

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

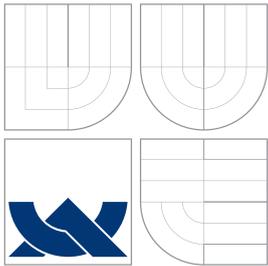
TAJMLAJN.CZ - WEBOVÁ SLUŽBA PRO PLÁNOVÁNÍ
DOSAŽENÍ CÍLE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

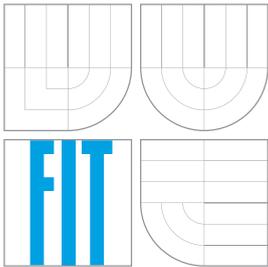
AUTOR PRÁCE
AUTHOR

MICHAL DOVIČOVIČ

BRNO 2015



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

TAJMLAJN.CZ - WEBOVÁ SLUŽBA PRO PLÁNOVÁNÍ DOSAŽENÍ CÍLE

TAJMLAJN.CZ - WEB SERVICE FOR GOAL PLANNING

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MICHAL DOVIČOVIČ

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. IGOR SZŐKE, Ph.D.

BRNO 2015

Abstrakt

Cílem této práce je návrh a realizace webové aplikace, která slouží k plánování a vede k dosažení cílů. Aplikace umožňuje vyhledávání přátel v této službě a vzájemné podporování na základě úspěchů. Dále mohou uživatelé komunikovat se svým vedoucím a nebo využít přihlašování na konzultace. Tato služba je postavena na frameworku AngularJS, jQuery a její serverová část je implementovaná v jazyce PHP. Díky responzivnímu designu poskytuje možnost prohlížení na mobilních zařízeních.

Abstract

The goal of this thesis is to design and implement a web application which is intended for planning and leads to achievement of goals. The application allows to search for friends in this service and mutually support each other on the basis of achievements. Users can also communicate with their supervisor or make an appointment for a consultation. This service is based on AngularJS, jQuery frameworks and its server side is implemented in PHP. Thanks to the responsive design it is also possible to browse the pages via mobile devices.

Klíčová slova

časová os, webová, aplikace, plánování, dosažení, cíl

Keywords

timeline, web, application, planning, achievement, goal

Citace

Michal Dovičovič: Tajmlajn.cz - webová služba pro plánování dosažení cíle, bakalářská práce, Brno, FIT VUT v Brně, 2015

Tajmlajn.cz - webová služba pro plánování dosažení cíle

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Igora Szóke, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Michal Dovičovič

20. května 2015

Poděkování

Chcel by som poďakovať Ing. Igorovi Szóke, Ph.D. za vedenie mojej práce, usmerňovanie a cenné rady počas jej tvorby. V neposlednom rade ďakujem rodičom a ostatným, ktorí ma podporovali počas štúdia.

© Michal Dovičovič, 2015.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	2
2	Cieľ webovej služby Tajmlajn.cz	4
3	Návrh implementácie	9
3.1	HTML - Bootstrap	9
3.2	CSS	10
3.3	JavaScript - klient	11
3.3.1	AngularJS	11
3.3.2	jQuery	12
3.4	MD5	13
3.5	PHP - server	13
3.6	API	14
3.7	Databáza	15
3.7.1	Vysvetlenie jednotlivých entít	16
3.8	Implementácia	18
3.8.1	Registrácia pomocou sociálnej siete	18
3.8.2	Udržiavanie relácie	18
3.8.3	Hlavné menu	19
3.8.4	Funkcionalita notifikácií	19
3.8.5	Zobrazovanie grafov	20
3.8.6	Filtrovanie položiek zoznamu	20
4	Testovanie	22
4.1	Dosiahnuté ciele a budúcnosť aplikácie	23
5	Záver	25
A	Obsah CD	27
B	Plagát	28

