura	18
4.2	Modely	19
4.3	Kontrolery	20
4.4	Providery	21
4.5	Databáze	22
4.6	Funkcionalita aplikace	24
4.7	Rozdělení uživatelského rozhraní	27
4.7.1	Záhlaví dokumentu	28
4.7.2	Tělo dokumentu	28
5	Uživatelská dokumentace	29
5.1	Úvodní stránka	29
5.2	Registrace a přihlášení	29
5.3	Diskuse	30
5.4	Fotogalerie	31
5.5	Akce	32
5.6	Administrace	34

6 Nasazení	36
6.1 Minimální požadavky	36
6.2 Přidání aplikace do IIS	36
6.3 Databáze	36
6.4 Konfigurace	36
7 Možnosti dalšího vývoje aplikace	38
Závěr	39
Conclusions	40
Literatura	41
A Obsah přiloženého CD	42

Seznam obrázků

1	Schéma databáze	23
2	Funkcionalita aplikace	24
3	Úvodní strana	29
4	Registrační formulář	30
5	Panel pro práci s příspěvky v diskusi	31
6	Grafický prvek reprezentující fotogalerii	31
7	Kalendář akcí	33

Seznam tabulek

Seznam zdrojových kódů

1	Vlastnost modelu s validací (model)	20
2	Vlastnost modelu s validací (pohled)	20
3	Ukázka práce s providery	22
4	Registrace providerů	22

1 Úvod

1.1 Zadání bakalářské práce

Cílem této práce bylo vytvořit webový portál pro správu hudebních akcí. Portál bude obsahovat informace o skupině Story Music, kalendář jejích akcí, diskusi a fotogalerii.

Portál by měl rozlišovat několik uživatelských rolí:

1. Návštěvník - má umožněno prohlížení webu, registraci a přihlášení, pokud je registrován.
2. Přihlášený uživatel - má umožněnu rezervaci a následnou editaci termínu akce, může volit preferovaný hudební styl akce, editovat seznamy skladeb u akcí a účastnit se diskuse.
3. Administrátor - bude v kalendáři určovat termíny, které nepůjdou rezervovat, spravovat diskusi a fotogalerie všech uživatelů, může zakázat činnost uživatele v diskusi, bude spravovat seznam skladeb, ze kterých je uživateli umožněn výběr skladeb pro akce.

Doporučená technologie vývoje: ASP.NET

1.2 Analýza

1.2.1 Skupina Story Music

Skupina Story Music se zabývá převážně zajištěním hudební kulisy u akcí různého typu (oslav, firemních večírků, svateb a dalších) reprodukovanou hudbou. Ojedinele pomáhá s ozvučením kapel, které nemají vlastní vybavení, nebo jejich vybavení není pro danou příležitost dostatečné.

Akce, na kterých skupina potvrdí svou účast, jsou závazné. Z tohoto důvodu je pro ni důležitý systém, který umožní efektivní správu akcí přehledným a jednoduchým způsobem.

1.2.2 Současný stav

Aktuálně skupina vůbec neprezentuje informace prostřednictvím webových stránek. Kontakt na některého ze členů skupiny je možné získat pouze osobním doporučením.

Rezervace termínu pak probíhá pouze na základě telefonické domluvy, často rozdělené do několika hovorů kvůli ověření dostupnosti termínu v diáři.

Tento způsob komunikace je v dnešní době zastaralý a nepraktický, navíc může vést k nedorozuměním a nespokojenosti zájemce.

Cílem této bakalářské práce je vytvořit webový portál pro prezentaci informací o skupině, který bude obsahovat přehledný kalendář akcí s možností rezervace volného termínu.

1.2.3 Požadavky skupiny

Uživatelé očekávají, že nový systém bude:

- co nejjednodušší na ovládání
- umožňovat efektivní správu akcí
- umožňovat uživatelům diskusi a přidání fotek pro jednotlivé akce
- podporovat jednoduché přidávání a editaci obsahu webu, zejména pak novinek a diskuse, bez znalostí webových technologií
- vyvinut jako bezplatný s nulovými, nebo minimálními náklady na provoz

1.2.4 Možnosti řešení

1. Zdarma dostupné editory pro tvorbu webových stránek

Tyto editory jsou určeny pro tvorbu jednoduchých webových stránek. Práci s databází neumožňují vůbec, nebo ve velice omezené míře. Omezené možnosti mají také při tvorbě složitějšího dynamického obsahu. Kvůli popsaným omezením hodnotím použití těchto editorů jako nevhodnou možnost.

2. Zdarma dostupné redakční systémy

Redakční systémy jsou ve většině případů detailně propracované aplikace, které umožňují intuitivní tvorbu obsahu webových stránek. Nabízí velké množství šablon, díky kterým může uživatel pozměnit téma webu. Jednotlivá témata obsahují balíčky, které mění vzhled a obsah vytvářených webových stránek. Další výhodou je dostupnost mnoha funkčních doplňků. Mezi nejznámější volně šířené redakční systémy patří například Joomla a Wordpress.

Vhodné téma pro prezentaci djské skupiny, vyhovující jejím požadavkům, se bohužel nepodařilo najít. Ani dostupný kalendář není zcela vyhovující pro rezervační systém. Využití některého z dostupných řešení by vyžadovalo značnou úpravu.

3. Vytvoření řešení na míru

Vzhledem k výše uvedeným faktům jsem zvolil právě možnost vytvoření komplexního řešení na míru. Toto řešení bude poskytovat všechny požadované funkce. Pro tvorbu portálu jsem zvolil technologii ASP.NET, dále skriptovací jazyk JavaScript (a knihovnu jQuery). Vzhled portálu je formátován pomocí kaskádových stylů. Tyto i další použité technologie budou popsány dále v textu.

1.3 Rozbor existujících aplikací

Najít na internetu webové stránky, které prezentují hudební, nebo dj skupiny není žádný problém. Stejně tak, pokud hledáme web s rezervačním systémem, najdeme jich mnoho. Horší situace nastane, když hledáme webové stránky hudební skupiny, nebo dj, které obsahují rezervační systém.

Jelikož srovnání webových aplikací podle rezervačního systému není příliš běžné, vybral jsem několik aplikací, které určitým způsobem korespondují se zadáním této práce a níže je zhodnotím.

<http://www.supersaas.cz/>

Jedná se o zástupce univerzálních rezervačních systémů. Tento systém disponuje třemi druhy rozvrhů a velkým množstvím doplňkových funkcí, jako je odesílání informačních e-mailů, nebo sms zpráv, propojení s google kalendářem a mnoho dalších. Je dostupný v několika variantách, z nichž pouze jedna je poskytována zdarma. Uživatel si v aplikaci vytvoří požadovaný rozvrh a na svůj web jej umístí pomocí rámu (iframe), nebo odkazu na rozvrh. Celý systém tedy běží mimo vlastní web uživatele. Přes veškeré možnosti v nastavení vzhledů se mi nepodařilo rozvrh zcela přizpůsobit stylu stránek Story Music. Systém obsahuje vlastní správu uživatelů. Plné propojení s webem Story Music se mi nepodařilo ani v tomto případě. Verze poskytovaná zdarma, která by byla skupinou preferována, navíc obsahuje rušivé reklamy. Vlastní rezervace termínu je však velice jednoduchá a uživatelsky přívětivá. Pokud někdo hledá rezervační systém bez nutnosti registrace, pak systémy tohoto typu jsou jistě vhodné řešení.

<http://www.rezervujstul.cz/>

Aplikace je komerční rezervační systém se zaměřením na restaurace. Systém disponuje jednoduchým a přehledným uživatelským rozhraním. Rezervace je možná bez nutnosti registrace. Po výběru požadovaného stolu, data, času a počtu osob se zobrazí formulář pro zadání osobních údajů a rezervace je potvrzena e-mailem. Práce v prostředí je intuitivní a jednoduchá. V demoverzi umístěné na stránkách projektu jsem nenarazil na žádné výrazné nedostatky, snad jen matoucí použití výstražné červené barvy u tlačítka pro dokončení rezervace. Využití tohoto systému pro správu akcí v kalendáři není možné pro jeho úzké zaměření.

<http://www.djschaff.cz/>

V tomto případě se jedná o webovou prezentaci Dj Schaff. Stránky jsou jednoduché a přehledné. Poskytují potřebné informace o tomto Dj a jeho činnosti. Stránky neobsahují žádný kalendář akcí ani online rezervační systém. Objednání akce probíhá buď pomocí telefonní či e-mailové komunikace, nebo vyplněním k tomu určeného formuláře.

Výše popsané webové aplikace jsou pouze zástupci velkého množství podobně zaměřených aplikací. Na jejich srovnání můžeme vidět, že rezervační systémy se

buď snaží být co nejvíce univerzální a často vyžadují více doplňujících nastavení, ve kterých se uživatel nemusí snadno orientovat, nebo jsou naopak úzce zaměřeny na určitou oblast. Stránky pro prezentaci hudebních, nebo Dj skupin většinou poskytují pouze informace a online rezervační systém nemají vůbec.

Tato bakalářská práce má za cíl vytvoření jednoduchého systému pro správu akcí skupiny Story Music. Zvolil jsem proto vytvoření rezervačního systému zaměřeného na správu jednodenních hudebních akcí.

Otázku, zda rezervační systém v oblasti hudebních akcí bude přínosem, či nikoli, zodpoví až dlouhodobé používání aplikace.

2 Použité nástroje a technologie

2.1 Microsoft Visual Studio 2013

Microsoft Visual Studio je vývojové prostředí společnosti Microsoft, které umožňuje vývoj širokého spektra aplikací od jednoduchých konzolových aplikací, po rozsáhlé programy se složitým GUI¹ a webové aplikace.

