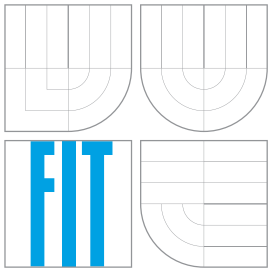


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

COUCHSURFING PRO HUDEBNÍKY

COUCHSURFING FOR MUSICIANS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

VIOLA KLEINOVÁ

VEDOUcí PRÁCE
SUPERVISOR

Prof. Ing. ADAM HEROUT, Ph.D.

BRNO 2016

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2015/2016

Zadání bakalářské práce

Řešitel: **Kleinová Viola**

Obor: Informační technologie

Téma: **Couchsurfing pro hudebníky**
Couchsurfing for Musicians

Kategorie: Uživatelská rozhraní

Pokyny:

1. Prostudujte problematiku tvorby webových uživatelských rozhraní a dostupné nástroje pro efektivní vývoj aplikací.
2. Navrhněte "minimal viable product" verzi aplikace, která nezávislým muzikantům poskytne podobné služby, jak aplikace couchsurfing poskytuje cestovatelům.
3. Implementujte navržené uživatelské rozhraní.
4. Na základě zpětné vazby uživatelů a statistik o používání aplikace vylepšujte aplikaci.
5. Zhodnoťte dosažené výsledky a navrhněte možnosti pokračování projektu; vytvořte plakátek a krátké video pro prezentování projektu.

Literatura:

- dle pokynů vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2,
- značné rozpracování bodů 3 a 4.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Herout Adam, doc. Ing., Ph.D.**, UPGM FIT VUT

Datum zadání: 1. listopadu 2015

Datum odevzdání: 18. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
602 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Cielom tejto práce, je navrhnuť a vytvoriť užívateľské rozhranie pre projekt podobný Couchsurfingu so zameraním na hudobníkov. Súčasťou práce je rozbor existujúcich riešení s podobným účelom, rozbor problematiky webových užívateľských rozhraní, návrh produktu, výber technológií a čiastočná implementácia produktu, vrátane spätnej väzby od testujúcich užívateľov.

Abstract

The goal of this work, is to design and create an user interface for project similar to Couchsurfing with special interest in musicians. A part of this work, is analysis of existing solutions with similar purpose, discussion of common issues with web user interfaces, blueprint of such product, selection comparison of available technologies and partial implementation of product itself, including feedback from testing sample of consumers.

Kľúčové slová

užívateľské rozhranie, Symfony framework, PHP, Bootstrap, web

Keywords

user interface, Symfony framework, PHP, Bootstrap, web

Citácia

KLEINOVÁ, Viola. *Couchsurfing pro hudebníky*. Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Herout Adam.

Couchsurfing pro hudebníky

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracovala samostatne pod vedením pána Adama Herouta. Uviedla som všetky literárne pramene a publikácie, z ktorých som čerpala.

.....

Viola Kleinová
18. mája 2016

Podakovanie

Ďakujem svojmu vedúcemu bakalárskej práce prof. Ing. Adamovi Heroutovi, Ph.D. za jeho čas, odborný dohľad, konzultácie a vedenie. Taktiež ďakujem všetkým najbližším ľuďom v mojom okolí za podporu a akúkoľvek formu pomoci.

© Viola Kleinová, 2016.

Táto práca vznikla ako školské dielo na FIT VUT v Brně. Práca je chránená autorským zákonom a jej využitie bez poskytnutia oprávnenia autorom je nezákonné, s výnimkou zákonne definovaných prípadov.

Obsah

1 Úvod	2
2 Prehľad existujúcich riešení a návrh požiadaviek webovej aplikácie	3
2.1 Existujúce riešenia podobných webových aplikácií	3
2.2 Princípy tvorby užívateľského rozhrania	7
3 Návrh požiadaviek funkčnosti webovej aplikácie	8
4 Popis výberu webových technológií použitých pri vývoji projektu	9
4.1 Výber jazyka	9
4.2 Výber frameworku	10
4.3 Použité technológie	11
5 Návrh a implementácia projektu Couchsurfing pre hudobníkov	14
5.1 Lokálne vývojové prostredie	14
5.2 Cloud	14
5.3 Návrh užívateľského rozhrania	14
5.3.1 Bootstrap	16
5.4 Výber názvu pre projekt	17
5.5 Logická štruktúra projektu	17
5.6 Návrh a implementácia databázy	18
5.7 Implementácia prihlasovania užívateľov	18
5.8 Implementácia užívateľského rozhrania	20
6 Testovanie a návrh na rozšírenie projektu	25
6.1 Návrh rozšírenia projektu	28
7 Záver	29
Literatúra	30
Prílohy	32
Zoznam príloh	33
A Manual a obsah CD	34

Kapitola 1

Úvod

Cieľom tejto práce je vytvoriť užívateľské rozhranie webovej aplikácie podobnej *Couchsurfingu*, ktorý je používaný cestovateľmi z rôznych svetových lokalít. Cieľovou skupinou zadaného projektu – *Couchsurfing pre hudobníkov* budú hudobníci a kapely vo všetkých vekových kategóriách. Pomocou tejto webovej aplikácie sa budú môcť nájsť, získať na seba kontakt, poskytnúť si navzájom ubytovanie a pomôcť si zorganizovať koncerty v iných (aj zahraničných) lokalitách. Táto stránka taktiež môže slúžiť ako webová prezentácia kapiel a hudobníkov, keďže mnoho kapiel má často nekvalitné oficiálne webové stránky, a hudobníci ako jednotlivci často oficiálne webové stránky nemajú vôbec.

Nápad pre vypracovanie tejto práce vznikol vďaka ľuďom v mojom okolí, účasti na hudobných projektoch a spoznania hudobníkov zo zahraničia. Veľké množstvo týchto hudobníkov a skupín by sa rado presadilo, a koncertovalo v iných krajinách. Je to však pre nich náročné, pretože im chýbajú vhodné kontakty a často disponujú nízkym finančným rozpočtom.

V tejto práci si opíšeme postup pri tvorbe užívateľského rozhrania zadanej webovej aplikácie. Na to, aby sme mohli navrhnúť potrebné funkčné požiadavky, a užívateľské rozhranie aplikácie, najskôr preskúmame existujúce projekty s podobným zameraním a preštudujeme problematiku tvorby užívateľských rozhraní. Potom vyberieme vhodné technológie pre vývoj tejto aplikácie a bližšie sa pozrieme na návrh, a implementáciu užívateľského rozhrania aplikácie. V poslednej časti práce si opíšeme testovanie webovej aplikácie a pozrieme sa na navrhnuté vylepšenia, a jej prípadné rozšírenia.

Kapitola 2

Prehľad existujúcich riešení a návrh požiadaviek webovej aplikácie

V tejto kapitole si opíšeme existujúce riešenia podobné webovej stránke, pre ktorú je potrebné vytvoriť užívateľské rozhranie. Opíšeme si účel každej z aplikácií, ich funkčnosť a zameriame sa na menší rozbor užívateľského rozhrania z vizuálneho hľadiska. Potom sa pozrieme na zoznam požiadaviek na funkčnosť vytváranej aplikácie a v poslednej časti sa budeme venovať princípom tvorby užívateľského rozhrania.

2.1 Existujúce riešenia podobných webových aplikácií

Na internete je možné nájsť mnoho sociálnych sietí a stránok, ktorých cieľovou skupinou sú hudobníci a ich fanúšikovia, avšak každý z daných projektov má iný prvotný zámer. Nižšie si opíšeme štyri rôzne portály. Získame tým prehľad toho čo by náš projekt mal obsahovať. Pri prieskume týchto webových stránok sa zameriame na prvotný zámer, návštevnosť, logiku a kvalitu užívateľského rozhrania.

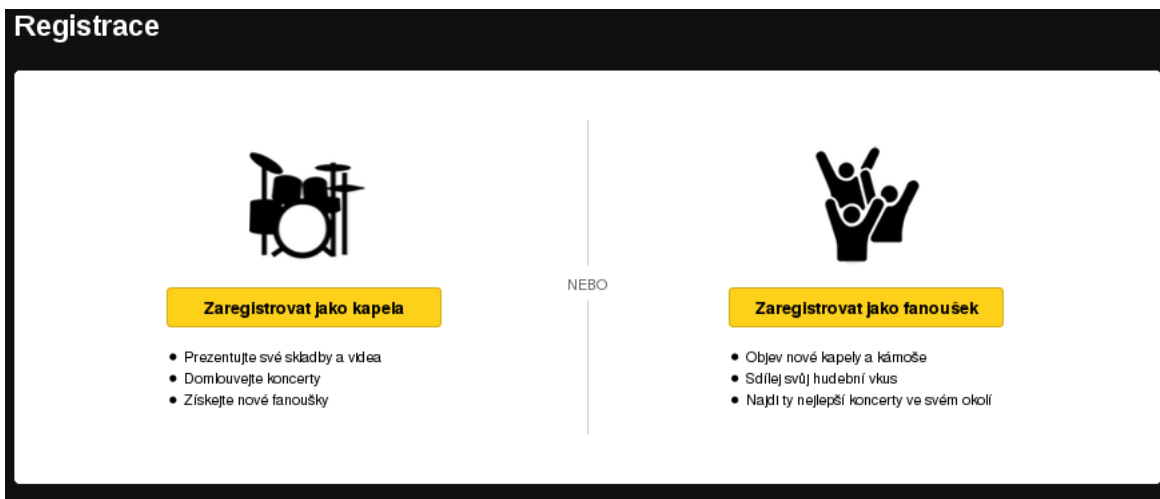
Couchsurfing je internetová služba, sociálna sieť pomocou ktorej si užívatelia môžu nájsť alebo ponúknuť bezplatné ubytovanie. Táto služba je veľmi obľúbená a využívaná najmä milovníkmi cestovania vo všetkých lokalitách. Každý rok využije túto službu približne 4 mil. ľudí [19] a komunitu tvorí viac ako 10 mil. užívateľov [20]. Návštevník bez prihlásenia má možnosť vidieť iba hlavnú stránku a pravidlá používania. Až po prihlásení užívateľ môže využívať plný potenciál aplikácie. Užívatelia sa môžu pridať do rôznych skupín, tvoriť obojstranné priateľstvá, písať referencie iným užívateľom, ponúknuť ubytovanie alebo hľadať ubytovanie vo vybranej lokalite. Taktiež stránka ponúka možnosť vytvárania rôznych udalostí. Menu je trochu netradičné, nachádza sa vo vrchnej časti stránky a funguje ako vyhľadávanie obr. 2.1. Profil užívateľa obsahuje množstvo osobných údajov, fotiek a referencií od priateľov, hostí alebo hostiteľov daného užívateľa. K týmto údajom majú prístup všetci prihlásení užívatelia.

The screenshot shows the couchsurfing.com interface. At the top, the logo is on the left, and navigation links for 'Events', 'Groups', and a user profile are on the right. A search bar contains 'Brno, Czech Republic' with a dropdown menu open, listing options: 'Explore', 'Find Hosts', 'Find Members', 'Find Travelers' (highlighted), 'Find Events', 'Find Groups', and 'Find Local Discussions'. Below the search bar, the main content area shows '21 Travelers Found near Brno, Czech Republic'. Two profiles are visible: Merlijn Hulzenga (from Netherlands, 5 references, speaks Dutch and English) and Laurent Fontaine (from France, speaks French and English). To the right, a 'Search Filters' sidebar includes a 'Search' button, 'RADIUS' set to 'within 10 km', 'DATE RANGE' with 'Arrival' and 'Departure' date inputs, and 'TRAVELER INFO' with checkboxes for 'Has references' and 'Verified', and a 'Gender' dropdown set to 'All'.