Kapitola 1

Úvod

Problémom dnešných, hlavne mladých ľudí sa stal fakt, že všetkú prácu, ktorú by mali robiť priebežne odkladajú na neskôr a tak množstvo týchto úloh narastá častokrát až do neúnosných rozmerov. Existuje mnoho časových plánovačov, angl. “Time managment”, no väčšina s nimi odmieta pracovať, prípadne neobjavili ich dôležitosť a tak nad týmto prístupom ani neuvažujú. Jedným z dôvodov môže byť aj skutočnosť, že časové plánovače predstavujú prácu navyiac a to ľudí mnohokrát odrádza. Táto služba sa nesnaží byť jedným z ďalších odrádzajúcich plánovačov, ale práve naopak – možnosťou ako získať plnú kontrolu nad svojim projektom a neustály kontakt so svojim vedúcim v prípade potreby skonzultovať problém, či už osobne alebo krátkou výmenou správ. Keďže bola služba testovaná študentmi píšucimi bakalárske alebo diplomové práce, obsahuje základné dáta prispôbené tomuto typu projektu. Vďaka predpripravenej forme a rozdeleniu akademického roku do jednotlivých týždňov sa navyše študenti nemusia starať o plánovanie. Jednou z hlavných motivácií pre vznik portálu *Tajmlajn.cz* bola vzájomná podpora študentov. Ponúka možnosť zistiť ako pristupujú k projektu užívatelovi priatelia a tak aj touto formou vyvolať u študenta vôľu pristupovať k dlhodobému projektu zodpovedne a pravidelne. Názov služby *Tajmlajn.cz* je odvodený od anglického slova “timeline”, čo v preklade znamená časová os. Tento názov však nezodpovedá zároveň názvu domény. Táto aplikácia sa momentálne nachádza na adrese <http://www.stud.fit.vutbr.cz/~xdovic00/tajmlajn/>.

V tejto dobe je vývoj a využívanie webových aplikácií stále populárnejší, aj napriek vznikajúcim mobilným aplikáciám. Častokrát sa popri webových službách vyvíjajú aj mobilné, ako ich plne alebo čiastočne “offline” varianty. Táto bakalárska práca bol navrhnutá a implementovaná ako webová aplikácia, ktorej úlohou je pomôcť pri definovaní, plánovaní a sledovaní postupu k dosiahnutiu cieľa. Služba je prvotne určená pre tých, ktorí ukončujú niektorú z úrovní vysokej školy a pracujú na bakalárskej alebo diplomovej práci, avšak je rozšíriteľná aj o ďalšie dlhodobé projekty. Umožňuje zaznamenávať pokroky pri práci, zdieľať ich s priateľmi, ktorí taktiež používajú túto webovú aplikáciu, pridávať si pripomienky a koordinovať tak svoj postup v danom projekte. Rovnako ponúka možnosť komunikácie formou správ medzi vedúcim projektu a jeho vypracovateľom. V prípade, že by samotné správy nestačili a je potrebné vyriešiť a skonzultovať dôležitejší problém, študent sa prihlási na niektorý z termínov vypísaných jemu prideleným vedúcim projektu.

Ďalšou z výhod tohto portálu je multiplatformnosť, funkčnosť na najpoužívanejších prehliadačoch Google Chrome, Mozilla Firefox, Internet Explorer a Opera. Taktiež dbá na prehľadnosť, rýchlu odozvu jednotlivých krokov a intuitívnosť užívateľského rozhrania. Pre svoj chod využíva najmodernejšie knižnice a postupy. Funguje ako jednostránková aplikácia s dynamickým prekresľovaním obsahu, čo umožňuje do určitej miery konzistentnosť

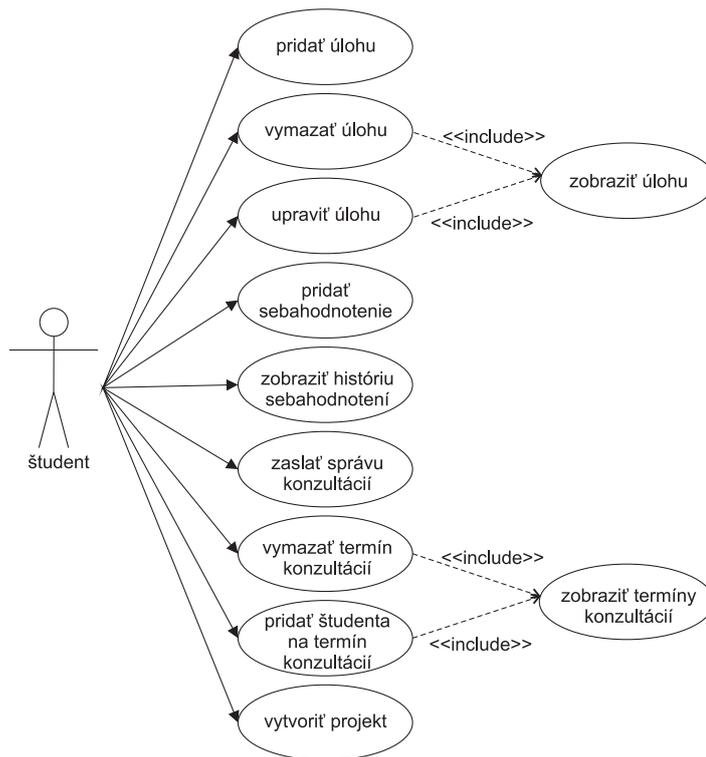
jednotlivých stránok, jej rýchlejšie prehliadanie a využívanie. Vzhľadom na implementáciu jednotného rozhrania vzniká do budúcnosti potenciál pre vývoj mobilných aplikácií so znovupoužitím serverovej časti. Jeden server teda môže obsluhovať viacero klientských aplikácií rôznych platforiem.