Umožňuje použití několika programovacích jazyků, obsahuje přehledný editor kódu a nástroje pro efektivní a jednoduché ladění programu.

2.2 ASP.NET

ASP.NET je technologie vyvíjená společností Microsoft. Hlavní myšlenkou je posunout vývoj webových aplikací co nejbližší vývoji klasických aplikací pro Windows. „ASP.NET je součástí .NET Frameworku pro tvorbu webových aplikací a služeb. Je nástupcem technologie ASP (Active Server Pages) a přímým konkurentem JSP (Java Server Pages).“^[1]

ASP.NET je založen na CLR². Umožňuje tedy využít pro vývoj webové aplikace kterýkoli z jazyků podporujících CLR, nejčastěji C# a VisualBasic, ale například i C++, Jscript a další.

ASP.NET rovněž disponuje propracovaným systémem výjimek, který umožňuje snadné ošetření chyb (například tzv. XSS³ útoku)

V současné době Microsoft nabízí 2 způsoby jak s použitím technologie ASP.NET webové aplikace vytvářet. Jsou to ASP.NET WebForms a ASP.NET MVC.

2.2.1 WebForms

Způsob programování ASP.NET WebForms připomíná tvorbu aplikací pro desktopové prostředí. Při programování webu se používají ovládací prvky (Controls), které mají své vlastnosti a události. Dále se pomocí skrytých formulářových polí uchovávají souvislosti mezi jednotlivými požadavky (tento mechanismus se nazývá ViewState). Přináší tedy do prostředí webu s jinak bezstavovým protokolem HTTP možnost zachování stavu. Tento přístup může programátorovi ušetřit mnoho času, ale přináší s sebou i některé nevýhody^[2].

Mezi největší nevýhody patří:

- **ViewState** - ačkoli je ViewState příjemným pomocníkem, velikost dat v poli ViewState neustále narůstá s každým dotazem. Při špatném návrhu

¹GUI je grafické uživatelské rozhraní, anglicky "Graphical User Interface", které umožňuje ovládání aplikace pomocí interaktivních grafických prvků, jako například ikony, menu a podobně

²CLR (Common Language Runtime) - společné běhové prostředí jazyků .NET, ve kterém se vykonává kód

³Cross-Site Scripting - útok na stránky, kdy uživatel podvrhne místo textu nebezpečný kód, například script v jazyce JavaScript

aplikace může ViewState zabírat i několik stovek kilobytů, které se přenáší s každým dotazem na server a zpět a může vést ke zpomalování odezvy serveru.

- **Omezená kontrola nad HTML** - ovládací prvky vykreslují samy sebe jako HTML, ale ne vždy takové, jaké očekáváme. Může například docházet k nepředvídané tvorbě identifikátorů prvků a následně obtížnému přístupu pomocí Javascriptu.

2.2.2 MVC

ASP.NET MVC je technologie vývoje webových aplikací využívající architekturu model-view-controller. Tato architektura klade důraz na rozdělení aplikace do tří komponent tak, aby na sobě byly v maximální možné míře nezávislé. Takovéto rozdělení zajišťuje snadnou udržovatelnost aplikace, lepší možnosti testování a celkově čistější strukturu celé aplikace. Každá ASP.NET MVC aplikace je rozdělena do tří hlavních částí.

- **Model (model)** - v architektuře MVC modely představují ty části aplikace, které pracují přímo s daty. Model má svůj vnitřní stav, obsahuje aplikační logiku a poskytuje rozhraní, pomocí kterého je možné s modelem manipulovat.
- **View (pohled)** - je interaktivní část aplikace, tj. webová stránka. Stará se o zobrazení dat z modelu v uživatelsky přijatelné podobě. Poskytuje rozhraní pro práci s modelem (například formulář). Také se stará o odeslání dat zadaných uživatelem do kontroleru.
- **Controller (řadič)** - vyřizuje příchozí požadavky od uživatele, provádí potřebné operace nad modely a vybírá pohled pro zobrazení vyžádaných dat. Zajišťuje tedy komunikaci mezi modelem a pohledem.

Vzhledem ke zmíněným nevýhodám WebForms byla pro portál Story Music zvolena technologie MVC4⁴.

2.3 Microsoft SQL Server

Microsoft SQL Server (často zkráceně MSSQL) je „vysoce výkonný relační databázový systém založený na architektuře client/server. Má integrovanou podporu jazyka XML, je škálovatelný a bezpečný“^[3]. Byl navržen pro zpracování online transakcí (OLTP⁵), ukládání a uchování dat (datové sklady).

Podporuje uložené procedury, trigger, transakce, reporty, fulltextové vyhledávání apod. Obsahuje nástroje pro analýzu zatížení a optimalizaci databáze. Přehledné GUI poskytuje nástroj SQL Server Management Studio.

⁴V době zadání práce byla MVC4 poslední stabilní verzí. Dnes je aktuální verze MVC5

⁵Online Transaction Processing - technologie uložení dat v databázi. Umožňuje snadnou a bezpečnou modifikaci dat ve víceuživatelském prostředí

Dotazování probíhá pomocí jazyka SQL (Structured Query Language).

2.4 HTML5

HyperText Markup Language (zkratka HTML) je značkovací jazyk používaný pro tvorbu webových stránek. Aktuální je, po dlouhé pauze ve vývoji, verze HTML5. Tato verze je zpětně kompatibilní se svými předchůdci, ale navíc eliminuje některé zastaralé a nepoužívané prvky, které nyní často nahrazuje formátování pomocí kaskádových stylů (například tagy font a big pro úpravu textu, nebo frameset pro rozvržení stránky). Naopak přináší řadu nových prvků, které jsou pro moderní web důležité nebo přínosné. Uvedme třeba podporu multimédií (audio a video), podporu vykreslování vektorové grafiky, formulářové prvky pro některé datové typy (e-mail, datum apod.), události pro „drag and drop“, atribut pro data a mnoho dalšího. Přestože nejznámější prohlížeče podporují HTML5, stále můžeme narazit na prvky s omezenou podporou, jako příklad uvedeme formáty multimediálního obsahu.

2.5 CSS3

Kaskádové styly (CSS) popisují, jak se má daná HTML stránka zobrazit (určuje rozvržení prvků, barvy, chování atd.).

Kaskádové styly se zapisují do samostatných souborů dostupných napříč aplikací. Tím zamezují vzniku redundantního kódu a také umožňují oddělení obsahu stránky od jejího vzhledu.

Styly se na jednotlivé prvky navážou pomocí selektorů. Těmi mohou být identifikátory, třídy, nebo tagy, ale i další.

V současné době je aktuální verzí CSS3. Třetí verze obsahuje vše z verze 2 a navíc přináší mnoho nových možností, které se v předchozích verzích musely řešit přidáním HTML tagů, scriptů, nebo obrázků, což komplikovalo kód stránky.

Mezi vítané novinky můžeme zařadit:

- barevné přechody
- stíny
- zakulacené rohy
- přehlednější strukturu layoutů
- několikanásobná pozadí
- průhlednost a barvy v RGBA formátu
- vlastní fonty
- nové selektory (např. `a[href$=".cz"]` - všechny odkazy, kde atribut href končí řetězcem ".cz")

2.6 JavaScript

JavaScript je multiplatformní objektově orientovaný prototypový interpretovaný jazyk, který se používá nejčastěji jako programovací jazyk pro webové stránky. Je podporován většinou dnešních prohlížečů a umožňuje běh skriptů na straně klienta, manipulaci s prohlížečem, dynamickou úpravu webové stránky bez komunikace se serverem, načítání nového obsahu některých bloků bez aktualizace stránky, asynchronní komunikaci se serverem (AJAX), vytváření dynamických prvků stránky apod. Standardizovanou verzí jazyka je ECMAScript.

2.7 jQuery

jQuery je rozšířená a široce podporovaná javascriptová knihovna. Tak jako CSS odděluje vzhled stránky od obsahu, jQuery odděluje chování prvků od struktury HTML (například specifikace on-click události není v kódu HTML, ale stránka řízená jQuery najde vhodný element a potom přiřadí jeho on-click události vhodnou obsluhu). Knihovna značně zjednodušuje scriptování na straně klienta mimo jiné pomocí:

- jednoduchého výběru, manipulace a úpravy elementů v DOM⁶
- manipulace s CSS
- efektů a animací
- událostí
- AJAXu

2.8 Gimp

„GIMP neboli GNU Image Manipulation Program („GNU program pro úpravy grafiky“) je svobodná multiplatformní aplikace pro úpravu a vytváření rastrové grafiky. Používá se zejména pro úpravy fotografií, tvorbu webové grafiky a podobné účely.“^[4]

Všechny grafické prvky použité pro úpravu vzhledu webu StoryMusic, které nejsou vytvořené pomocí CSS, vznikly v programu GIMP.

⁶Document Object Model - objektová reprezentace webové stránky

3 Vývoj aplikace

Cílem této kapitoly je přiblížit čtenáři, jak jsem postupoval při studiu problematiky a vývoji aplikace.

3.1 Posloupnost vývoje

Vývoj aplikace jsem zahájil navržením funkcionality a zpracováním vzhledu aplikace. Tento návrh jsem konzultoval se zástupcem skupiny Story Music, který bohužel neměl žádné připomínky. Později jsem vzhled aplikace konzultoval ještě se svým vedoucím bakalářské práce. Zpracoval jsem jeho připomínky, kompletně přetvořil vzhled aplikace a výsledek jsem použil jako základ pro další práci.