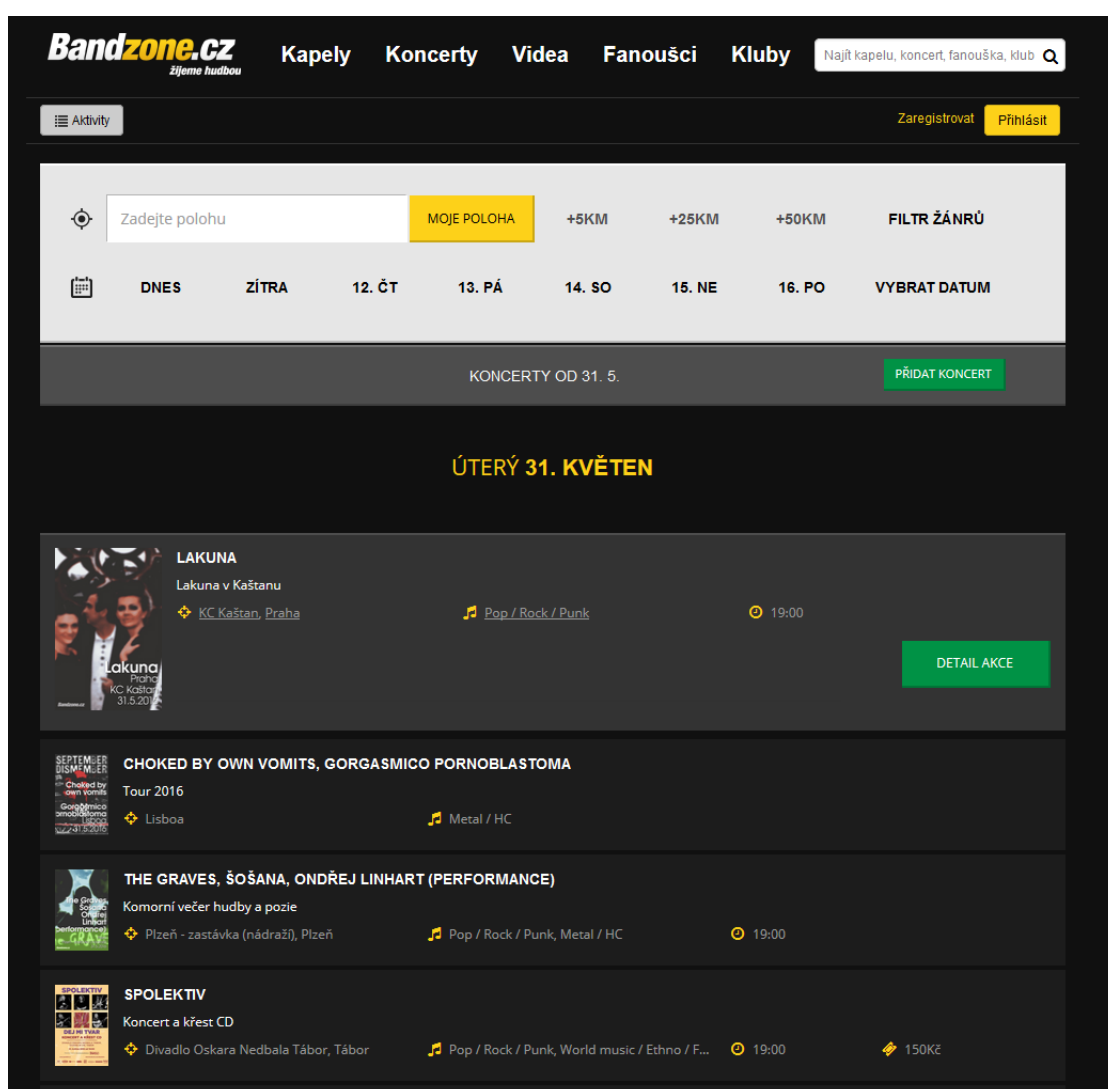
Obr. 2.1: Stránka s vyhledanými cestovateľmi na *couchsurfing.com*

Facebook je jedna z najväčších a najpoužívanejších webových sociálnych sietí na svete. Podľa oficiálnych údajov k marcu 2016 má facebook viac ako 1.6 miliardy aktívnych užívateľov [14]. Užívateľia tu majú možnosť uzatvárať obojstranné priateľstvá, zdieľať názory, multimédiá, fotografie, komunikovať spolu, sledovať iné osoby, pridať sa do skupín, vytvárať rôzne udalosti. Užívateľ si nastavuje, kto môže vidieť jeho údaje. Pomocou *Facebooku* je ťažšie získať kontakty neznámych ľudí, kapiel, hudobníkov a podobne, pretože vyhľadávanie je tu možné hlavne podľa názvu.

Bandzone je podľa oficiálnych stránok [1] najrozsiahlejšiou komunitou kapiel a ich fanúšikov na českom internete. Má viac ako 40 000 registrovaných skupín a skoro 200 000 registrovaných fanúšikov. Hlavným účelom *Bandzone* je prezentácia kapiel. Prihlasovanie do aplikácie je určené najmä pre kapely a ich fanúšikov, užívatelia definujú, do ktorej skupiny patria pri registrácii. Registrácia je teda rozdelená do dvoch skupín obr. 2.2 a je otáznou či je to najlepšie riešenie. Návštevníci bez prihlásenia môžu vidieť zoznam kapiel, fanúšikov, koncertov, profily kapiel, videá. Vyhľadávanie kapiel je možné podľa názvu, mesta, žánru alebo kraja. Taktiež je tu možné použiť filter ktorý vám zobrazí nové, populárne kapely a iné. Jedným z nedostatkov grafického rozhrania je, že nie je nijak označená časť menu, v ktorom sa práve nachádzame.



Obr. 2.2: Stránka s rozdělenou registrací na portáli *Bandzone.cz*



Obr. 2.3: Stránka s vyhledáváním koncertů na portáli *Bandzone.cz*

Na obr. 2.3 môžeme vidieť stránku s vyhľadávaním koncertov. Na vyhľadávanie je možné využiť zobrazený filter, ktorý nie je graficky najkrajšie spracovaný, avšak jeho funkčnosť je rozsiahla a efektívna. Po kliknutí na jeden výsledok vyhľadávania sa nám zobrazí viac informácií o koncerte a po kliknutí na tlačidlo **DETAIL AKCE** budeme presmerovaní na samostatnú stránku s koncertom. Profil kapely obr. 2.4 je celkom dobre spracovaný. Je vizuálne rozdelený na viacero častí ako fotografia a názov kapely, hudobné ukážky, informácie o kapele, fanúšikovia a spriatelnené kapely. Kontakt na profile kapiel je chránený pred botmi tým, že na jeho zobrazenie je potrebné zadať zobrazený kód.

Obr. 2.4: Profil kapely na portáli *Bandzone.cz*

Trashkimono má najpodobnejší účel ako projekt, ktorého návrh je cieľom tejto práce. Je to webová stránka, kde si kapely môžu vytvoriť profily za účelom tzv. swap gigs¹, ďalej vidieť zoznam kapiel rozdelený podľa hlavných žánrov avšak bez možnosti vyhľadávania. Taktiež stránka kapelám umožňuje vytvoriť nový gig, ale nie je tu možnosť kontaktovania ostatných kapiel, a nie je viditeľný ani kontakt (napríklad. e-mail) na kapelu. Vo svojom profile majú kapely možnosť prezentovať sa krátkym popisom kapely, fotografiou, odkazmi na

¹swap gigs – spriatelnenie a zorganizovanie výmeny alebo spoločných koncertov kapiel z rôznych lokalít.

nahrávky na SoundCloude, odkaz na Facebook, Bandcamp a ich oficiálnu webovú stránku. Na profile sú ďalej vidieť swap gigs danej kapely. Podľa oficiálnych údajov [21]. *Trashkimono* aplikáciu vytvára český developer. Vďaka obmedzeným možnostiam použitia, nedostatku informácií na oficiálnych stránkach a dlhodobému nízkemu počtu užívateľov, môžeme tvrdiť, že projekt nie je dobre vypracovaný.

2.2 Princípy tvorby užívateľského rozhrania

Vizuálna podoba a architektúra užívateľského rozhrania rozhoduje o jednoduchosti, obľúbenosti a pohodlí používania aplikácie pre užívateľov. Hlavná stránka by mala zaujať, keďže je to prvé miesto ktoré užívateľ uvidí. Často to rozhodne o tom či užívateľ na stránke zostane alebo odíde. Z domovskej stránky by mal byť jasný cieľ webovej aplikácie a taktiež by sa tu mal nachádzať hlavný výber (menu). Užívateľské rozhranie by malo byť prehľadné, intuitívne, nenútené, efektívne a jednoducho ovládateľné s jasnými inštrukciami. U dobre vytvoreného užívateľského rozhrania by nemalo dochádzať k rozptyľovaniu užívateľa nepotrebnými prvkami. Je vhodné, aby dôležitejší obsah mal na stránke lepšiu pozíciu a väčší priestor ako ostatné prvky [22]. Pozrieme sa ešte na zopár ďalších bodov od *GoodUI ideas* [11], ktoré je doporučené dodržiavať pri tvorbe užívateľského rozhrania:

- združenie podobných funkcií (informácií) do jednej časti namiesto fragmentácie užívateľského rozhrania
- vizuálne odlíšiť časti, na ktoré je možné kliknúť od ostatných častí / vizuálne odlíšiť časť menu, v ktorej sa užívateľ práve nachádza
- jasný popis funkčnosti tlačidiel – limitovať možnosť nerozhodnosti užívateľa vďaka nejasnému popisu funkčnosti
- dlhé formuláre rozdeliť na viacero kratších
- v niektorých prípadoch je lepšie konvenčné riešenie, ako vymýšľanie nového riešenia (napríklad ikonky označujúce možnosť zmazania, úpravy a podobne)
- malé prvky, na ktoré je možné kliknúť nahradiť väčšími a viditeľnejšími
- použitie ikoniek vždy s popisom
- použitie väčšieho paddingu u prvkov / zabránenie nahustenia prvkov veľmi blízko pri sebe
- vizuálne viditeľné hranice prvkov

Kapitola 3

Návrh požiadaviek funkčnosti webovej aplikácie

Aby sme mohli vybrať vhodné technológie pre vývoj webovej aplikácie, je dobré najskôr vedieť, čo by aplikácia mala obsahovať a akému účelu slúžiť. Po analýze existujúcich riešení, som sa teda zamerala na spísanie požiadaviek funkčnosti aplikácie. Tu prikladám zoznam požiadavok ktoré som vytvorila:

1. umožniť vytvoriť profily hudobníkom (jednotlivcom), ako aj kapelám a hudobným telesám
2. vyhľadávanie členov, kapiel
3. nájdenie hostiteľa v danej lokalite alebo ponúknutie ubytovania
4. kontaktovanie členov alebo možnosť získania kontaktu členov
5. vytvorenie udalostí – koncertov, workshopov...
6. hľadanie nových spoluhráčov / členov do kapely
7. „bazár“, predaj / kúpa hudobných nástrojov a techniky z druhej ruky
8. umožniť vytvorenie profilov pre miesta (kaviarne, koncertné miesta, hudobniny a podobné) a hudobné školy

Jedným z cieľov tohto projektu je navrhnuť „minimal viable product“, takže pre dosiahnutie podobnej funkcionality ako u Couchsurfingu sa zameriame najmä na body 1 – 4.

Kapitola 4

Popis výberu webových technológií použitých pri vývoji projektu

Keďže sa navrhoval a vytváral úplne nový projekt, bolo potrebné porovnať a vybrať vhodné technológie. V tejto kapitole je opísaný postup výberu technológií a ich porovnanie. Hlavnými požiadavkami na výber bolo, aby použité technológie fungovali na princípe server-side, popularita (rozšírenosť), multiplatformita technológií a možnosť použitia objektového návrhu.

4.1 Výber jazyka

Jazykov používaných na vývoj webových aplikácií je veľké množstvo a z tohto dôvodu bolo nutné vybrať užšiu skupinu. Zamerala som sa teda na najpoužívanejšie programovacie jazyky, pretože u menej rozšírených alebo nových nastáva riziko zlej alebo zložitej kompatibility s inými technológiami, straty podpory alebo zániku daného jazyku.