Vytýčený cieľ tejto webovej služby je ďalej popísaný v kapitole 2. Kapitola 3 rozoberá problém do väčšej hĺbky a to spôsob implementácie, architektúru celej služby a jednotlivých komponent. A na záver testovanie v kapitole 4, ktorá popisuje spôsob, akým bola táto webová služba nasadená do reálneho používania, jej upravovanie a opravu chýb na základe sledovania aktivity používateľov.

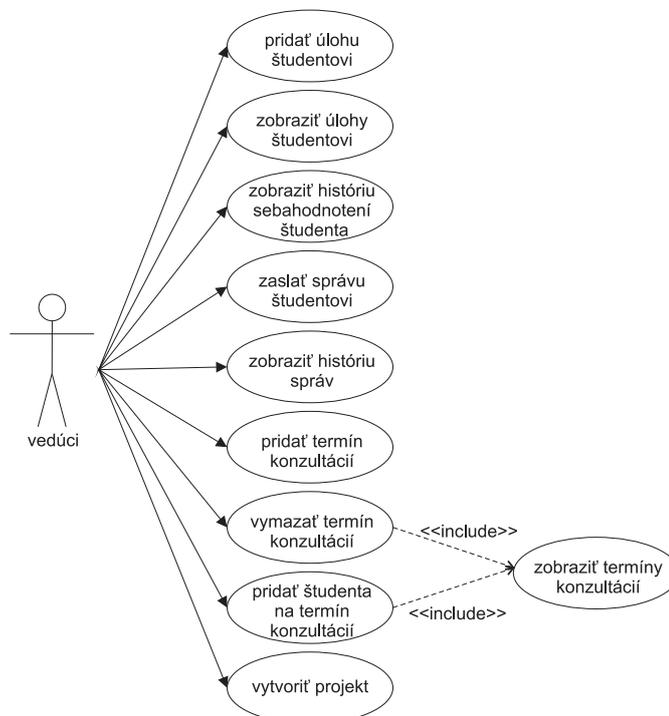
Kapitola 2

Cieľ webovej služby Tajmlajn.cz

Ako už bolo spomenuté, služba *Tajmlajn.cz* má za úlohu pomáhať pri vývoji dlhodobých projektov a to všetko z jedného miesta. Jej cieľom je dosiahnuť to, aby sa eliminoval stres a dobiehanie práce tesne pred odovzdaním alebo ukončením projektu. V tomto období zvyčajne vznikne najviac chýb, preto je nutné sa tejto skutočnosti vyhnúť a prácu dokončiť už v predstihu. Pravidelný a intenzívny prístup zefektívňuje prácu a poskytuje viac času na premyslenie vzťahov a architektúry jednotlivých podzložiek projektu. Cieľom tejto bakalárskej práce nie je len aplikácia s jednoduchou správou úloh, ale ide aj o nadobudnutie väčšieho rozhľadu účinkujúcich v danej problematike. Keďže je táto aplikácia určená ku koordinácii projektov, jej používateľov by mala rozdeľovať na dve skupiny: vedúci projektu a osoba, ktorá projekt vykonáva. Logicky by mala byť rozdielna aj ich funkcionality, práva a postavenie vzhľadom k práci. Obrázky 2.1 a 2.2 zobrazujú výrez diagramu prípadov použitia pre uvedenie príkladu rozdielnosti základnej funkcionality pre používateľský účet typu profesor (vedúci projektu) a študent.



Obrázok 2.1: Výrez diagramu prípadov použitia pre rolu študenta



Obrázok 2.2: Výrez diagramu prípadov použitia pre rolu vedúceho

Vďaka správe konzultácií sa naskytuje možnosť synchronizácie časových možností k stretnutiu. Ak toto stretnutie nie je možné z momentálnych časových dôvodov alebo sa jedná len o menší problém, systém by mal ponúknuť vyriešiť ho prostredníctvom textových správ. Medzi hlavné motívy tejto služby patrí aj podpora od priateľov. Používateľ by mal mať možnosť nahliadnuť do postupu spriateľených ľudí v tejto sieti a tak sa oboznámiť s ich aktivitami. Z psychologického hľadiska by daná osoba mala získať väčšiu motiváciu ak zistí, že jej kolega je na tom lepšie a bude chcieť tento rozdiel dobehnúť.