Na počátku vývoje jsem aplikaci vytvářel v technologii ASP.NET WebForms. Bylo to z důvodu lepší dostupnosti literatury i širší komunity uživatelů, kteří mohli pomoci s řešením případných problémů. Ze začátku jsem tuto technologii považoval za dobrou volbu, programování bylo jednoduché a velice podobné vývoji klasických aplikací pro Windows. Později se začaly projevovat nedostatky technologie, nebo její nedostatečné pochopení. Jako příklad uvedu, že si elementy stránky generovaly své vlastní identifikátory a ve většině případů měly předvytvořené obsluhy událostí. Hledání a odstraňování chyb bylo velice pracné a zdlouhavé.

Pozastavil jsem další vývoj aplikace a začal se více zajímat o technologii ASP.NET MVC 4. Technologie má rovněž svá omezení, ale celkově mi práce v ní vyhovovala mnohem více už z důvodu, že jsem měl plnou kontrolu nad vytvářeným kódem. Z původní WebForms aplikace jsem pro další práci použil pouze upravenou šablonu vzhledu a kaskádové styly.

U technologie MVC jsem narazil na nedostatek kvalitní literatury v českém jazyce. Hlavními zdroji v anglickém jazyce byly oficiální stránky technologie[5] a knihy *Pro ASP.NET MVC 4*[2] a *Professional ASP.NET MVC 4*[6]. Použitou literaturu pro studium ASP.NET WebForms v práci vůbec neuvádím, protože v současné verzi aplikace není vůbec využita.

Po základním seznámení s technologií a vytvoření šablony vzhledu bylo dalším krokem vytvoření datové vrstvy. Aplikaci jsem vytvářel ve vývojovém prostředí Microsoft Visual Studio 2012, proto jsem zvolil databázi Microsoft SQL Server 2012 Express. Přístup k databázi je možný několika způsoby. Nabízí se například využití Microsoft Entity Frameworku. Tento framework umožňuje svázat objekty aplikace s databází tak, že příslušné tabulky i metody pro práci s nimi vytvoří framework sám. Pro tuto aplikaci jsem zvolil vytvoření vlastních rozhraní a implementujících tříd pro objekty ukládané do databáze.

Předpokládám, že tento návrh umožní snadnější udržitelnost datové vrstvy, jednodušší úpravy například při změně struktury databáze a lepší možnosti rozšíření aplikační logiky.

Nejzajímavější problém na který jsem při práci s databází narazil je dotazování na příspěvky v diskusi. Každý příspěvek, kromě prvního v tématu, je reakcí na jiný příspěvek. Z tohoto důvodu bylo nutné využít rekurzivních dotazů[7].

Následně jsem vytvořil vlastní logiku aplikace, využívající již existující datovou vrstnu, a příslušné webové stránky pro komunikaci s uživatelem. V této části vývoje bylo klíčové pochopení komunikace a předávání dat mezi řadičem a pohledem, případně mezi metodami, nebo řadiči. K předávání dat mezi řadičem a pohledem slouží dynamický objekt ViewBag a kolekce ViewData. Pro předávání dat mezi metodami, nebo řadiči, kdy dochází k přesměrování, je určena kolekce TempData[8].

Pro správu rolí a uživatelských účtů jsem implementoval systém rolí a členství, který je popsán v části 4.4.

K zakódování hesel uživatelů jsem použil algoritmus SHA1 prostřednictvím třídy `System.Security.Cryptography.SHA1CryptoServiceProvider`.

Posledím krokem vývoje bylo ošetření neočekávaných stavů aplikace a dalších chyb jako je například zadání neexistující adresy v rámci aplikace. Příjemným usnadněním této části vývoje byl propracovaný systém výjimek v ASP.NET, díky kterému je možné zachytávat neošetřené chyby v aplikaci a korektně na ně reagovat.

3.2 Charakteristika webové aplikace

Výsledná webová aplikace je informační systém, který:

- rozlišuje přihlášeného a nepřihlášeného uživatele, u přihlášeného uživatele rozlišuje role „admin“ a „uživatel“ a na základě rozpoznání uživatele zobrazuje obsah
- obsahuje kalendář akcí s možností rezervace termínu
- umožňuje filtraci volných a pro administrátora rovněž rezervovaných termínů v kalendáři
- obsahuje diskusi a fotogalerii
- umožňuje přidání diskuse a fotogalerie skladeb ke každé potvrzené akci
- podporuje seznamy skladeb, které mají a které nesmí na akci zaznít
- obsahuje kategorie, kam administrátor může přidávat obsah

4 Programátorská dokumentace

Aplikace je rozvržena do řady adresářů, podadresářů a souborů. Některé z nich byly automaticky vygenerovány při zakládání projektu a často mají speciální význam, jiné byly postupně přidávány. Jejich rozmístění a význam nastíníme níže.

4.1 Adresářová struktura

- **App_Data** - automaticky generovaný adresář pro data aplikace, je zde umístěna databáze projektu nazvaná StoryMusicData.mdf
- **App_Start** - automaticky generovaný adresář, ve které jsou soubory s různým nastavením pro start aplikace (nastavení routování, zachytávání výjimek apod.)
- **Content** - automaticky generovaný adresář pro umístění scriptů a prvků ovlivňujících vzhled aplikace, obsahuje:
 - **Obrazky** - obrázky použité pro vzhled aplikace
 - **Skripty** - obsahuje soubory pro jQuery a kompletní sadu funkcí StoryMusicSkripty.js
 - **Styly** - hlavní kaskádové styly aplikace, styly menu a gridů
- **Controllers** - adresář obsahující všechny kontrolery (řadiče). Právě zde aplikace hledá vhodný řadič pro vyřízení příchozího požadavku.
- **Fotogalerie** - adresář do kterého se ukládají fotografie nahrané uživateli. Nejprve jsou uloženy do podadresáře Temp, následně po úpravě velikosti jsou přesunuty do adresáře podle názvu galerie, do které patří. Původní fotografie je z Temp odstraněna. Pokud adresář chybí, bude automaticky vytvořen při nahrávání první fotografie.
- **JineSoubory** - adresář pro dočasné uložení souborů se seznamem skladeb přidávaných do databáze. Soubor je následně načten, zapracován do databáze a odstraněn. Pokud adresář chybí, bude automaticky vytvořen při prvním přidání souboru se skladbami.
- **Models** - v adresáři jsou umístěny modely, které slouží k interakci s uživatelem (model pro registraci, přihlášení, vytvoření akce atp.), nebo jiné modely, které nejsou předávány pohledům, ale nepracují s daty v databázi.
- **Tridy** - v adresáři jsou umístěny datové modely, pomocné funkce a jiné, obsahuje:
 - **DataModel** - adresář obsahuje všechny třídy pro práci s daty

- **DataPristup** - adresář s třídami, které se starají o vlastní komunikaci s databází
 - **Provider** - obsahuje třídy pro práci s integrovanými providery v ASP.NET
 - **Rozhrani** - obsahuje třídy poskytující rozhraní třídám v adresáři DataPristup
 - **Staticke** - zde jsou statické třídy pro kódování hesel pomocí SHA1⁷ a pomocné funkce
- **Views** - adresář obsahuje podadresáře, jejichž názvy korespondují s názvy řadičů ve složce Controllers. V těchto podadresářích pak najdeme pohledy volané příslušnými metodami řadiče. Navíc v adresáři najdeme:
 - **Shared** - adresář pro sdílené pohledy webu. Jsou zde `_fotkaDlazdice.cshtml`, `_galerieDlazdice.cshtml` a `Chyba.cshtml`. První dvě zmínované jsou uživatelsky definované prvky vytvořené pomocí tzv `PartialView`, které se používají v pohledech různých řadičů. `Chyba.cshtml` je pohled, který vypíše informace o chybě v uživatelsky přijatelné formě.
 - **__ViewStart.cshtml** - v tomto souboru je definována šablona pro každý pohled webu (pokud v pohledu není uvedeno jinak)
 - **Sablona2.cshtml** - šablona vzhledu celého webu
 - **Web.config** - tento konfigurační soubor se v aplikaci StoryMusic nevyužívá, obecně slouží například k dodatečné definici uživatelů, kteří mají právo přistupovat k pohledům a proto je vyžadován.
 - **Global.asax** - obsahuje kód reagující na události vzniklé na aplikační úrovni (například `Application_Start`)
 - **packages.config** - soupis balíčků potřebných pro běh aplikace, tento soupis umožňuje automatické stažení chybějících částí při překladu.
- Web.config** - hlavní konfigurační soubor aplikace

4.2 Modely

V této části přiblížíme některé významné modely.

- **Akce** - je model, který představuje naplánovanou akci v kalendáři. Má na starost její vytváření, získávání, úpravu a mazání.
- **Clanek** - je model určený pro vytváření, získávání, úpravy a odstranění administrátorem přidávaných článků (do sekce „O nás“).

⁷Secure Hash Algorithm - hešovací funkce, která ze vstupních dat vytváří "otisk"fixní délky, vyznačuje se tím, že malá změna na vstupu vede k velké změně na výstupu

- **Fotka** - je model reprezentující fotografie (myšleno soubory ve formátu jpg, jpeg, png, gif, tiff, nebo bmp), každá fotografie je v aplikaci reprezentována instancí této třídy. Pomocí statických metod se model Fotka rovněž stará o uložení, změny, získání a odstranění informací o fotce z databáze.
- **Fotogalerie** - model má na starosti vytváření, získávání, úpravy a odstranění fotogalerií.
- **Osoba** - model pro správu uživatelů portálu. Dědí ze třídy MembershipUser (základní třída, která se v ASP.NET používá pro správu uživatelů) a tím umožňuje využití integrovaných nástrojů ASP.NET pro správu uživatelů a uživatelských rolí. O tomto se zmíníme později.
- **Prispevek** - model má na starost vytváření, získávání, úpravu a mazání příspěvků v diskusi.
- **Skladba** - model reprezentuje hudební skladbu a má na starost manipulaci s ní (získávání, vytváření, úpravy a mazání).