Python je vysokoúrovňový viacúčelový interpretovaný skriptovací jazyk s dynamickou typovou kontrolou. Pre niektorých výhodou, pre iných nevýhodou sú jeho prísne pravidlá písania kódu. Vďaka nim je však kód čistejší a prehľadnejší ako u iných jazykov. Ďalšou výhodou je, že je rýchlejší ako jazyk PHP. Pre vývoj webových aplikácií slúži developerom napr. framework *Django* alebo *Flask*.

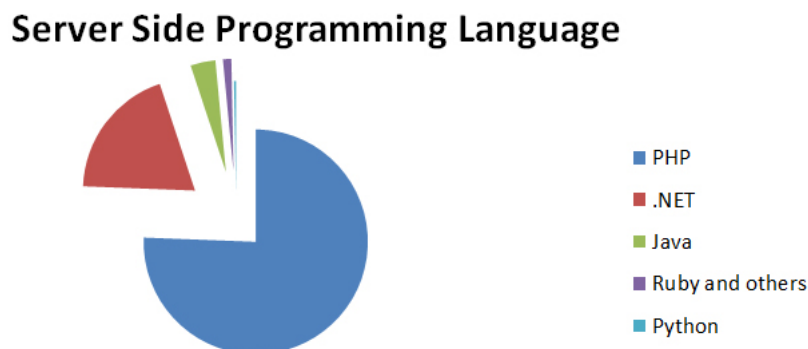
Ruby je veľmi podobný jazyku Python. Je to plne objektovo orientovaný interpretovaný skriptovací programovací jazyk. Ruby je pomerne starý jazyk (1995), ktorého popularita stúpila najmä vďaka príchodu frameworku *Ruby on Rails* pre vývoj webových aplikácií.

Java je objektovo orientovaný, čiastočne interpretovaný, staticky silne typovaný programovací jazyk. Syntax Javy vychádza z jazykov C a C++. Je výkonnejšia ako interpretované jazyky. Je často používaná pre vývoj webových aplikácií a hier. Java frameworky sú veľmi obľúbené najmä vďaka svojej rýchlosti. Nevýhodou Javy je, že pre jej beh z webového prehliadača je nutný plugin.

ASP .NET je súčasť .NET frameworku. Je postavený na CLR (Common Language Runtime), čo rieši problém kompatibility rôznych programovacích jazykov. Aplikácie sú predkompilované do jedného či niekoľkých málo DLL súborov, vďaka čomu sú rýchlejšie ako aplikácie napísané v skriptovacích jazykoch. Všetok kód je kompilovaný do byte code. Výhodou je, že programátori môžu realizovať svoje projekty v akomkoľvek jazyku podporujúcom CLR (C#, Visual Basic.NET, Managed C++, JScript .NET, mutácie Perlu alebo Pythonu a iné) [9]. A naopak nevýhodou je, že ku svojmu behu potrebuje server s operač-

ným systémom Windows, alebo implementáciu .NETu Mono ak ho chceme spustiť na Linux serveroch.

PHP je skriptovací interpretovaný, slabo typovaný jazyk, predovšetkým využívaný na tvorbu dynamických interaktívnych webov [5]. Podľa *Stone River eLearning* [12] je v súčasnosti najpoužívanejším programovacím jazykom na strane servera. Je použitý vo viac ako 75% všetkých webových stránkach. Tento jazyk disponuje veľkou podporou a nekladie žiadne špecifické požiadavky na hardware. Toto sú hlavné dôvody, prečo bol jazyk PHP vybraný pre vývoj tohto projektu.



Obr. 4.1: Graf použitia programovacích jazykov webových aplikácií na strane servera. (prevzaté zo [12])

4.2 Výber frameworku

Ná vývoj tejto aplikácie som sa rozhodla použiť framework, pretože nám ponúka výhody oproti čistému jazyku. Framework je aplikácia vyvíjaná viacerými programátormi alebo komunitou. Často nám uľahčuje vývoj aplikácie tým, že nám poskytuje pridané funkcionality, ako napríklad komunikáciu s databázou, nastavenie bezpečnosti, generovanie formulárov, routing (prekl. smerovanie) a iné. Framework má predom určenú štruktúru, ktorá sa počas písania aplikácie udržiava, a tak sa zdrojový kód stáva prehľadnejším. Aj vďaka tomu, je použitie frameworku veľmi vhodné pre väčšie projekty. Teraz sa bližšie pozrieme na 4 vybrané PHP frameworky.

Laravel je pomerne nový open source framework (rok vydania 2011) s MIT licenciou, ktorý sa zameriava na elegantný kód, rýchlosť a jednoduchosť. Jeho základným stavebným prvkom je Symfony. Má mnoho vstavaných modulov [3]. Používa šablónovací systém Blade. Každá nová verzia so sebou prináša dôležité užitočné moduly. V poslednej verzii 5.2 boli pridané funkcionality ako napríklad Authentication drivers, Array validation a iné. Long-time support (prekl. dlhodobá podpora) obsahovala prýkrát až verzia 5.1 [2].

Zend je úspešný framework, vyvíjaný samotnými PHP vývojármi s viac ako 15 mil. stiahnutí [23]. Zend framework ponúka veľké množstvo obľúbených funkcií a komponentov. Zend Technologies taktiež produkujú aplikačný server Zend Server PHP. Zend framework projekty nevyžadujú pre spustenie Zend Server, ale spustenie týchto projektov cez Zend Server môže priniesť značné výhody, ako napríklad introspekciu pre ľahšie odhaľovanie chýb, komplexné funkcie (prekl. template) a iné [8].

Nette je český open source framework vyvíjaný Nette Foundation a zároveň je najpoužívanejším PHP frameworkom u nás [16]. Má vysokú krivku učenia, podporuje prácu v tíme, má vlastnú debugovaciu knižnicu Tracy, ktorá sa udomácnila pod názvom Laděnka, používa šablonovací systém Latte, má vlastnú databázovú vrstvu NotORM a mnoho iných nástrojov. Disponuje českou dokumentáciou [13]. Nevýhodou pre vývojárov, ktorí neovládajú češtinu alebo slovenčinu, môže byť problém nájdenia riešených problémov pri vývoji aplikácií na rôznych fórach v angličtine, keďže sa viac využíva čeština. Napríklad na portáli Stackoverflow má zo všetkých spomínaných frameworkov najmenej označených príspevkov. Chýba tu generátor kódu, ktorý je dostupný u iných frameworkov.

Symfony Symfony je vyvíjaný a sponzorovaný spoločnosťou Sensio Labs, ktorý integruje mnoho open source projektov. Základná verzia Symfony projektu obsahuje iba najdôležitejšie komponenty. Tie ostatné je možné si pridať pomocou balíčkov zvaných Bundles. Taktiež má rozsiahle debugovacie “features” (prekl. funkcie). Tento framework je v porovnaní s ostatnými o trochu komplexnejší [7] a aj z toho dôvodu patrí spolu so Zend frameworkom medzi ťažšie naučiteľné PHP frameworky.

Podľa prieskumu na portáli *Sitepoint* [16] bol najpopulárnejším PHP frameworkom Laravel. Druhým najpopulárnejším frameworkom bol Symfony projekt a tretím sa stal Nette framework. Všetky tieto frameworky poskytujú pokročilú podporu MVC. Pre ich použitie je potrebné rozumieť objektovo orientovanému programovaniu.

MVC, Model-View-Controller je návrhový vzor, ktorý oddeľuje od seba model (prekl. datový model), view (prekl. užívateľské rozhranie) a controller (prekl. riadiacu logiku) [9]. Každá z týchto častí má presne určenú funkciu:

- **model** je datový objekt, ktorý slúži na komunikáciu s databázou.
- **view** je prezentačný objekt, ktorý tvorí užívateľské rozhranie pomocou získaných dát z models a controllers.
- **controllers** sa nachádzajú medzi models a views. Prijímajú požiadavky užívateľov, získavajú dáta z models a prenášajú ich do views.

4.3 Použité technológie

Prvou voľbou bol najnovší zo spomínaných frameworkov a to *Laravel*. Avšak po istej dobe som zistila, že to nebola správna voľba. Projekt sa stále rozvíja a nevýhodou bolo, že tu chýbali niektoré nástroje, ktoré boli už implementované v Symfony. Taktiež bol problém nájsť tutoriály pre novú verziu a dokumentáciu k nie úplne dobre zdokumentovaným častiam. Často boli dostupné riešenia pre staršie verzie, ktoré však neboli kompatibilné s novšími verziami. Možno to bolo neskúsenosťou začiatočníka, ale po pár týždňoch “bojov” s kompatibilitou nástrojov a problémom konfigurácií balíkov ako napr. Socialite, ktorý nebol v tej dobe príliš dobre zdokumentovaný, padlo rozhodnutie použiť na vývoj projektu iný framework. Z vyššie uvedených bol nakoniec vybraný Symfony framework. Jeho výhodou je stabilnosť, čistota kódu, pretože je nutné dodržiavať štruktúru ktorú rámec udáva, dlhodobá podpora vydaných verzií, ďalej veľká komunita programátorov napríklad na IRC alebo webovom portáli Stackoverflow. Keďže *Symfony* je založené na technológií MVC, máme oddelený frontend a backend. Tento framework používa systém komponent (Bundles), kde zvyčajne každá hlavná funkcionalita alebo stránka je jedna komponenta. Použitie tohto systému nám dáva možnosť znovupoužitia tejto komponenty v iných projektoch. Taktiež nám

ponúka mnoho externých komponentov, ktoré je možné stiahnuť a používať. Ak daná stiahnutá komponenta nespĺňa naše požiadavky, môžeme ju zmodifikovať podľa našich potrieb. Symfony nám dáva veľkú voľnosť pri výbere databázy. Nemá vlastnú databázovú vrstvu, ale tá môže byť ľahko nahradená napríklad s *Doctrine*. Tento framework používa šablonovací nástroj *Twig* a k testovaniu frameworku slúži *PHP unit*. Konfiguračné súbory je možné zapisovať vo formáte *XML*, *PHP* alebo *YML*. Implicitne sa používa *YML*, ktorý slúži na serializáciu dát, tak aby boli ľahko čitateľné a zrozumiteľné. Pevne je daná aj adresárová štruktúra *Symfony*. Je rozdelená do štyroch hlavných častí [18]:

- **app** – táto časť sa stará o chod celého frameworku. Sú tu uložené nastavenia a konfigurácie:
 - *console* je spustiteľný súbor, cez ktorý sa riadi celý framework v príkazovom riadu.
 - *AppKernel.php* zaisťuje správu komponentov.
 - v podadresári *app/config/* je uložená celá konfigurácia. Nastavenia frameworku sa nachádzajú v súbore *config.yml*. V súbore *parameters.yml* sú uložené parametre pripojenia na databázu a použitého maileru. Nastavenia presmerovania sú uvedené v súbore *routing.yml* a zabezpečenie, hierarchia prístupových práv sa nachádza v súbore *security.yml*.
 - do adresára *app/Resources/* sú ukladané často používané šablóny (templates) alebo súbory pre preklad.
 - taktiež sa tu nachádzajú adresáre *app/cache/* a *app/logs/*, ktoré obsahujú všetky cache a log súbory generované aplikáciou.
- **src** – v tejto časti sa nachádza väčšina zdrojového kódu napísanej aplikácie. Nachádzajú sa tu všetky komponenty (Bundles) vytvorené vývojárom aplikácie. Každá komponenta obsahuje väčšinu častí z návrhového vzoru MVC. Môžeme tu nájsť podadresár *Resources* v ktorom sa nachádzajú views vo formáte Twig a súbory určené pre smerovanie. Podadresár *Entity* obsahuje objekty, ktoré definujú tabuľky v databáze a ich metódy, ktoré umožňujú prácu s dátami. V podadresári *Form* sa nachádzajú objekty a funkcie ktoré definujú formuláre, ich vykresľovanie, a prácu s nimi. Podadresár *Controller* obsahuje Controller classes (prekl. triedy), ktoré riadia jednotlivé akcie.
- **vendor** – táto časť obsahuje všetky komponenty (Bundles), ktoré sú využívané frameworkom a všetky externé komponenty. Ak máme nejaké externé komponenty, ktorých funkcionality potrebujeme pozmeniť, odporúča sa zmeniť funkcionality v zdedenej komponente, ktorú umiestníme do adresára s vlastnými komponentami – *src*.
- **web** – obsahuje všetky súbory potrebné na zobrazenie užívateľského rozhrania, ako sú CSS súbory, JavaScript súbory a obrázky.

Pri práci so *Symfony* je veľmi užitočné použiť *Composer*. **Composer** je správca balíčkov používaný v moderných PHP aplikáciach. V *Symfony* nám pomáha sťahovať a spravovať externé komponenty [17]. Súbor v našom projekte nazvaný *composer.json* vo formáte JSON obsahuje použité knižnice a súbor *composer.lock* taktiež vo formáte JSON obsahuje verzie použitých knižníc, ktoré majú byť pre daný projekt použité.

Bootstrap bol použitý pre tvorbu vizuálnej stránky. Je to HTML, CSS a JS framework pôvodne vyvíjaný spoločnosťou Twitter. V auguste 2011 sa Bootstrap stal open source pod

licenciou MIT a v súčasnosti je druhým najobľúbenejším projektom na portáli Github.com. Principiálne ide o front-end framework, pre tvorbu rozhraní, s dôrazom na responsívny dizajn a opätovné použitie kódu. Zaujímavosťou Bootstrapu, je súčasné (v3.3) využívanie Less jazyka (implementácia v Javascripte) ako náhrada za CSS a v budúcej verzii (v4) je zase nahradená Sass (implementácia v Ruby) a SassScriptom, z ktorého po kompilácii vznikne CSS3 kompatibilný kód.

Databáza je potrebnou súčasťou tohto projektu. Pre tento projekt bolo zvolené použitie relačnej databázy konkrétne databázový systém MySQL. **MySQL** je najpopulárnejšia open source (prekl. slobodný software) databáza podporovaná na viacerých platformách [15]. Dáta sú ukladané do tabuliek, kde reláciám odpovedajú riadky tabuľky a atribútom názvy stĺpcov tabuľky [6]. Práca s MySQL databázou je vykonávaná pomocou dotazov napísaných v programovacom jazyku SQL (Structured Query Language), ktorého syntax je ľahko čitateľná.

Pri použití relačnej databázy a objektovo-orientovaného programovacieho jazyka je potrebné tieto dve odlišné technológie prepojiť. K tomuto účelu slúžia databázové vrstvy ako je **ORM**, Object-relational mapping (prekl. Objektovo relačné mapovanie). Symfony framework nemá žiadnu vlastnú databázovú vrstvu, avšak súčasťou každého štandardného Symfony projektu je Doctrine. Táto databázová ORM vrstva bola použitá pri vývoji tohto projektu. **Doctrine** je sada PHP knižníc, ktoré nám umožňujú pracovať s dátami ako s objektami. Využíva návrhový vzor *Data Mapper*, ktorý je reprezentovaný pomocou *Entity Manageru*. Entity v ňom uchovávajú štruktúru tabuliek a ich atribútov [10]. Jednou z kľúčových vlastností Doctrine je písanie databázových dotazov použitím *Doctrine Query Language* (DQL) ¹, kde názov entity predstavuje názov tabuľky a premenné entity predstavujú atribúty danej tabuľky.

¹DQL – je dotazovací jazyk využívajúci entity a objekty.

Kapitola 5

Návrh a implementácia projektu Couchsurfing pre hudobníkov

V tejto kapitole sa bližšie oboznámime s návrhom a implementáciou celého projektu. Povieme si o prostredí, v ktorom bol projekt vyvíjaný a o cloud riešení. Taktiež sa oboznámime s procesom návrhu, implementácie užívateľského rozhrania a databázy.

5.1 Lokálne vývojové prostredie

Na vývoj bol použitý operačný systém GNU/Linux, variácie Debian a IDE PhpStorm. Samozrejmosťou potrebnou k vývoju bol MySQL server s lokálnou databázou. Pre vývoj aplikácie bolo použité Symfony 2.7, PHP 5.6.14, MySQL 5.5.46.

5.2 Cloud

Na projekt bolo zvolené cloud riešenie od Red Hat-u, s názvom OpenShift. Pre projekty ako je tento, je možné použiť neplatený účet, na ktorom je možné mať tri projekty. *OpenShift* ponúka jednoduchú administráciu stránky, prístup k zdrojovým kódom pomocou terminálu, verejnú doménu a možnosť škálovateľnosti. Aplikácia na cloude využíva PHP 5.4, pretože spustenie a kompatibilita použitím PHP 5.4 so Zend Server 6.1 bola problematická. Pri rozličnom testovaní mnohých aspektov projektu, boli vytvorené mnohé samostatné inštancie webovej aplikácie (alpha, beta, gamma). Nevýhodou tohto cloudového riešenia sú problematické (kvôli obmedzeným právam) zmeny, napríklad nejakých konfiguračných súborov. Na správu databázy pomocou webováho rozhrania bol použitý *phpMyAdmin*. Projekt je dostupný na adrese <http://gamma-mymfriends.rhcloud.com>. Na cloude, kde sa projekt nachádza, sú použité verzie PHP 5.4.40, MySQL 5.5.45., phpMyAdmin 4.0.

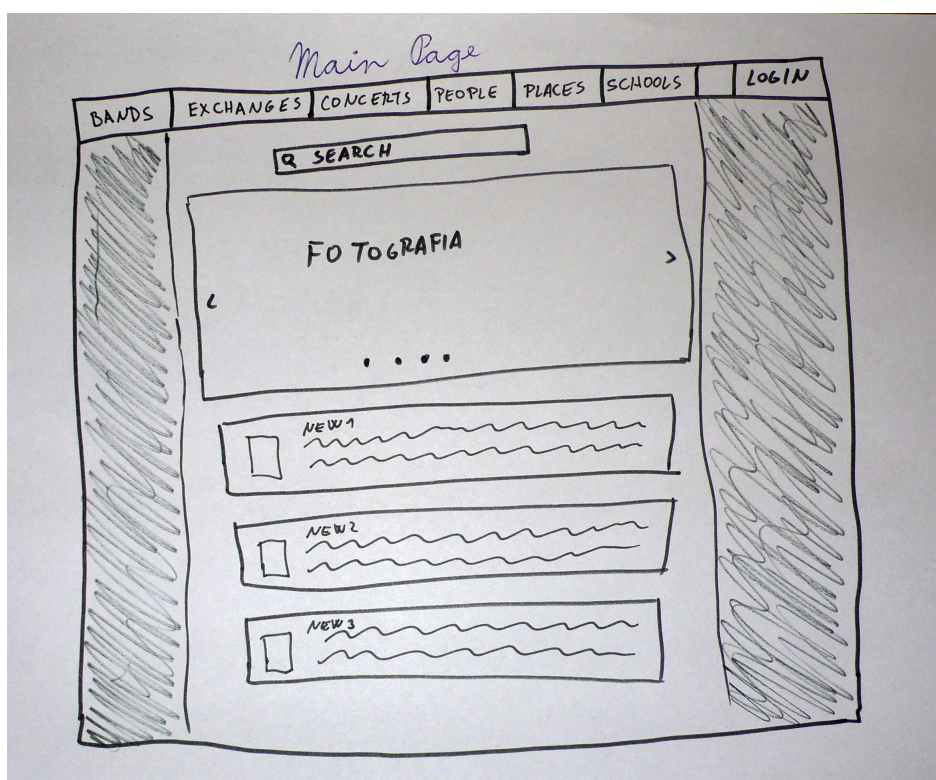
5.3 Návrh užívateľského rozhrania

V počiatočnej fáze navrhovania užívateľského rozhrania, je veľmi vhodné použiť jednu zo základných techník návrhu – *skicovanie*. Cieľom skicovania, je vytvoriť prvotné návrhy užívateľského rozhrania na papier. Pri návrhu užívateľského rozhrania je vhodné vytvoriť niekoľko skíc, porovnávať ich a modifikovať. Podľa knihy THE USER EXPERIENCE TEAM OF ONE, A Research and Design Survival Guide [4] najväčšie prednosti skíc sú:

- **cena** – ich cena je minimálna vďaka tomu, že nám stačí pero a papier.
- **rýchlosť vytvorenia návrhu** – behom niekoľkých minút je možné nakresliť viacero nápadov, čo je zvyčajne rýchlejšie ako návrh vytvorený pomocou počítačového programu.
- **pomáhajú bádať** – skicovanie nám pomáha premýšľať nad problémom a pomáha nám hľadať rozličné riešenia.
- **zabraňujú perfekcionizmu** – vytvorenie niekoľkých skíc pomáha tomu, aby dizajnér nezostal závislý od jedného nápadu.
- **rýchla spätná väzba** – ak dizajnér ukáže niekomu skice, tak dostane rýchlu spätnú väzbu zameranú na hlavné riešené problémy. Keď totiž ukáže ľuďom výsledný dizajn, tak sa sústreďujú viac na detaily ako sú farby, typografia a iné.

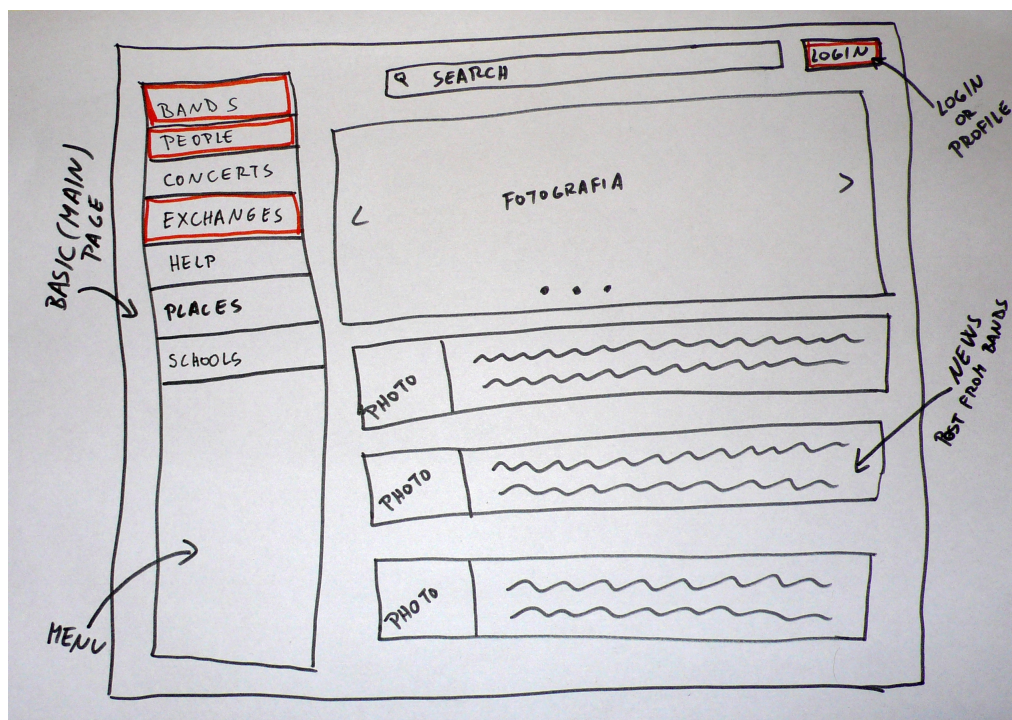
Metóda skicovania bola využitá aj pri vývoji tohto projektu. Táto metóda mi osobne viac vyhovuje ako návrh pomocou počítača, hlavne kvôli vyššie spomenutým výhodám. Taktiež, však vďaka vedomiu, že pozeranie na čistý biely papier podporuje moju kreativitu oveľa viac ako navrhovanie priamo na počítači. Pri vytváraní návrhu pomocou počítača, je moje podvedomie ako aj vedomie, ovplyvňované a rozptyľované aj tými najmenšími prvkami (ikonkami, vzormi, farbami...). Ďalšou výhodou skicovania, je možnosť prezerat si všetky skice naraz, tak ich porovnávať a získavať lepší prehľad o tom, ako je vhodné pri návrhu postupovať. V počiatočných fázach návrhu tohto projektu bolo vytvorených niekoľko skíc, ktoré boli porovnávané a modifikované. Jednu z počiatočných skíc môžete vidieť na obr.