Kromě výše zmíněných modelů aplikace obsahuje ještě modely pro interakci s uživatelem. V těchto modelech se často setkáme s atributy knihovny `System.ComponentModel.DataAnnotations`, které ulehčují validaci modelu předávaného zpět aplikaci. Použití atributů v modelu ukazuje kód 1, jeho využití v pohledu pak kód 2.

```

1     [Required(ErrorMessage = "Telefonní číslo musí být zadáno!")]
2     [RegularExpression(@"^(\\+420)? ?[0-9]{3} ?[0-9]{3} ?[0-9]{3}\\$",
3         ErrorMessage = "Špatný tvar čísla, očekává se 777 777 777, nebo
4         777777777, nebo +420 777 777 777!")]
5     public string Telefon { get; set; }

```

Zdrojový kód 1: Vlastnost modelu s validací (model)

```

1     @Html.TextBoxFor(model => model.Telefon, new { @class = "
2         mujTextBox" })
3     @Html.ValidationMessageFor(model => model.Telefon, "", new {
4         @class="hlaskaValidace" })

```

Zdrojový kód 2: Vlastnost modelu s validací (pohled)

4.3 Kontrolery

Všechny kontrolery dědí z abstraktní třídy `Controller`. Tato třída poskytuje metody pro zpracování HTTP požadavků. Název každého kontroleru v adresářové

struktura vypadá takto: „<název>Controller.cs“. Pro přehlednost a jednoduchost v následujícím výčtu vynecháme „Controller.cs“ a uvedeme si pouze názvy. Tedy místo „AkceController.cs“ uvedeme pouze „Akce“ apod.

- **Administrace** - umožňuje administraci některých částí webu (například přidávání článků, výpis uživatelů a změnu jejich rolí).
- **Akce** - obsahuje metody pro správu akcí v kalendáři (jejich vytvoření, úpravy, zobrazení, mazání a další pomocné metody, ověřuje práva uživatele na přidání, nebo úpravu akce, pokud to situace vyžaduje, informuje uživatele a administrátory o vytvoření akce, nebo provedených změnách).
- **Diskuse** - obsluhuje diskusi (zobrazení témat, zobrazení příspěvků k tématu, úpravu a mazání témat a příspěvků a další).
- **Domu** - je první dotazovaný řadič po příchodu na stránky, zobrazuje základní informace o skupině dostupné všem uživatelům.
- **Fotogalerie** - obsahuje metody pro práci s fotkami a fotogaleriemi (jejich vytvoření, úpravu údajů, zobrazení, mazání a další pomocné metody; ověřuje práva uživatele na přidání, nebo úpravu fotky a fotogalerie; pokud to situace vyžaduje, informuje uživatele a administrátory o vytvoření akce, nebo provedených změnách).
- **Chyba** - tento řadič obsahuje jedinou metodu pro zpracování vyvolané neošetřené výjimky a zajištění přípustné reakce (informuje uživatele o chybě). O zachycení výjimky se stará obsluha `public void Application_Error(Object sender, EventArgs e)` registrovaná v souboru `Global.asax`.
- **Mail** - obsahuje metody pro odesílání pošty.
- **Skladby** - zajišťuje manipulaci se skladbami a seznamy skladeb (jejich vytvoření, uložení, získání, odstranění a další pomocné metody).
- **SouboryAPI** - zajišťuje ukládání nahrávaných souborů do příslušných adresářů aplikace.
- **Ucet** - obsahuje metody pro práci s uživatelskými účty (registraci, přihlašování uživatele, změnu údajů, změnu hesla apod.).

4.4 Providery

- **MujMembershipProvider** - dědí ze třídy `MembershipProvider`. Ta poskytuje základní sadu metod pro práci s uživatelským účtem (vytváření, úprava, reset hesla apod.).
- **MujRoleProvider** - dědí ze třídy `RoleProvider` a poskytuje metody pro práci s uživatelskými rolmi (vytvoření a odstranění role, přidání role uživateli, získání rolí pro uživatele apod.).

U obou zmíněných providerů jsou implementovány pouze metody, které se v portálu využívají. Použití ostatních metod by vyvolalo výjimku „NotImplementedException“.

Využití providerů v aplikaci přináší značné zjednodušení ověřování práv uživatele na konkrétní funkcionalitu, jak naznačuje zdrojový kód 3.

```
1 [HttpGet]
2 [Authorize(Roles = "admin")]
3 [ActionName("PridatClanek")]
4 public ActionResult PridatClanek()
5 {
6     // kód metody
7 }
```

Zdrojový kód 3: Ukázka práce s providery

Atribut `Authorize` označuje, že metodu `PridatClanek` smí volat pouze přihlášený uživatel, nepovinný parametr `Roles` pak navíc omezuje přístup pouze pro uživatele v roli `admin`. O vše ostatní se starají metody providerů.

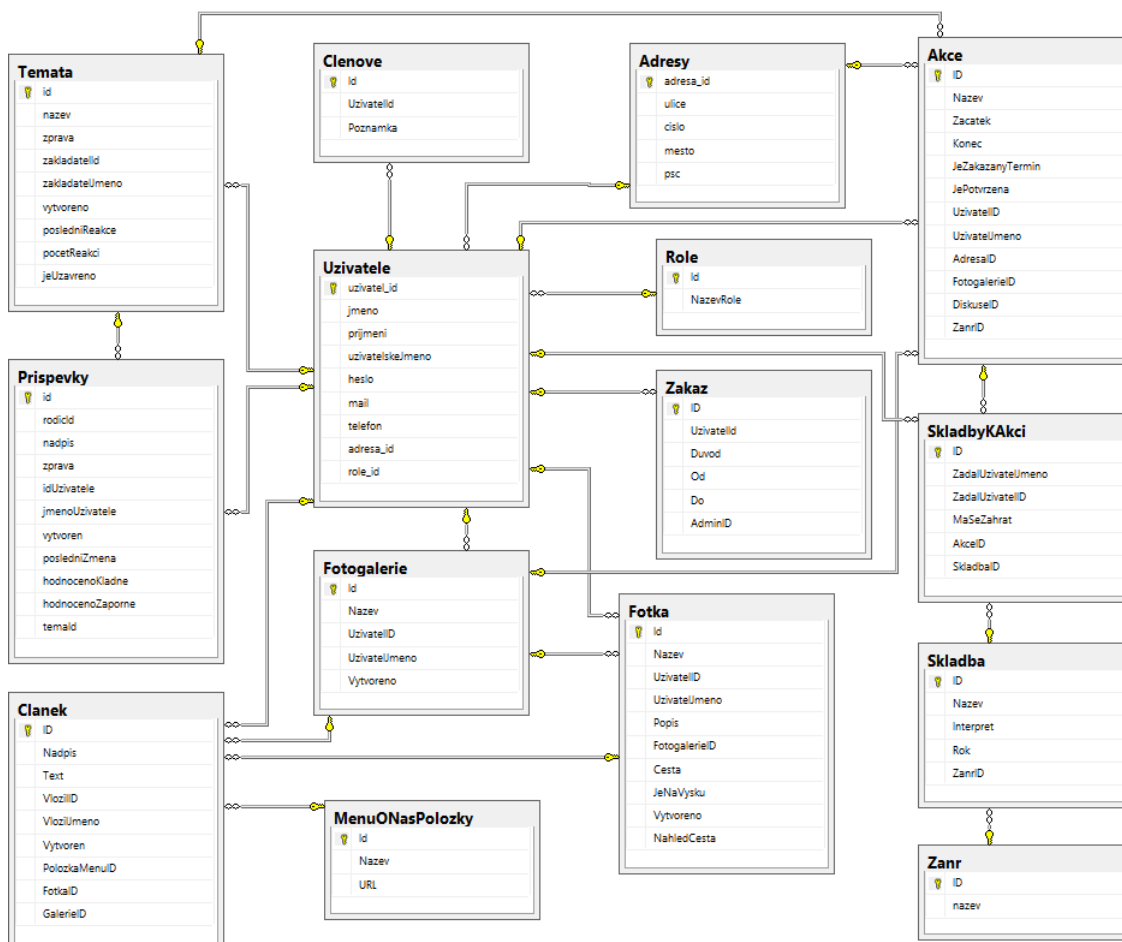
Providery je nutné před použitím zaregistrovat v konfiguračním souboru `Web.config` v kořenovém adresáři aplikace. To provedeme přidáním kódu 4 do sekce `<system.web>`.

```
1 <membership defaultProvider="MujMembershipProvider">
2   <providers>
3     <clear />
4     <add name="MujMembershipProvider" type="StoryMusic.Tridy.
5       Provider.MujMembershipProvider" />
6   </providers>
7 </membership>
8 <roleManager defaultProvider="MujRoleProvider" enabled="true">
9   <providers>
10    <clear />
11    <add name="MujRoleProvider" type="StoryMusic.Tridy.Provider.
12      MujRoleProvider" />
13  </providers>
14 </roleManager>
```

Zdrojový kód 4: Registrace providerů

4.5 Databáze

Datovou vrstvu aplikace tvoří databáze nazvaná „StoryMusicData.mdf“. Tato aplikace využívá Microsoft SQL 2012 Express (technologie byla stručně popsána v kapitole 2.3). Schéma databáze zachycuje obrázek 1.



Obrázek 1: Schéma databáze

Databáze obsahuje následující tabulky:

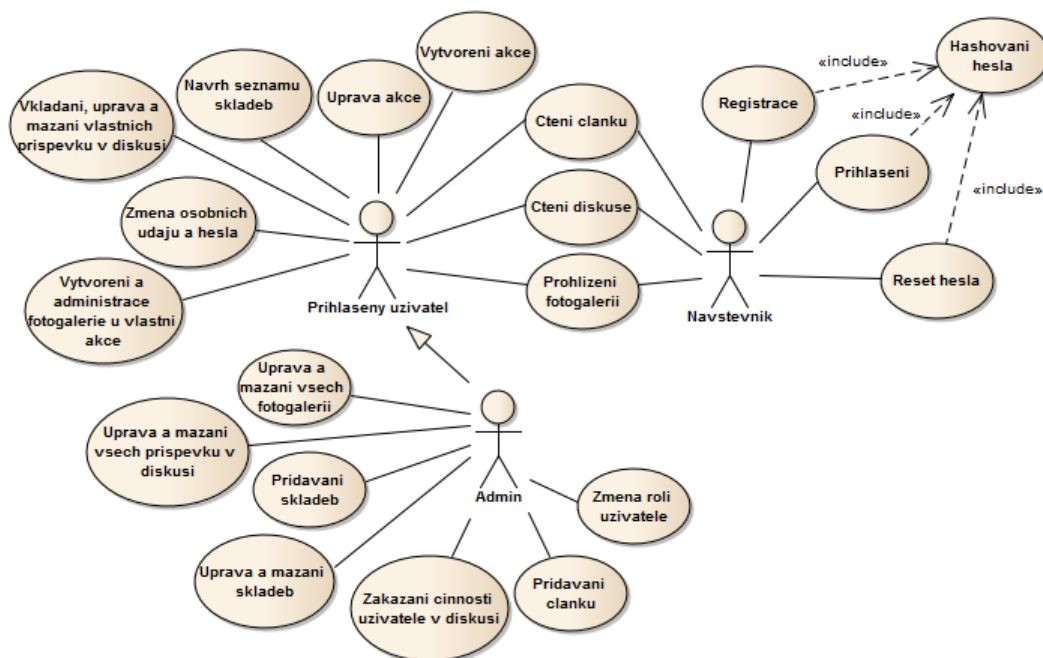
- Tabulka **Adresy** obsahuje adresy uživatelů a míst konání jednotlivých akcí.
- Tabulka **Akce** obsahuje informace o jednotlivých akcích v kalendáři (termín, místo konání, seznamy skladeb apod.).
- V tabulce **Clanek** jsou umístěny administrátorem přidané články (například Novinky).
- Tabulka **Clenove** přiřazuje členům skupiny doplňující údaje a umožňuje budoucí rozšíření zobrazovaných informací o členech skupiny.
- Tabulka **Fotka** obsahuje základní informace o nahraných fotografiích (název, popis, umístění apod.).
- Tabulka **Fotogalerie** obsahuje informace o fotogaleriích.
- V tabulce **MenuONasPolozky** je seznam položek menu, kam je možné přidávat články.

- Tabulka **Prispevky** obsahuje všechny příspěvky do diskuse.
- V tabulce **Role** je seznam všech uživatelských rolí používaných v aplikaci.
- Tabulka **Skladba** obsahuje informace o všech skladbách, ze kterých uživatelé tvoří seznamy pro akce (náze, interpret a jiné).
- Tabulka **SkladbaKActi** přiřazuje jednotlivé skladby ke konkrétním akcím v kalendáři.
- V tabulce **Temata** je seznam všech témat diskuse.
- Tabulka **Uzivatele** reprezentuje registrované uživatele webu.
- V tabulce **Zakaz** jsou informace o uživateli, kteří mají administrátorem zakázánu činnost v diskusi.
- Tabulka **Zanr** obsahuje seznam hudebních žánrů, které se přiřazují jednotlivým skladbám a akcím.

4.6 Funkcionalita aplikace

Funkcionalita této aplikace byla určena požadavky ze strany skupiny StoryMusic, nápady autora a návrhy a připomínkami vedoucího bakalářské práce.

Základní náhled do funkcionality nabízí obrázek 2, podrobnější popis pak následuje dále v textu, kde k jednotlivým užitím přiřadíme odpovídající metody.



Obrázek 2: Funkcionalita aplikace

1. Registrace

V `Controllers/UcetController.cs` je volána metoda `[HttpGet] Registrace`. Tato metoda volá pohled `Views/Ucet/Registrace.cshtml`. Po odeslání dojde k vyvolání `[HttpPost] Registrace`. Pokud je model validní, dojde k vytvoření účtu a odeslání informačního e-mailu.

2. Přihlášení

V `Controllers/UcetController.cs` je volána metoda `[HttpGet] Prihlaseni`. Tato metoda volá pohled `Views/Ucet/Prihlaseni.cshtml`. Po odeslání dojde k vyvolání `[HttpPost] Prihlaseni`. Pokud je model validní, dojde k přesměrování na úvodní stránku, jinak vrací pohled s chybou.

3. Reset hesla

V `Controllers/UcetController.cs` je volána metoda `[HttpGet] ZapomenuteHeslo`. Tato metoda volá pohled `Views/Ucet/ZapomenuteHeslo.cshtml`. Po odeslání dojde k vyvolání `[HttpPost] ZapomenuteHeslo`. Pokud je model validní, dojde k přesměrování na přihlašovací obrazovku a je odeslán e-mail s novými údaji.

4. Prohlížení fotogalerií

V `Controllers/FotogalerieController.cs` je volána metoda `Fotogalerie`. Tato metoda volá pohled `Views/Fotogalerie/Fotogalerie.cshtml`. Pro zobrazení každé fotogalerie je uvnitř pohledu navíc volán pohled `Views/Shared/_fotogalerieDlazdice.cshtml`.

5. Čtení diskuse

V `Controllers/DiskuseController.cs` je volána metoda `VsechnaTemata`. V příslušném pohledu je pak možnost po kliknutí na název diskuse zobrazit příspěvky (dojde k zavolání metody `ZobrazitTema` ve stejném kontroleru).

6. Čtení článků

V `Controllers/DomuController.cs` je volána metoda `ZobrazitClanky`. Ta podle předaného identifikátoru příslušné položky v menu vyhledá a předá pohledu všechny články v dané kategorii. Volaný pohled je `Views/Domu/-ZobrazitClanky.cshtml` a pro každý článek volá `Views/Domu/_clanekDlazdice.cshtml`.

7. Vytvoření akce

Po vybrání volného termínu v kalendáři akcí dojde k zavolání metody `[HttpGet] VytvoritAkci` v `Controllers/AkceController.cs`. Po vyplnění údajů k akci se volá `[HttpPost] VytvoritAkci`, dochází k validaci a kontrole dostupnosti termínu. Po úspěšné validaci dojde k vytvoření akce a odeslání informačních e-mailů administrátorovi a uživateli.

8. Úprava akce

Kliknutím na příslušný odkaz v detailu akce (který je dostupný pouze pro zakladatele akce a uživatele v roli admin) dojde k zavolání metody `[HttpGet] UpravitDetailAkce` v `Controllers/AkceController.cs`. Po změně údajů se volá `[HttpPost] UpravitDetailAkce`. Dále je situace obdobná jako u vytvoření akce.

9. Návrh seznamu skladeb

Kliknutím na příslušný odkaz v detailu akce (který je dostupný pouze přihlášeným uživatelům) dojde k zavolání metody `[HttpGet] ZobrazitSkladby` v `Controllers/SkladbyController.cs`. Uvnitř pohledu `Views/Skladby/ZobrazitSkladby.cshtml` dojde k zavolání metody `ZobrazitSkladbyKAkci`. Ze zobrazeného seznamu skladeb lze pomocí příslušných tlačítek přidávat, nebo odebírat skladby k akci.

10. Vkládání, úprava a mazání vlastních příspěvků v diskusi

Tuto funkcionalitu poskytuje `Controllers/DiskuseController.cs` prostřednictvím metod `ZalozitNoveTema`, `ZobrazitTema`, `Odpovedet`, `UpravitPrispevek`, `UpravitTema`, `OdstranitPrispevek` a `OdstranitTema`. Všechny metody volají stejně pojmenované pohledy. Dochází k ověření, zda má aktuální uživatel na danou akci právo a v případě odstranění, kde jde o nevratnou akci, je vyzván k potvrzení záměru.

11. Změna osobních údajů a hesla

Metody pro změnu osobních údajů i hesla najdeme v `Controllers/UcetController.cs`. Změna osobních údajů je možná po zavolání metody `[HttpGet] DetailUctu`, heslo pak `[HttpGet] ZmenaHesla`, kde dojde ke zobrazení příslušných pohledů.

12. Vytvoření a administrace fotogalerie u vlastní akce

Kliknutím na příslušný odkaz v detailu akce (který je dostupný pouze pro zakladatele akce a uživatele v roli admin) dojde k zavolání metody `VytvoritGaleriiUAkce` v `Controllers/AkceController.cs`. Vlastní úpravy pak probíhají v `Controllers/FotogalerieController.cs` voláním metod `UpravitDetailFotogalerie`, `PridatFotkyDoGalerie`, `UpravitDetailFotografie`, `OdstranitFotografii` a `OdstranitFotogalerii`. V případě odstranění je uživatel vyzván k potvrzení akce (jde o nevratnou akci). Pokud editaci, nebo odstranění provádí administrátor u „cizí“ galerie, je vyzván k udání důvodu a zakladateli galerie je odeslán informační e-mail se souhrnem změn a odůvodněním.

13. Přidávání skladeb

Dochází k zavolání `Controllers/SkladbyController.cs` a metody `PridatSkladby`. V metodě dojde k ověření role admin a je zavolán pohled `Views/Skladby/-PridatSkladby`. Tento pohled poskytuje možnost nahrát soubor se skladbami.

Pro dodržení potřebného formátu je zde uveden příklad vstupu. Vybráním souboru a odesláním se pomocí javascriptové funkce zavolá nejprve metodu `NahratSkladbyNaServer` v `Controllers/SouboryApiController.cs` a po úspěšném přidání souboru na server zavolá `ZpracovatSkladbyZExcelu` v `Controllers/SkladbyController.cs`, kde dojde ke zpracování souboru a uložení skladeb do databáze.