5.1.



Obr. 5.1: Ručne kreslená skica návrhu užívateľského rozhrania domovskej stránky s hlavným menu orientovaným vo vrchnej časti stránky.

Pri tomto type návrhu, som sa inšpirovala existujúcimi riešeniami a hlavné menu som umiestnila k vrchnej časti stránky. Ostatné prvky sa nachádzali v strednej časti stránky. Po konzultácii tohto typu návrhu so známymi a prezretí si ďalších webových stránok, boli vytvorené iné skice. Ďalšie vytvorené skice mali menu orientované do ľavej časti stránky obr. 5.2, pretože väčšina obrazoviek používaných v dnešnej dobe má väčšiu šírku ako výšku. Menu orientované na ľavej strane bude výhodné aj pri budúcich rozšíreniach projektu, kedy bude potrebné pridať do neho nové položky. Používanie viacúrovňového menu je užívateľsky prívetivejšie taktiež viac v ľavej časti ako v hornej.



Obr. 5.2: Ručne kreslená skica návrhu užívateľského rozhrania domovskej stránky s hlavným menu orientovaným v ľavej časti stránky.

5.3.1 Bootstrap

Po naskicovaní užívateľského rozhrania, bolo potrebné vybrať vhodné komponenty pre tvorbu užívateľského rozhrania. Ako bolo spomenuté v sekcii výberu technológií, pre tento účel bol zvolený *Bootstrap*, konkrétne template (prekl. vzor) *SB Admin 2*. Tento template je dostupný pod MIT licenciou od organizácie Start Bootstrap. Prvky tohto template sú veľmi podobné prvkom bootstrapu, nájdeme tu niekoľko malých úprav a doplnkov pôvodných bootstrap štýlov. Mnoho prvkov, z ktorých bolo užívateľské rozhranie vytvorené, je súčasťou tohto template alebo Bootstrapu. Niektoré prvky boli upravené pre potreby vytvoreného projektu. Upravené prvky sa nachádzajú v súboroch s pôvodnými štýlmi za označením *CUSTOM*. Vďaka využitiu Bootstrapu a jeho *grid system* vytvorená aplikácia spĺňa podmienky responzívneho web dizajnu¹. V grafickom dizajne *grid* predstavuje štruktúru (zvyčajne dvoj-dimenzionálnu), ktorá sa skladá z pretínajúcich sa čiar. Vytvorená mriežka potom slúži ako rámec, na ktorom môže dizajnér usporiadať grafické prvky.

¹Responzívny web dizajn – predstavuje automatické prispôbenie web stránky veľkosti obrazovky.

5.4 Výber názvu pre projekt

"Pomenovanie je lingvistický dizajn a dobré doménové meno, je dôležitou súčasťou celkového dizajnu webových stránok" (Christopher Johnson)

Pri výbere vhodného mena projektu som sa riadila niekoľkými pravidlami. Snažila som sa navrhnúť meno, ktoré je ľahko zapamätateľné, opisuje cieľ stránky a jeho doména je dostupná. Vytvorený názov je anglický, pretože hlavným jazykom vytvorenej webovej aplikácie je angličtina, keďže cieľovou skupinou užívateľov sú hudobníci z rôznych krajín. Pri návrhu názvu aplikácie som si vypísala slová, ktoré vystihujú cieľ projektu a rôzne ich kombinovala. Takto vznikol názov určený pre túto webovú aplikáciu – *My Music Friends*, skrátene *MyMfriends*.

5.5 Logická štruktúra projektu

Pre každú hlavnú stránku v menu bol vytvorený bundle (komponenta). Týmto spôsobom je aplikácia rozdelená na menšie časti, ktoré môžu byť znovu použité v iných aplikáciach a stávajú sa ľahšie modifikovateľnými. Taktiež bol vytvorený samostatný bundle pre obsluhu prihlasovania užívateľov – *UserBundle*. Každý bundle využíva MVC návrhový vzor. Všetky vytvorené bundles sa nachádzajú v jednom z hlavných adresárov – *src/*. Hlavná cesta (v smerovaní) každého z bundlov má určitý názov (*/priklad/*). Vedľajšie stránky vykresľované určitým bundle majú vždy názov hlavnej cesty ako prefix (*/priklad/podstranka*). *ProfileBundle* používa viac hlavných ciest. Zoznam niekoľkých hlavných použitých bundles, ich cestu (v smerovaní) a ich účel môžete vidieť v tabuľke 5.1.

Bundle	Path (cesta)	Stará sa o časť
AppBundle	/, /home/	Homepage
Mymf\BandsBundle	/bands/	Bands, Choirs...
Mymf\ConcertsBundle	/concerts/	Concerts
Mymf\ExchangesBundle	/exchanges/	Exchanges, travelling
Mymf\HelpBundle	/help/	Help and Support
Mymf\PeopleBundle	/people/	People
Mymf\ProfileBundle	/profile/userName	Profile

Tabuľka 5.1: Tabuľka zobrazujúca bundles, ktoré sú súčasťou vytvorenej aplikácie.

Cesta profilu užívateľa obsahuje v sebe užívateľské meno. To mierne komplikuje presmerovanie užívateľa po prihlásení na jeho profil, keďže užívateľské meno je dostupné k použitiu až po dokončení autentifikácie. Tento problém bol vyriešený pomocou *AuthenticationSuccessHandler* implementovaním *AuthenticationSuccessHandlerInterface* ako *LoginSuccessHandler* (nachádza sa v *UserBundle*). Pomocou neho môžu byť vykonávané akcie medzi prihlásením a presmerovaním užívateľa na prvú stránku po prihlásení. V tomto *LoginSuccessHandler* získame užívateľské meno, ktoré je použité v ceste potrebnej na presmerovanie užívateľa.

5.6 Návrh a implementácia databázy

Každý použitý bundle sa stará o funkcionálnosť nejakej hlavnej časti celej webovej aplikácie. Z toho dôvodu bol navrhnutý systém databázy, kde máme jednu hlavnú tabuľku, ktorá je využívaná pri prihlasovaní užívateľov. Ostatné tabuľky sú na ňu zvyčajne naviazané použitím vzťahu 1:N. Každý z bundlov obsahuje Entity (tie sú pomocou ORM prevedené na objekty), ktoré sú využité k funkčnosti danej časti aplikácie. Všetky entity disponujú unikátnymi identifikátormi. Zoznam použitých entít a ich účel môžete vidieť tu:

- **VZOR: BundleName/EntityName**, *Table name* – využitie
- **UserBundle/User**, *user_login* – hlavná tabuľka, nachádzajú sa v nej základné údaje o užívateľoch, využíva sa napríklad pri registrácii, prihlasovaní, resetovaní a zmene hesla.
- **ProfileBundle/PAbout**, *p_about* – je naviazaná na *user_login*. Obsahuje informácie potrebné k zobrazeniu “About” informácií v profile.
- **ProfileBundle/PBasic**, *p_basic* – je naviazaná na *user_login*. Je druhou najhlavnejšou tabuľkou, pretože je používaná aj v iných bundles. Obsahuje základné údaje o užívateľovi, ktoré sú potrebné pri zobrazovaní prvkov, v ktorých sa nachádza užívateľ (je ich majiteľom).
- **ProfileBundle/PBio**, *p_bio* – je naviazaná na *user_login*. Obsahuje informácie potrebné k zobrazeniu “Music Bio” informácií v profile.
- **ProfileBundle/PContact**, *p_contact* – je naviazaná na *user_login*. Obsahuje informácie potrebné k zobrazeniu “Contact” informácií v profile.
- **ProfileBundle/Mymf**, *p_mymf* – je naviazaná na *user_login*. Táto tabuľka obsahuje *Mfriends* (priateľov, kontaktov).
- **ExchangeBundle/Exchanges**, *ex_exchanges* – je naviazaná na *user_login*. Táto tabuľka obsahuje vytvorené exchanges (prekl. výmeny) slúžiace k hľadaniu alebo ponúknutiu ubytovania.

5.7 Implementácia prihlasovania užívateľov

Pre implementáciu prihlasovania a registrácie užívateľov bol použitý najobľúbenejší externý open source bundle: **FOSUserBundle** od skupiny vývojárov *Friends of Symfony* (FOS). FOSUserBundle obsahuje podporu práce s užívateľmi a s databázou pomocou Doctrine ORM, ale aj iných databázových technológií, ako napríklad MongoDB/CouchDB ODM alebo Propel. FOSUserBundle taktiež poskytuje mnoho features (prekl. doplnky, vlastnosti), ako napríklad formulár pre login, registráciu, zmenu hesla, reset hesla a iné. Tento bundle neposkytuje autentifikáciu², avšak môže poskytnúť user provider, ktorý poskytne informácie o užívateľovi, potrebné pre jej jadro.

Po jednoduchej inštalácii FOSUserBundle-u pomocou Composer-u bola vytvorená entita User, ktorá je oddelená od triedy FOS\UserBundle\Model\User.

```
class User extends FOS\UserBundle\Model\User
```

¹ Autentifikácia – overenie pravosti.

Táto nová entita sa nachádza v adresári s vlastnými komponentami aplikácie *app/src/Mymf/UserBundle*.

Po vytvorení entity bolo potrebné nakonfigurovať FOSUserBundle. Do súboru *routing.yml* boli importované potrebné routing (prekl. smerovacie) súbory. V súbore *config.yml* boli definované: druh použitej databázy, povolenie potvrdzovacieho e-mailu registrácie, druh použitého mailer-u a ďalšie.

Na odosielanie e-mailov bol použitý mailer **SwiftMailer**. Ako e-mailový server bol použitý hosťovací e-mailový provider (prekl. poskytovateľ), *Gmail*. Pri konfigurácii tohto hosťovacieho provideru nastalo zopár komplikácií. Dôležitou súčasťou konfigurácie je *povolenie prístupu do gmail účtu pre menej bezpečné aplikácie* na použítom gmail účte. Avšak, aj keď bol tento prístup povolený a ostatné konfigurácie správne nastavené, stále bol problém s odosielaním e-mailov. Dôvodom tohto problému bolo nastavenie stredne silného hesla na gmail účte, ktoré aplikácia využívala. Až po zmene hesla na veľmi silné bol problém vyriešený. Po prvých testoch odosielania e-mailov odosielanie prestalo znovu fungovať. Tentokrát z dôvodu zablokovania gmail účtu jeho poskytovateľom. Tento problém bol vyriešený napísaním žiadosti poskytovateľovi o odblokovanie účtu s opisom použitia účtu.

V súbore *security.yml* sa nachádzajú napríklad tieto nastavenia:

- nastavenie druhu **enkryptcie užívateľských hesiel** – bol použitý bcrypt. *Upozornenie*: pri nižších verziách PHP treba doinštalovať knižnicu *ircmaxell/password-compat*. Bolo potrebné ju doinštalovať pri verzii PHP 5.4.4, pri verzii PHP 5.6.14 to nebolo potrebné. Doinštalovať ju je možné pomocou Composer-u príkazom:

```
php composer.phar require ircmaxell/password-compat
alebo
composer require ircmaxell/password-compat
```

- nastavenie **hierarchie rolí** – užívatelia majú rolu *ROLE_USER*, súčasťou *ROLE_ADMIN* sú práva *ROLE_USER* a *ROLE_SUPER_ADMIN* disponuje rolami *ROLE_ADMIN*, *ROLE_ALLOWED_TO_SWITCH*. Rola *ROLE_ALLOWED_TO_SWITCH* patrí medzi tie zaujímavejšie, pretože nám umožňuje prepnúť sa na akéhokoľvek užívateľa. Týmto spôsobom vizuálne vidíme rovnaké užívateľské rozhranie a funkcionality ako vybraný užívateľ.
- nastavenia **kontroly prístupov na stránky** – v tejto časti sú určené prístupové role alebo druh autentifikácie potrebné pre dané smerovacie cesty (path).