14. Úprava a mazání skladeb

Dojde k zavolání metody `[HttpGet] ZobrazitSkladby` v `Controllers/SkladbyController.cs`. Uvnitř pohledu `Views/Skladby/ZobrazitSkladby.cshtml` dojde k zavolání metody `ZobrazitSkladbyKUprave`. Ze zobrazeného seznamu skladeb lze pomocí příslušných tlačítek sklady odstraňovat, nebo upravovat.

15. Zákaz činnosti uživatele v diskusi

Role admin má u každého příspěvku v diskusi možnost zakázat činnost uživateli, který příspěvek vložil. `Controllers/DiskuseController.cs` zavolá metodu `[HttpGet] ZakazatCinnost` a dojde k zavolání příslušného pohledu. Po vyplnění údajů a odeslání formuláře je zavolána metoda `[HttpPost] ZakazatCinnost`.

16. Přidávání článků

`Controllers/AdministraceController.cs` volá metodu `[HttpGet] PridatClanek`. Tato metoda získá z databáze kategorie, do kterých je možné článek přidat. Kategorie zobrazí v pohledu `Views/Administrace/PridatClanek.cshtml` spolu s formulářem pro přidání článku. Ve formuláři je rovněž možnost přidat k článku fotogalerii, nebo fotku. Po odeslání formuláře požadavek převezme metoda `[HttpPost] PridatClanek`. Pokud byla přidána fotka, je zpracována a uložena do složky `Fotogalerie/Ostatni`.

17. Změna role uživatele

Tato funkcionality je dostupná pouze pro roli admin. Dojde k zavolání metody `[HttpGet] ZmenitRoliUzivatele`. Z databáze jsou načteny všechny uživatelské role a zobrazeny v pohledu `Views/Administrace/ZmenitRoliUzivatele.cshtml`. Po nastavení role a odeslání se volá metoda `[HttpPost] ZmenitRoliUzivatele` a nová role je uložena do databáze.

U výše zmíněných pohledů mnohdy dochází k editaci, nebo žádosti o potvrzení prostřednictvím javascriptových funkcí inspirovaných návody na těchto adresách [9].

4.7 Rozdělení uživatelského rozhraní

Uživatelské rozhraní aplikace je horizontálně rozděleno do tří logických částí:

- záhlaví dokumentu
- tělo dokumentu
- zápatí dokumentu

4.7.1 Záhlaví dokumentu

V záhlaví dokumentu je umístěna sekce pro přihlášení a hlavní menu aplikace.

- Přihlášení - sekce pro přihlášení uživatele je umístěna v pravé horní části záhlaví
 - při vstupu nepřihlášeného uživatele na stránku jsou v této části umístěny odkazy *Přihlášení* a *Registrace* uživatele.
 - po přihlášení uživatele do aplikace se odkazy v této sekci změní na uživatelské jméno aktuálně přihlášeného uživatele a *Odhlásit se*
- Hlavní menu
 - pro nepřihlášeného uživatele hlavní menu nabízí sekce *O nás*, *Fotogalerie* a *Diskuse*.
 - po přihlášení běžného uživatele se menu rozšíří o sekci *Akce* a v případě, že přihlášený uživatel je administrátor, tak navíc o sekci *Administrace*.

4.7.2 Tělo dokumentu

Tělo dokumentu je vertikálně rozděleno na dvě části:

- Uživatelské menu - tato část je viditelná pouze pro přihlášené uživatele. Nemění svůj obsah v závislosti na uživatelské roli a obsahuje následující položky:
 - *Upravit profil* - odkaz vede na formulář s možností změnit údaje uživatelského účtu zadané při registraci
 - *Moje Akce* - odkaz vede na stránku s výpisem akcí daného uživatele
 - *Moje fotogalerie* - odkaz vede na stránku s fotogaleriemi, které jsou založeny daným uživatelem, nebo je pro tohoto uživatele založil administrátor
 - *Moje diskuse* - vede na stránku s výpisem všech témat diskuse, které daný uživatel založil, nebo se zapojil do diskuse
- Obsahová část - je nejdynamičtější část uživatelského rozhraní aplikace. Zobrazuje data (vlastní obsah stránek) podle toho, jaký odkaz uživatel využil.

5 Uživatelská dokumentace

Tato sekce poskytuje základní návod na používání webu pro správu hudebních akcí skupiny Story Music.

5.1 Úvodní stránka

Úvodní stránka (obr. 3) je první, která se zobrazí po příchodu uživatele na tento web. Obsahuje základní informace o činnosti skupiny, důležité kontakty na její členy a stručný přehled o funkcích webu.

Uživatel se může na úvodní stránku kdykoliv vrátit kliknutím na logo STORY MUSIC DJs v horní části obrazovky, nebo odkaz *STORY MUSIC DJs* v zápatí stránky.



Obrázek 3: Úvodní strana

5.2 Registrace a přihlášení

Bez přihlášení do systému web umožňuje pouze prohlížení stránek. Pro zpřístupnění všech funkcí webu je nutné se nejprve přihlásit, pokud je uživatel již zaregistrován, nebo se zaregistrovat.

K tomu slouží odkazy *Přihlášení* a *Registrace* v pravém horním rohu.

Registrační formulář (obr. 4) ověří zda jsou zadané údaje kompletní a ve správném tvaru. Ověřuje se například tvar e-mailové adresy, zda se shodují hesla, tvar telefonního čísla a podobně.

The image shows a registration form on a dark blue background. The form is divided into three main sections: 'Osobní údaje', 'Kontakt', and 'Adresa'. Each section contains several input fields with pre-filled or entered values. Error messages are displayed in red text next to the fields. At the bottom of the form is a 'Registrovat' button.

Registrace		
Osobní údaje		
Jméno:	Inf	
Příjmení:	Upol	
Uživatelské jméno:	Infupol	
Heslo:	••••••	
Heslo znovu:	••••••	Zadané hesla se neshodují!
Kontakt		
e-mail:	e-mail@	E-mailová adresa nemá správný formát! (př: xxx@yyy.cz)
Telefon:	7	Špatný tvar čísla, očekává se 777 777 777, nebo 777777777, nebo +420 777 777 777!
Adresa		
Ulice:	17. listopadu	
Číslo popisné:	12	
Město:	Olomouc	
PSČ:	77146	
<input type="button" value="Registrovat"/>		

Obrázek 4: Registrační formulář

Po úspěšné registraci se může uživatel přihlásit. Přihlášení probíhá zadáním zvoleného uživatelského jména a hesla do přihlašovacího formuláře. Pokud si uživatel nepamatuje své heslo, může zažádat o reset hesla. K tomu slouží odkaz *Nepamatuji si heslo* ve spodní části přihlašovacího formuláře. Pro reset hesla je nutné zadat uživatelské jméno a registrační e-mail.

5.3 Diskuse

Nepřihlášený uživatel může diskusi pouze číst. Po přihlášení získá uživatel možnost vytvářet nová témata diskuse a odpovídat na existující příspěvky.

Témata diskuse je možné zobrazit kliknutím na odkaz *Diskuse* v hlavním menu. Témata jsou stránkována po pěti a řazena od nejnovějšího podle data posledního přidaného příspěvku.

- **Vytvoření nového tématu** - k tomuto účelu slouží tlačítko *Vytvořit nové téma*, které je umístěno vlevo pod tabulkou existujících témat. Zobrazí se formulář pro vytvoření tématu, kde uživatel zadá název tématu a text prvního příspěvku. Téma se vytvoří po kliknutí na tlačítko *Vytvořit* pod formulářem.
- **Zobrazení příspěvků v diskusi** - všechny příspěvky k danému tématu se zobrazí po kliknutí na název tématu ve sloupci *Téma* v tabulce všech témat.
- **Panel pro práci s příspěvkem** - každý příspěvek má pod sebou panel pro práci s příspěvkem. Položky v tomto panelu se liší podle role přihlášeného

uživatele a jestli příspěvek vložil uživatel sám, nebo někdo jiný. Panel je zachycen na obrázku 5.

Odpovědět | Upravit | Odstranit | Zakázat činnost (ban)

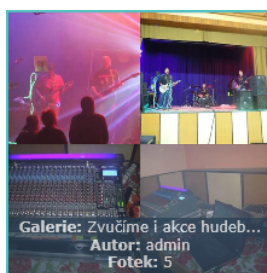
Obrázek 5: Panel pro práci s příspěvky v diskusi

Panel může obsahovat následující volby:

1. **Odpovědět** - slouží pro zadání odpovědi na daný příspěvek.
 2. **Upravit** - slouží k úpravě textu odpovědi. Každý uživatel má tuto volbu u vlastních příspěvků, administrátor u všech příspěvků.
 3. **Ostranit** - slouží k odstranění daného příspěvku a všech příspěvků, které vznikly jako reakce na tento odstraňovaný. Jde o nevratnou akci, která vyžaduje potvrzení. Každý uživatel má tuto volbu u vlastních příspěvků, administrátor u všech příspěvků.
 4. **Zakázat činnost (ban)** - tuto volbu má v panelu pouze administrátor a umožňuje zakázat činnost uživatele v diskusi na zadané období.
- **Úprava, odstranění tématu** - provádí se pomocí volby upravit, nebo odstranit u prvního příspěvku v tématu (zadaného při vytvoření tématu).

5.4 Fotogalerie

Všechny fotogalerie můžeme zobrazit kliknutím na odkaz *Fotogalerie* v hlavním menu. Prohlížení fotogalerií nevyžaduje přihlášení uživatele. Pro každou fotogalerii v aplikaci se zobrazí grafický prvek, který ji reprezentuje (obr 6).