Taktiež sa tu priamo môžu nachádzať napríklad definovaní užívatelia s ich rolami *ROLE_ADMIN* alebo *ROLE_SUPER_ADMIN*. Avšak jednoduchšie, prehľadnejšie a najrýchlejšie môžeme priradiť potrebnú rolu administrátorovi pomocou Composer-u príkazom:

```
php app/console fos:user:promote
```

Ďalšie role, ktoré užívateľ môže mať a ktoré nám definujú druh autentifikácie sú:

- **IS_AUTHENTICATED_REMEMBERED** – túto rolu majú všetci prihlásení užívatelia. Majú ju aj tí, ktorí sú prihlásení pomocou “remember me cookie” (prekl. zapamätaj si ma).
- **IS_AUTHENTICATED_FULLY** – táto rola je silnejšia ako predchádzajúca. Majú ju prihlásení užívatelia, ale nemajú ju užívatelia prihlásení pomocou “remember me cookie”.

- **IS_AUTHENTICATED_ANONYMOUSLY** – túto rolu majú všetci, teda aj neprihlásení užívatelia.

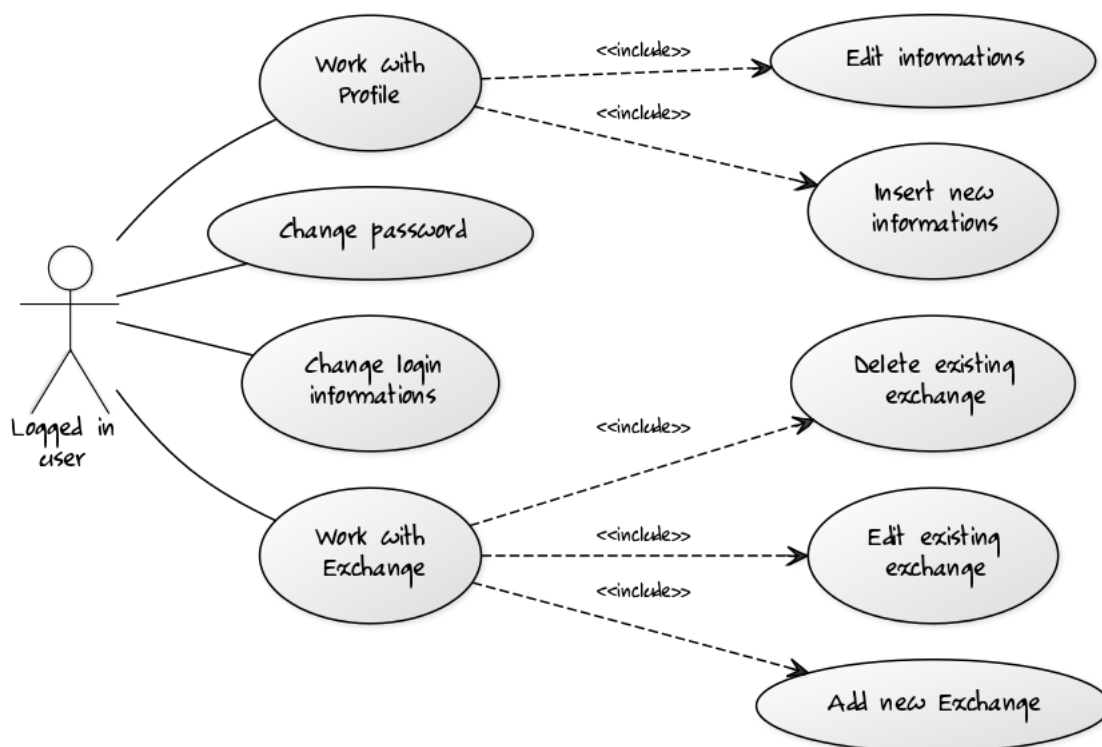
Použitím týchto rolí, môžeme umožniť alebo zakázať prístup na určité URLs (prekl. cesty). Získať informácie o rolách určitého užívateľa je možné vo viacerých častiach aplikácie (napr. Twig, Controllers), vďaka čomu je práca s prístupmi a zobrazovaním jednoduchšia.

Pri registrácii do vytvorenej webovej aplikácie sa vyžaduje od užívateľa zadanie hesla a jedinečného prihlasovacieho mena (prekl. username), a e-mailu. Po kliknutí na tlačidlo “Register” sa odošle na užívateľov e-mail URL adresa, ktorá obsahuje aplikáciou vygenerovaný jedinečný token. Po kliknutí na túto URL adresu, sa stáva registrácia platnou a je možné prihlásenie sa do aplikácie. Prihlasovanie je možné pomocou užívateľského mena ako aj jeho e-mailu a samozrejme hesla.

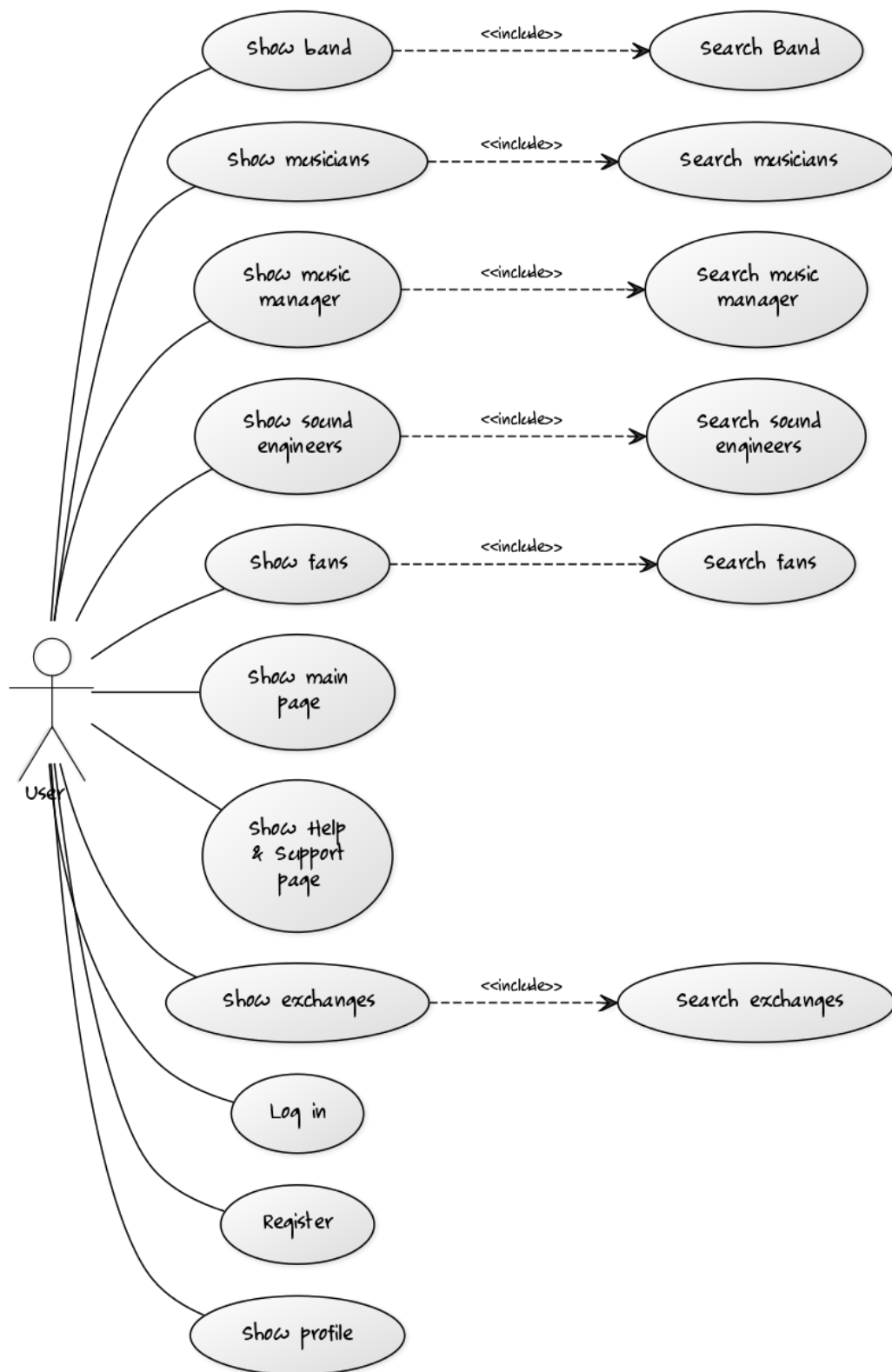
5.8 Implementácia užívateľského rozhrania

V tejto sekcii si opíšeme časti implementovaného užívateľského rozhrania a pozrieme sa na prípady použitia prihláseného užívateľa.

Užívatelia sa delia do viacerých skupín – *hudobníci, kapely a hudobné telesá, hudobní manažéri, zvukári a DJ a fanúšikovia*. Prihlasovanie je spoločné pre všetkých. Užívatelia sa zaradia do jednej zo skupín pri vyplňaní základných informácií vo svojom profile. Prihlásení užívatelia majú všetky možnosti použitia ako neprihlásení užívatelia.

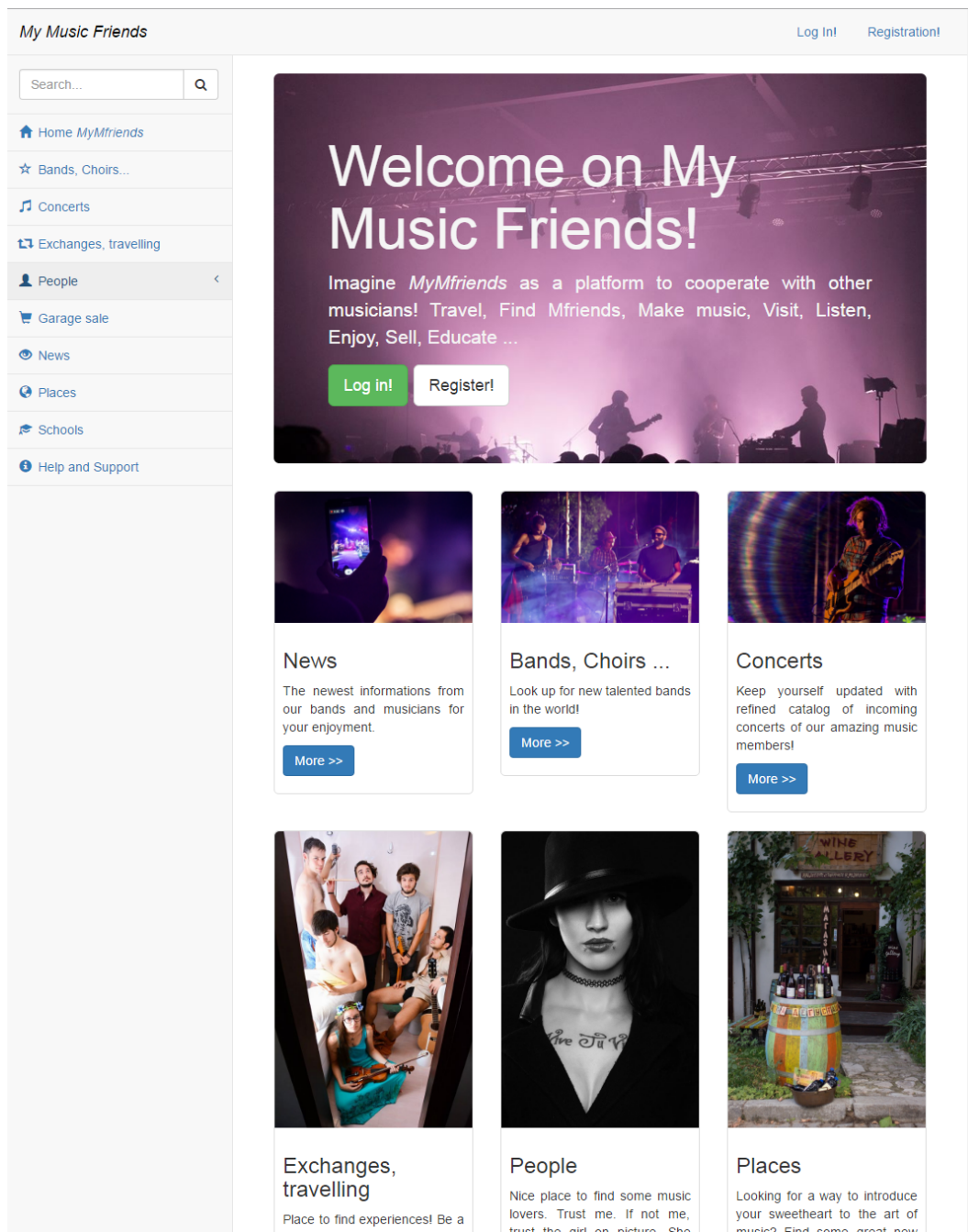


Obr. 5.3: Diagram prípadu použitia UseCase prihlásených užívateľov.