Obrázek 6: Grafický prvek reprezentující fotogalerii

- **Zobrazení fotogalerie** - náhledy všech fotek v galerii zobrazíme kliknutím na příslušný grafický prvek fotogalerie. Pro další text tuto stránku nazveme „Náhled fotogalerie“.

- **Odstranění, úprava fotogalerie, přidání fotek** - tyto úkony může provádět pouze majitel fotogalerie, nebo administrátor. Pokud uživatel splňuje tuto podmínku, vidí na stránce „Náhled fotogalerie“ vlevo pod nadpisem ikony pro odstranění (křížek), editaci (papír s tužkou) a přidání fotek (plus).
 1. **Ostranit** - slouží k odstranění celé fotogalerie včetně fotek, které obsahuje. Jde o nevratnou akci, která vyžaduje potvrzení.
 2. **Upravit** - slouží k úpravě informací o fotogalerii.
 3. **Přidat fotky** - umožňuje nahrání více fotek najednou. Fotky uživatel vybere na svém lokálním disku pomocí tlačítka procházet a následně odešle pomocí tlačítka *Nahrát*.
- **Zobrazení a odstranění fotky, úprava informací o fotce** - po kliknutí na příslušnou zmenšeninu fotky na stránce „Náhled fotogalerie“ se zobrazí příslušná fotka v rozlišení 800x600px a informace o fotce. Šipkami po stránkách fotky je možné přepínat mezi fotkami v příslušné galerii. Pokud je aktuální uživatel administrátor, nebo majitel fotky, pak pod pravým dolním rohem fotky má k dispozici ikony pro úpravu informací o fotce a pro její odstranění.
- **Vytvoření nové fotogalerie** - každý uživatel má možnost přidat fotogalerii k vlastní potvrzené akci (jak vytvořit akci si popíšeme později) kliknutím na ikonu pro přidání v části fotogalerie. Pro přidání fotogalerie mimo akci je nutné požádat o vytvoření administrátora. K tomuto účelu slouží ikona pro přidání na stránce s výpisem všech fotogalerií (odkaz v hlavním menu).

5.5 Akce

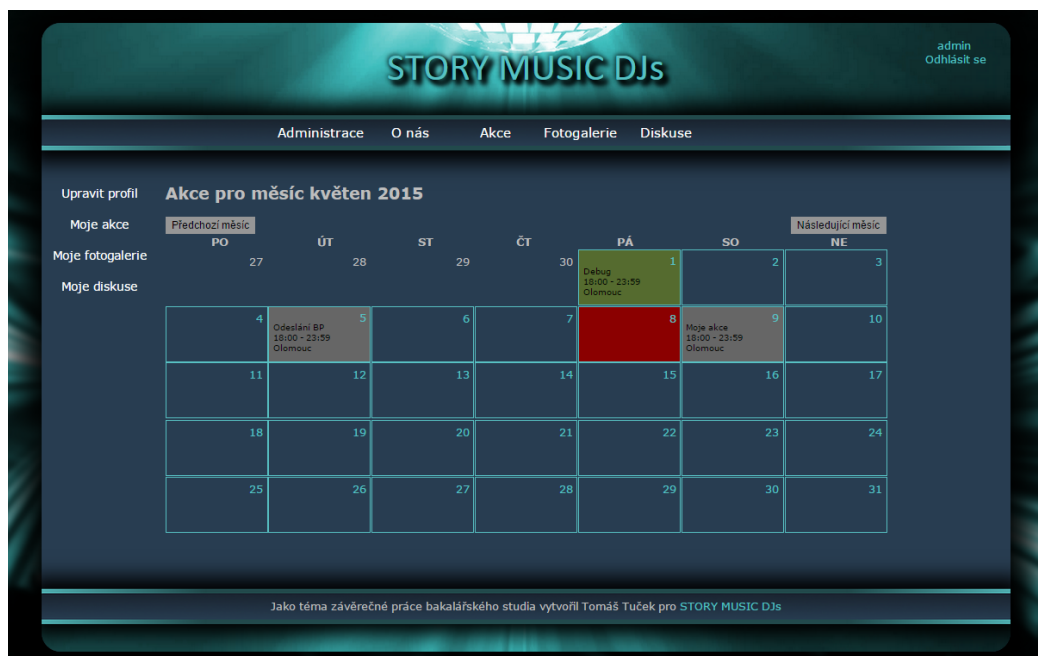
Sekce Akce slouží ke správě akcí skupiny. Je dostupná pouze přihlášeným uživatelům a obsahuje části *Kalendář akcí*, *Volné termíny* a pro administrátory ještě *Naplánované akce*.

Kalendář akcí

Tato část slouží ke grafickému znázornění akcí v daném kalendářním měsíci (viz obr. 7). Jednotlivé dny jsou reprezentovány grafickým prvkem, který mění svůj vzhled a obsah následovně:

- **Volný termín** - pole má průhledné pozadí a obsahuje pouze datum
- **Vlastní akce** - pole má zelenou barvu pozadí, obsahuje základní informace o akci
- **Cizí akce** - pole má šedou barvu pozadí, obsahuje základní informace o akci

- **Zakázaný termín** - pole má červenou barvu pozadí a označuje, že na daný den nejde naplánovat akce



Obrázek 7: Kalendář akcí

Volné teríny

Tuto část využije uživatel, který hledá vhodný termín pro vlastní akci a zájímají ho pouze volné termíny a ne celý kalendář. Umožňuje dva typy zobrazení volných termínů. Základní, kdy dny jsou zobrazeny postupně za sebou a kalendář, který je obdobný jako kalendář akcí, ale zobrazeny jsou pouze volné termíny.

Naplánované akce

Tato stránka je dostupná pouze pro administrátory a umožňuje zobrazení rezervovaných termínů. Obdobně jako sekce *Volné termíny* umožňuje dva typy zobrazení, základní a kalendář.

Vytvoření akce

Formulář pro vytvoření nové akce je dostupný po kliknutí na vybraný volný termín v kalendáři akcí, nebo na ikonu pro přidání akce v uživatelské sekci *Moje akce*. Po vyplnění údajů vložíme akci do systému tlačítkem *Vytvořit akci*.

Takto zadaná akce je pouze rezervace termínu, ne závazný termín akce. Aplikace při vytvoření odešle žádost administrátorovi o schválení termínu. Pokud administrátor tuto žádost potvrdí, uživatel obdrží potvrzující e-mail o potvrzení termínu.

Zobrazení akce

Stránka s detaily akce se zobrazí po kliknutí na obsazený termín v kalendáři akcí. Na stránce najdeme základní informace o samotné akci a pokud jde o potvrzený termín, pak může stránka obsahovat i diskusi, fotogalerii a navržené seznamy skladeb.

K rychlému zobrazení termínů vlastních akcí slouží sekce *Moje akce* v uživatelském menu, odkud k detailu akce přejdeme kliknutím na název akce v tabulce.

Úprava a odstranění akce

Tyto úkony může provést pouze zakladatel akce, nebo administrátor. Stejně, jako v případě fotogalerie, jsou k tomu určeny ikony vlevo pod nadpisem. Odstranění akce je nevratné a vyžaduje potvrzení.

Další možnosti úpravy akce, pokud je potvrzená:

- **Přidání fotogalerie** - každý uživatel může k vlastní akci přidat fotogalerii pomocí ikony pro přidání v části fotogalerie na stránce s detailem akce
- **Přidání diskuse** - každý uživatel může k vlastní akci přidat diskusi pomocí ikony pro přidání v části fotogalerie na stránce s detailem akce
- **Úprava seznamů skladeb** - každý uživatel může navrhopvat seznamy skladeb pro danou akci pomocí odkazu *Upravit skladby k akci*. Odkaz vede na stránku s výpisem všech skladeb, ze kterých je možné seznamy vytvářet. Stránka umožňuje filtraci skladeb podle žánru a vyhledávání zadaného textu ve jméně interpreta a v názvu skladby. Přidání zvolené skladby do příslušného seznamu provedeme kliknutím na tlačítko *CHCI zahrát*, nebo *NECHCI zahrát* u dané skladby. Odebrání ze seznamu provedeme kliknutím na tlačítko *Odebrat z akce*.
- **Odstranění diskuse a fotogalerie z akce** - odstranění diskuse a fotogalerie probíhá v příslušném detailu jak bylo popsáno v sekcích 5.3 a 5.4 tohoto textu.

5.6 Administrace

Sekce administrace slouží ke vkládání článků, přidávání a úpravám skladeb a poskytuje přehled uživatelů.

Přidání článku

Odkazem *Přidat článek* v sekci *Administrace* v hlavním menu se zobrazí formulář pro vytvoření článku. V tomto formuláři uživatel vyplní název a text článku. Dále formulář obsahuje výběr kategorií, kam je možné článek přidat. Přednastavená je kategorie Novinky.

Ke každému článku je možné přidat fotku, nebo fotogalerii (označením jednoho se zruší označení druhého). Při označení „Přidat fotku“ musí uživatel ještě vybrat soubor s fotkou. Článek pak vytvoří kliknutím na tlačítko *Vytvořit článek*.

Zobrazení, úprava a odstranění článku

Kategorie, kam je možné články přidat jsou dostupné v hlavním menu v sekci *O nás*. Jde o *Novinky*, *Naše vybavení* a *Nově hraje*. Kliknutím na příslušný odkaz zobrazíme všechny články v dané kategorii. Články nemusí být zobrazeny celé. Pro zobrazení plného znění je u každého článku odkaz *Zobrazit celý článek*, kterým přejdeme na detail tohoto článku.

Pro úpravu, nebo odstranění článku jsou v jeho detailu vlevo pod nadpisem umístěny příslušné ikony, které se zobrazí pouze administrátorovi.

Přidání skladeb

Pro přidání nových skladeb slouží odkaz *Nahrát skladby* v sekci *Administrace*. Po kliknutí na odkaz je zobrazen formulář pro přidání souboru se skladbami a příklad jak má daný soubor vypadat.

Úprava skladeb

K úpravě informací o jednotlivých skladbách slouží odkaz *Upravit skladby* v sekci *Administrace*. Po kliknutí na odkaz je zobrazena stránka s výpisem všech skladeb odpovídajících zadaným podmínkám pro vyhledání. Při prvním příchodu na stránku jsou vypsány všechny skladby stránkované po dvaceti. Jak již bylo naznačeno, tato stránka podporuje filtraci skladeb a to podle žánru a zadaného textu, který je vyhledáván ve jméně interpreta a v názvu skladby.

Každá skladba má ve sloupci „Úpravy“ ikony pro editaci a odstranění skladby.

Uživatelé

V sekci *Uživatelé* má administrátor k dispozici údaje o všech registrovaných uživateli. Pomocí odkazu *Změnit roli* ve sloupci „Role uživatele“ zde může změnit roli daného uživatele.

6 Nasazení

6.1 Minimální požadavky

Celý portál byl implementován v technologii ASP.NET MVC4, která pro běh aplikace využívá .NET Framework verze 4.5. Z tohoto důvodu musí být operační systém serveru Windows Server 2008 a novější[10], IIS 7.5 a novější a nainstalovaný .NET Framework 4.5 a novější.

Portál byl laděn pro aktuální verze rozšířených webových prohlížečů (Chrome, Internet Explorer, Mozilla Firefox, Opera a Safari).

Současná verze byla vyvíjena a laděna na konfiguraci: Windows 8.1 (64bit), IIS 8.5, .NET Framework 4.5.1, MS SQL Server 2012 Express (64bit verze včetně vývojářských nástrojů, „Stand-Alone“ instalace).

6.2 Přidání aplikace do IIS

Nejprve je nutné v adresáři *C:/inetpub/wwwroot* vytvořit adresář s názvem aplikace, do kterého nahrajeme obsah adresáře **bin** na přiloženém CD. Vytvořenému adresáři je nutné přidat práva pro zápis uživateli IIS_IUSRS.

Spustíme IIS Manager (zadáním příkazu „inetmgr“ do příkazového řádku Windows) a najdeme v adresářové struktuře přidáný adresář s aplikací. Pravým tlačítkem myši klikneme na adresář a zvolíme „Převést na aplikaci“.

6.3 Databáze

V adresáři **database** na přiloženém CD v souboru s názvem **dbScript.sql** jsou k dispozici exportovaná schémata tabulek a nezbytná data.

Součástí instalace MS SQL Server (ve verzi s nástroji) je nástroj **SQL Server Management Studio**, přes který je možné po přihlášení k databázi výše zmíněné definice databázových tabulek včetně nezbytných dat importovat.

6.4 Konfigurace

Pro správnou funkci aplikace je nutné v souboru Web.config zkontrolovat následující:

- **Connection String** - v sekci „configuration/connectionStrings“ musí být záznam s názvem „StoryMusicConStr“ a platnou hodnotou „connectionString“ pro aktuální databázi.
- **Nastavení e-mailu** - „configuration/appSettings“ musí být vyplněny následující údaje pro odesílání e-mailu:
 - **MailOdesilatel** - aktuální adresa uživatele webu
 - **MailUzivatel** - aktuální adresa uživatele webu

- **MailHeslo** - platné heslo k zadanému e-mailu
- **SMTPHost** - adresa zprostředkovatele (např smtp.gmail.com)
- **SMTPPort** - platný port pro komunikaci

7 Možnosti dalšího vývoje aplikace

Na základě připomínek členů skupiny, zpětné vazby od uživatelů, nebo nápadů ze strany autora je možné webový portál v budoucnu rozšířit nebo upravit například následovně:

- přidat více termínů akce na jeden den, v současné verzi je možnost rezervace pouze jedné akce na den z důvodu, že členové skupiny pořádají akce hlavně jako hobby při zaměstnání a více není z jejich strany možné
- umožnit jednorázovou rezervaci termínu akce bez nutnosti registrace
- vytvořit anketu mezi uživateli a na jejich připomínkách zpracovat jiný vzhled diskuse a dalších částí webu
- přidat možnost převodu akcí v kalendáři do formátu pdf pro tisk
- vypracvat jednoduchý editor fotografií přímo v portálu, který by umožňoval otočení fotografie a její oříznutí, případně ještě změnu jasu a kontrastu
- umožnit výběr více fotografií pro editaci, nebo odstranění najednou
- po přihlášení zobrazit nejbližší akci; pro uživatele vlastní, pro administrátora nejbližší potvrzenou
- přidat vyhledávání

Závěr

V rámci této bakalářské práce byla vytvořena webová aplikace pro správu hudebních akcí skupiny Story Music.

Tato aplikace rozlišuje několik uživatelských rolí, obsahuje kalendář akcí, diskusi a fotogalerii.

Pro přihlášeného uživatele umožňuje jednoduchou rezervaci termínu akce s automatickou kontrolou dostupnosti termínu a také filtraci volných termínů. U založené akce má zakladatel umožněno vytvořit pro danou akci diskusi a fotogalerii a také editovat údaje o akci. Každá akce může obsahovat dva seznamy skladeb - ty, které si uživatel přeje slyšet a ty, které si slyšet nepřeje.

Pro administrátora umožňuje filtraci obsazených termínů pro snadnou orientaci v naplánovaných akcích. Administrátor může zakázat činnost uživatele v diskusi a editovat příspěvky v diskusi. Může vytvářet, editovat a mazat fotogalerie a fotky v nich. Může editovat seznam skladeb, ze kterých si uživatelé vybírají skladby pro své akce. Může editovat údaje o akcích. Při zásahu administrátora do dat uživatele aplikace automaticky odesílá uživateli e-mail se souhrnem změn a odůvodněním, proč k nim došlo.

Conclusions

The web portal for disco event management was made during this thesis. Several user roles are distinguished by this portal. Portal also includes calendar, discussion and photo gallery.

For registered user portal allows reservation of free date, event management and free date filtering. Photogallery and discussion could be added to event and there are two lists of songs in the event - first includes songs which user want to be played, second which user do not want to be played.

Also portal provides booked date filtration, full action, discussion and photogallery management for administrator.

Portal sends e-mail to user if user data are changed by administrator.

Literatura

- [1] Elektronická publikace. <http://cs.wikipedia.org/wiki/ASP.NET>.
- [2] Adam FREEMAN and Steven SANDERSON. *Pro ASP.NET MVC 4*. Springer, 2012.
- [3] Elektronická publikace. http://cs.wikipedia.org/wiki/Microsoft_SQL_Server.
- [4] Elektronická publikace. <http://cs.wikipedia.org/wiki/GIMP>.
- [5] Microsoft. Oficiální web technologie asp.net. <http://www.asp.net>.
- [6] Jon GALLOWAY, Phil HAACK, Brad WILSON, and K Scott ALLEN. *Professional ASP. NET MVC 4*. John Wiley & Sons, 2012.
- [7] Microsoft. Rekurzivní dotazy. [https://technet.microsoft.com/en-us/library/ms186243\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms186243(v=sql.105).aspx).
- [8] Monjurul Habib. Předávání dat pomocí viewbag, viewdata a TempData. <http://www.codeproject.com/Articles/476967/WhatplusisplusViewData-2cplusViewBagplusandplusTem>.
- [9] Elektronická publikace - návody javascript alert, prompt a confirm.
alert(<https://www.developphp.com/video/JavaScript/Custom-Alert-Box-Programming-Tutorial>),
prompt(<https://www.developphp.com/video/JavaScript/Custom-Prompt-Box-Programming-Tutorial>),
confirm(<https://www.developphp.com/video/JavaScript/Custom-Confirm-Box-Programming-Tutorial>).
- [10] Elektronická publikace. [https://msdn.microsoft.com/cs-cz/library/8z6watww\(v=vs.110\).aspx](https://msdn.microsoft.com/cs-cz/library/8z6watww(v=vs.110).aspx).
- [11] Adam JORGENSEN, Steven WORT, Ross LOFORTE, and Brian KNIGHT. *Professional Microsoft SQL Server 2012 Administration*. John Wiley & Sons, 2012.
- [12] Microsoft. Developer network. <https://msdn.microsoft.com/en-us>.

A Obsah příloženého CD

Na samotném konci textu práce je uveden stručný popis obsahu příloženého CD/DVD, tj. jeho závazné adresářové struktury, důležitých souborů apod.

bin/

Kompletní adresářová struktura webové aplikace STORYMUSIC pro zkopírování na webový server. Adresář obsahuje i všechny runtime knihovny a další soubory potřebné pro bezproblémový provoz webové aplikace na webovém serveru.

database/dbScript.sql

Export tabulek a nezbytných dat.

doc/

Text práce ve formátu PDF, vytvořený s použitím závazného stylu KI PřF UP v Olomouci pro závěrečné práce, včetně všech příloh, a všechny soubory potřebné pro bezproblémové vygenerování PDF dokumentu textu (v ZIP archivu), tj. zdrojový text textu, vložené obrázky, apod.

src/

Kompletní zdrojové texty webové aplikace STORYMUSIC se všemi potřebnými (příp. převzatými) zdrojovými texty, knihovnami a dalšími soubory potřebnými pro bezproblémové vytvoření adresářové struktury pro zkopírování na webový server.

readme.txt

Instrukce pro nasazení webové aplikace STORYMUSIC na webový server, včetně všech požadavků pro její bezproblémový provoz, a webová adresa, na které je aplikace nasazena pro účel testování při tvorbě posudků práce a pro účel obhajoby práce.