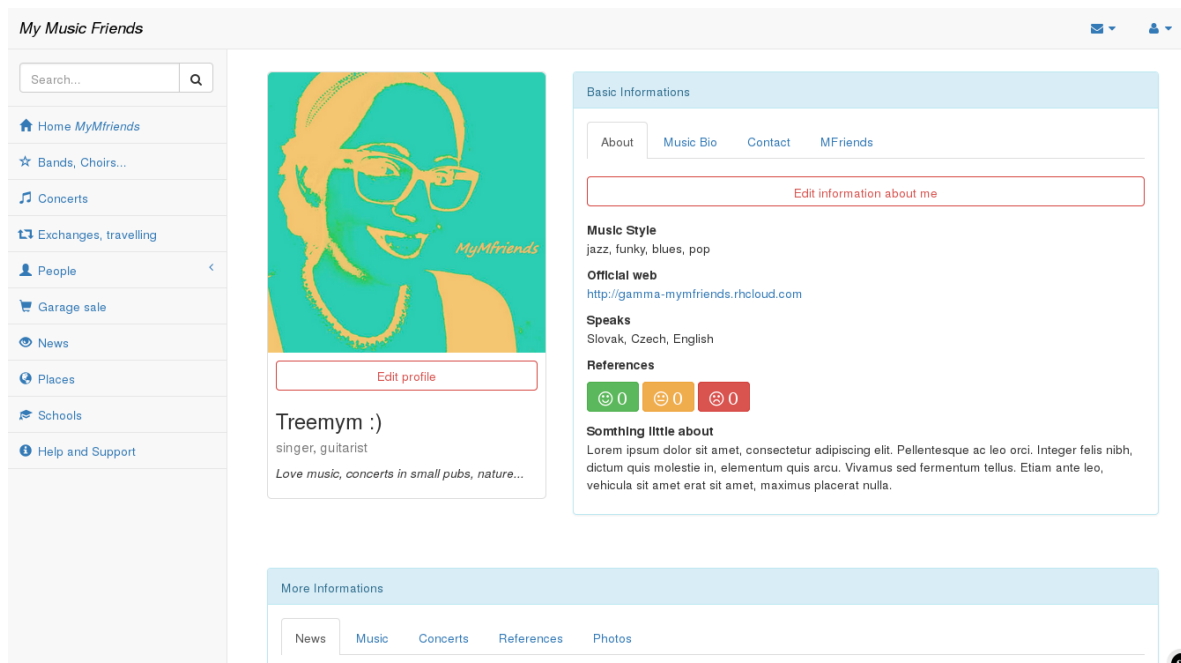
Obr. 5.4: Diagram prípadu použitia UseCase neprihlásených užívateľov.

Vytvorenú **Hlavnú stránku a Menu** môžete vidieť na obrázku 5.5. Hlavné menu tvorí ľavá časť. Vrchné menu obsahuje názov projektu a tlačidlá s možnosťou registrácie, a prihlásenia. Vrchné menu sa mení po prihlásení užívateľa, kedy ikonky “osoby” a “obálky” nahradia tlačidlá prihlásenie a registrácia. Po kliknutí na ikonku osoby sa zobrazí malé menu, ktoré odkazuje na profil užívateľa, zmenu hesla, nastavenia a odhlásenie užívateľa. Po kliknutí na ikonku obálky sa zobrazí menu, ktoré odkazuje na stránku s možnosťou napísania správy alebo na stránku s prijatými správami. Hlavná stránka obsahuje popis významu všetkých stránok a ich ilustračné fotografie, ktorých autorom je Michal Čerňanský [24].



Obr. 5.5: Vytvorené užívateľské rozhranie hlavnej stránky projektu My Music Friends.

Profil Profil užívateľa je verejne dostupný a pre umelcov môže slúžiť ako ich verejná prezentácia. Vytvorený profil užívateľa môžete vidieť na obrázku 5.6. Obsahuje profilovú fotografiu, základné informácie o užívateľovi, hudobnú históriu užívateľa, odkaz na sociálne siete užívateľa a iné. Farba profilu je rozdielna pre rôzne skupiny užívateľov.



Obr. 5.6: Vytvorené užívateľské rozhranie profilovej stránky užívateľa v projekte My Music Friends.

Súčasťou profilu nie je prepojenie so sociálnymi sieťami za účelom získania počtu “likes” na iných sociálnych webových aplikáciach. Taktiež, ani vlastný systém “likes” nebol implementovaný. Táto stránka je určená pre všetkých hudobníkov, kapely a poznať ich obľúbenosť prináša nevýhody. Ak by bol tento systém implementovaný, užívateľia by mali tendenciu, zameriavať svoju pozornosť na obľúbené kapely alebo hudobníkov. Tomuto sa chceme vyhnúť, pretože stránka má pomáhať užívateľom nájsť a objaviť nové, nepoznané kapely, hudobníkov. Pre potreby hodnotenia užívateľov (kde sa hodnotí spolupráca, pohostinnosť...) bol vytvorený systém referencií. Prihlásený užívateľ môže napísať referenciu komukolvek. Referencie sa delia do troch kategórií – pozitívne, negatívne, neutrálne.

Prihlásení užívateľia si môžu pridať akéhokoľvek iného užívateľa alebo kapelu do svojich *Mfriends*. Mfriends predstavujú kontakty alebo priateľov. Tento **systém priateľstva** nie je obojstranný. Mfriends slúžia hlavne užívateľovi pre rýchle nájdenie alebo uloženie kontaktov a kapiel, ktoré pozná, alebo sa mu páčia. Vďaka tomuto systému, nie je potrebné zložité rozdelenie možností “priateľstva” na obojstranné priateľstvá, “follow” (prekl. následovanie) a “like” (prekl. páči sa mi), ako je to implementované u facebooku. Týmto spôsobom sa aplikácia stáva prehľadnejšou a užívateľsky prívetivejšou.

Vzájomné **kontaktovanie užívateľov** nebolo zatiaľ implementované, pretože písomná komunikácia si vyžaduje kvalitnú implementáciu, ktorá je bezpečná a zabezpečená proti útokom botmi alebo šíreniu spamov. Takáto implementácia je zložitejšia a trvá dlhší čas. Pre potreby tohto projektu užívateľom postačí získať kontakty a komunikácia zatiaľ môže prebiehať pomocou iných sociálnych médií, ktoré sú bezpečnejšie. Užívateľia si však môžu

nájsť kontakty na seba v profiloch, kde sa nachádza ich e-mail a odkazy na sociálne siete. Táto časť však patrí medzi prvé v zozname potrebnej budúcej implementácie.

Hudobné výmeny sa nachádzajú pod časťou *Exchanges, travelling* v ľavom menu. Na tejto stránke užívatelia môžu nájsť alebo ponúkať ubytovanie.

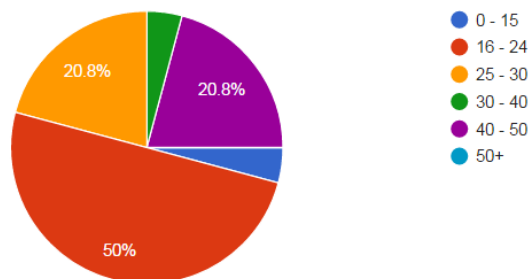
Kapitola 6

Testovanie a návrh na rozšírenie projektu

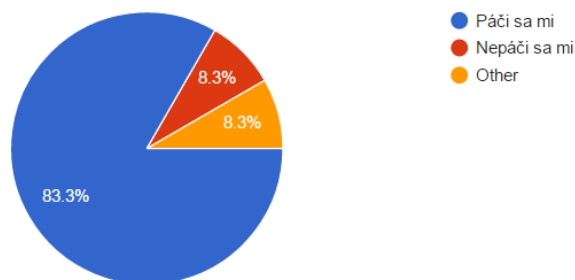
Dôležitou súčasťou vývoja užívateľského rozhrania je testovanie aplikácie užívateľmi. Vďaka tomuto testovaniu môžeme zistiť, či sa vývoj aplikácie ubera správny smerom, nájsť prípadné nedostatky, chyby. Po získaní spätnej väzby od užívateľov, navrhujeme vylepšenia alebo prípadné zmeny aplikácie pre potreby budúcej implementácie.

Po vytvorení základného užívateľského rozhrania webovej aplikácie *My Music Friends* bol vytvorený dotazník, v ktorom boli zadane úlohy testovacím užívateľom. Užívateľia hodnotili grafický dizajn webovej aplikácie a opísali rýchlosť, ktorou splnili dané úkony. Obsah a výsledky dotazníku môžete vidieť na nasledujúcich obrázkoch.

0. Vyberte vašu vekovú kategóriu. (24 responses)

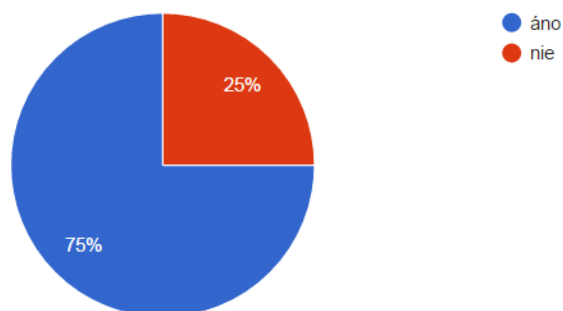


1. Prezrite si hlavnú stránku. Aký je váš názor na jej dizajn? (24 responses)

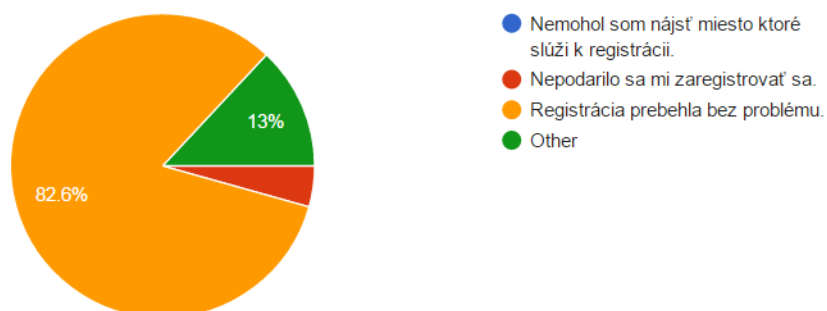


2. Viete k akému účelu stránka a jej podstránky slúžia z popisu na hlavnej stránke?

(24 responses)

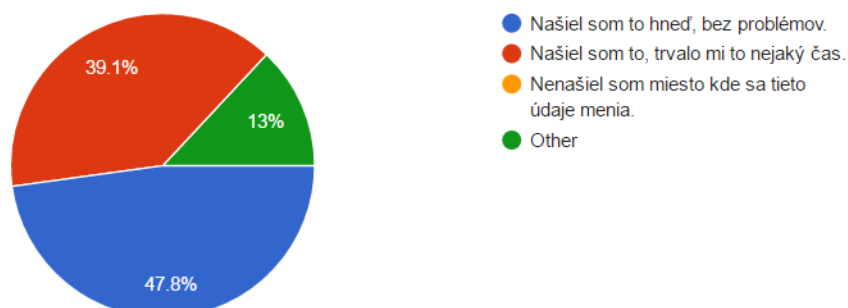


3. Zaregistrujte sa. Mali ste problém s registráciou? (23 responses)



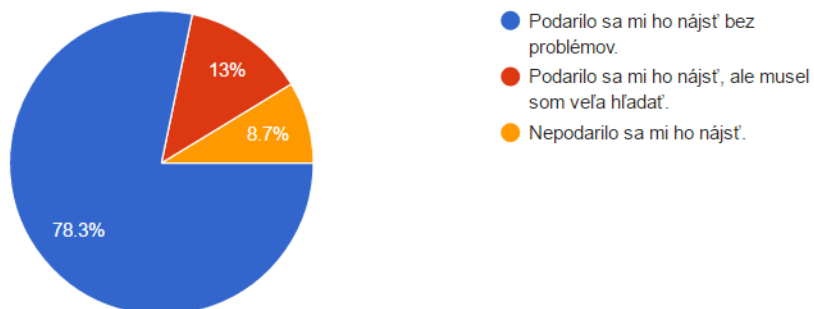
4. Skúste vyplniť vo vašom profile telefónne číslo / miesto kde bývate. Podarilo sa vám nájsť kde sa tieto údaje vyplňajú?

(23 responses)



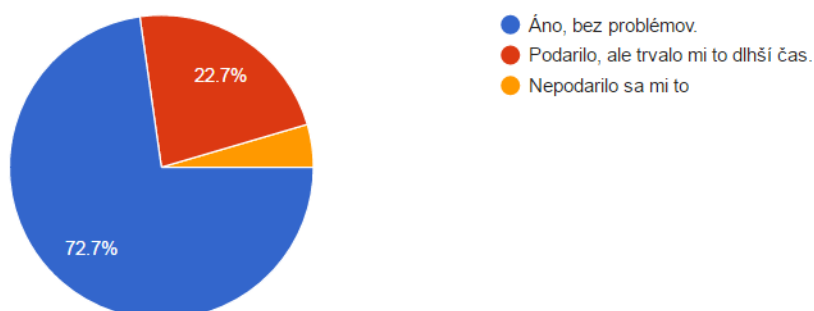
5. Skúste si zmeniť vaše prihlasovacie heslo. Našli ste miesto kde je to možné?

(23 responses)



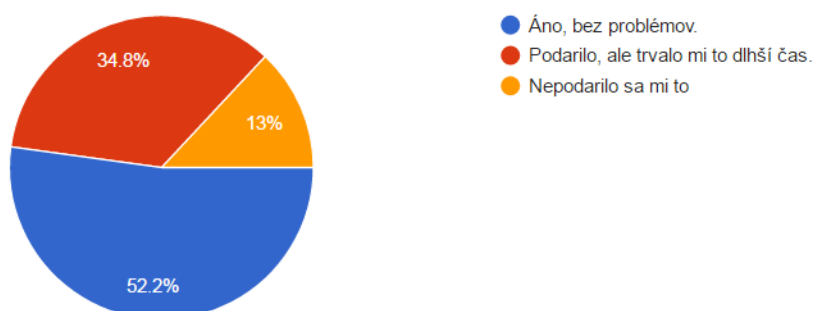
6. Skúste vyhľadať niekoho kto ponúka ubytovanie. Podarilo sa vám to?

(22 responses)



7. Vytvorte novú výmenu, kde budete ponúkať alebo hľadať ubytovanie. Podarilo sa vám to?

(23 responses)



Obr. 6.1: Na obrázkoch je zobrazený dotazník testu užívateľského rozhrania a jeho výsledky. V poslednej otázke, ktorá nie je zobrazená mohli užívatelia napísať svoje postrehy a nápady na vylepšenie. Najzaujímavejšími nápady na vylepšenie boli implementácia chatu a zmena formuláru pri vytváraní novej výmeny (tu bol návrh aby výber “Looking for: guest/host” bol vo formulári zobrazený ako prvý).

6.1 Návrh rozšírenia projektu

Vytvorená aplikácia je základnou verziou užívateľského rozhrania My Music Friends. Príkladám môj návrh vylepšení a doplnení implementácie:

- implementácia funkčných požiadavok z kapitoly 3, ktoré neboli implementované
- profilová stránka – doplniť ďalšie funkcionality ako nahrávanie fotografií a vytváranie albumov fotografií, nahranie video alebo audionahrávky ...
- implementácia všeobecného vyhľadávania
- implementácia rozšírených filtrov vyhľadávania
- otestovať bezpečnosť aplikácie a pozmeniť ju tak, aby bola bezpečnejšia
- implementácia vyhľadávania miest pomocou Google maps

Kapitola 7

Záver

V tejto práci bola popísaná prvá iterácia návrhu a implementácie užívateľského rozhrania webovej aplikácie *My Music Friends*. Na začiatku tohto cyklu boli popísané existujúce riešenia s podobným zameraním, základné princípy tvorby užívateľských rozhraní a následne na to, boli spísané požiadavky funkčnosti aplikácie. Potom bol popísaný výber technológii použitých pri vývoji aplikácie. V tejto časti je opísaný aj prvotný výber PHP frameworku a to, z akého dôvodu bol tento výber frameworku zmenený na iný použitý framework, pre vývoj tohto projektu. V kapitole 5 bol popísaný návrh a implementácia užívateľského rozhrania tejto aplikácie. Nachádzajú sa tu popísané aj riešené problémy, ktoré počas vývoja aplikácie nastali. Nakoniec, v kapitole 6 sa nachádzajú výsledky testu užívateľského rozhrania testujúcimi užívateľmi, a taktiež návrh budúceho vylepšenia a rozšírenia vytvoreného projektu.

Táto práca bola pre mňa veľmi obohacujúcou, vďaka získaniu nových vedomostí a skúseností. Popísaný návrh a implementácia užívateľského rozhrania, nie sú jediným správnym riešením, pretože každý dizajnér a vývojár používa iné metódy. Metóda návrhu, ktorá bola popísaná v kapitole 5 mi vyhovuje, ale postupy pri implementácii, by som so súčasnými vedomosťami pravdepodobne zmenila, tak aby práca využívala najnovšie verzie použitých technológií a bola kvalitnejšia.

Literatúra

- [1] Bandzone.cz: Bandzone official site. [Online], [navštívené dňa 2016-05-18].
URL <http://bandzone.cz/>
- [2] Barnes, E. L.: Laravel 5.1 is released. *Laravel News*, 2015, [Online], [cit. 2016-05-18].
URL <https://laravel-news.com/2015/06/laravel-5-1-released/>
- [3] Bean, M.: *Laravel 5 Essentials*. Birmingham B3 2PB, UK: Packt Publishing, 2015, ISBN 978-1-78528-301-7, 144 strán s.
- [4] Buley, L.: *THE USER EXPERIENCE TEAM OF ONE, A Research and Design Survival Guide*. Brooklyn, New York: Rosenfeld Media, LLC, 2013, ISBN 1-933820-18-7.
- [5] Doyle, M.: *Beginning PHP 5.3*. Indianapolis: Wiley Publishing, Inc., 2010, ISBN 978-0-470-41396-8.
- [6] Greenspan, J.; Bulger, B.: *MySQL/PHP Database Applications*. Foster City, CA, Chicago, IL, Indianapolis, IN, New York, NY: M&T Books, An imprint of IDG Books Worldwide, Inc, 2001, ISBN 0-7645-3537-4.
- [7] Grehan, R.: Fabulous PHP frameworks: Symfony. *IT WORLD*, 2011, [Online], [cit. 2016-05-18].
URL <http://www.itworld.com/article/2746670/development/fabulous-php-frameworks--symfony.html>
- [8] Grehan, R.: Fabulous PHP frameworks: Zend Framework. *IT WORLD*, 2011, [Online], [cit. 2016-05-18].
URL <http://www.itworld.com/article/2746706/development/fabulous-php-frameworks--zend-framework.html>
- [9] Jess Chadwick, T. S.; Panda, H.: *Programming ASP.NET MVC 4*. O'REILLY, 2012, ISBN 978-1-449-32031-7, 471 strán s.
- [10] Kubiček, M.: *Webový herní portál*. Diplomová práce, Vysoké Učení Technické v Brně, Brno, 2015.
- [11] Linowski Interaction Design: GoodUI ideas. [Online], [navštívené dňa 2016-05-18].
URL <http://www.goodui.org/>
- [12] Millares, G.: Top 5 Programming Languages Used In Web Development. *STONE RIVER eLEARNING*, 2015, [Online], [cit. 2016-05-18].
URL <http://blog.stoneriverelearning.com/top-5-programming-languages-used-in-web-development/>

- [13] Nette: Official web site. [Online], [navštívené dňa 2016-05-18].
URL <https://nette.org/>
- [14] Newsroom, F.: Stats. [Online], [navštívené dňa 2016-05-18].
URL <https://newsroom.fb.com/company-info/>
- [15] Oracle: Oracle official web site, Mysql Overview. [Online], [navštívené dňa 2016-05-18].
URL <http://www.oracle.com/us/products/mysql/overview/index.html>
- [16] Skvorc, B.: The Best PHP Framework for 2015: SitePoint Survey Results. *Sitepoint*, 2015, [Online], [navštívené dňa 2016-05-18].
URL <http://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/>
- [17] Symfony: Symfony Official Documentation, Installing Composer. [Online], [navštívené dňa 2016-05-18].
URL <http://symfony.com/doc/current/cookbook/composer.html>
- [18] Symfony: Symfony Official Documentation, Structuring the Application. [Online], [navštívené dňa 2016-05-18].
URL http://symfony.com/doc/2.7/best_practices/creating-the-project.html
- [19] The Couchsurfing team: Couchsurfing official site. [Online], [navštívené dňa 2016-05-18].
URL <https://www.couchsurfing.com/>
- [20] The Couchsurfing team: Couchsurfing official site, about. [Online], [navštívené dňa 2016-05-18].
URL <http://www.couchsurfing.com/about/about-us/>
- [21] Trashkimono: Trashkimono official site. [Online], [navštívené dňa 2016-05-18].
URL <http://trashkimono.com/dynastatics.php?st=about>
- [22] Weinmanová, L.: *Velká kniha webdesignu.4*. Koželužská 7, 602 00 Brno: ZONER software s.r.o., 2004, ISBN 80-86815-10-2.
- [23] Zend Framework 2: Zend Framework 2 official site, About. [Online], [navštívené dňa 2016-05-18].
URL <http://framework.zend.com/about/>
- [24] Čerňanský, M.: Fotogaléria. [Online], [navštívené dňa 2016-05-18].
URL <http://michalcernansky.net>

Prílohy

Zoznam príloh

A Manual a obsah CD

34

Príloha A

Manual a obsah CD

Prílohou k tejto práci je **CD**, ktoré obsahuje zdrojové kódy práce. Nachádzajú sa tu všetky potrebné súbory pre spustenie aplikácie.

Pre **spustenie** projektu nakopírujte súbory do počítača. Je potrebné mať vhodné verzie PHP a MySQL serveru (viz. sekcia Implementácia – Lokálne vývojové prostredie / Cloud). Postupujte podľa týchto bodov:

- Nainštalujte si Composer.
- Spustite MySQL server.
- Vytvorte prázdnu databázu a nastavte jej správne údaje v súbore *app/config/parameters.yml*.
- Pre vytvorenie tabuliek v databáze zadajte príkaz:

```
php app/console doctrine:schema:update --force
```

- Alikáciu spustíte príkazom:

```
php app/console server:run
```