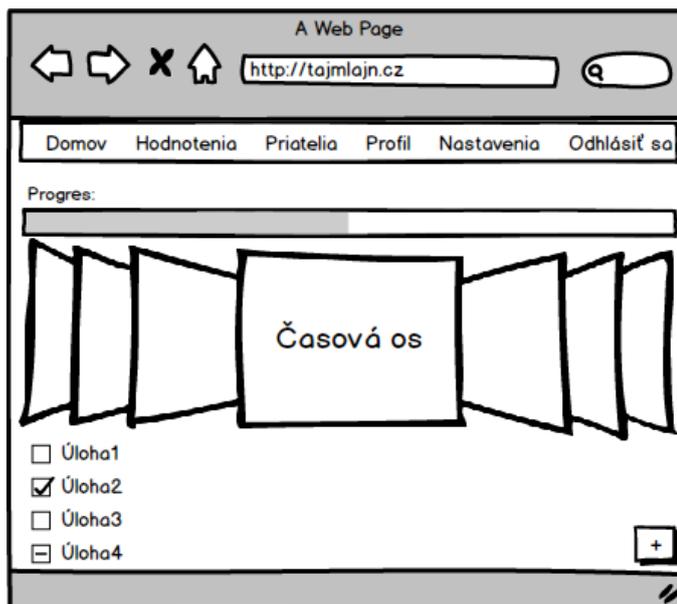
Cieľom je vytvoriť webovú aplikáciu založenú na jazyku JavaScript s podporou univerzálneho API (z angl. Application Programming Interface - rozhranie pre programovanie aplikácií). Služba bude podliehať vzhľadu moderných aplikácií a mala by zodpovedať potrebám a návykom používateľa. Je teda nutné brať ohľad na rozloženie jednotlivých prvkov na stránke a dbať na rýchlosť odozvy na jednotlivé pokyny zo strany návštevníka. Zobrazenie jednotlivých stránok by malo byť jednoduché a zároveň výstižné. Je preto potrebné zvážiť množstvo informácií na stránku.

Jedným z cieľov počas vývoja je navrhnuť a zrealizovať jednotné API, aby bolo k systému možno v budúcnosti napojiť ďalšie zariadenia, a tak celá služba mohla pracovať s rovnakými údajmi a teda rovnakou databázou. Z tohto dôvodu je potrebné rozdeliť aplikáciu na viac prvkov, pričom každá komponenta bude mať na starosti inú funkcionálnosť. Pri API sa teda jedná o časť aplikácie, ktorá obsluhuje zariadenia, v tomto prípade webového klienta, a poskytuje mu potrebné informácie a funkcionálnosť, ktorú vyžaduje. Tento prvok bude kompletne zastrešovať prácu s dátami a pripravovať ich tak aby zariadenie, ktoré o ne žiada malo s nimi čo najmenej práce. Rovnako tak aj s ukladaním užívateľských údajov a informácií potrebných pre chod celej služby.

Na svete existuje veľké množstvo aplikácií, ktoré vyžadujú registráciu. Rovnako to bude aj v tomto projekte. Okrem natívnej registrácie by mal systém ponúkať navyše aj prihlásenie pomocou sociálnej siete. Týmto sa výrazne skráti doba registrácie, zbytočné opakovanie vyplňania rovnakých údajov a používateľ môže plnohodnotne aplikáciu používať. Zo sociálnej siete sa získajú základné informácie o danej osobe a prípadne jej profilová fotka.

Po spustení každej aplikácie by sa mala zobrazíť úvodná stránka. Cieľom tejto stránky je informovať používateľa hneď pri vstupe do aplikácie o najdôležitejších udalostiach a informáciách. Je veľmi dôležité vedieť dobre vyvážiť množstvo údajov a ich celkovú obsahovú hodnotu. Veľké množstvo detailov na jednej stránke môže diváka odradiť od ich sledovania a rovnako priviesť do zmätku a napokon dôležité údaje sa v ňom môžu jednoducho stratiť. Hlavnou úlohou je teda oboznámiť používateľa s najdôležitejšími poznatkami. Keďže táto služba má udržiavať kontinuálny vývoj projektu, mala by na tejto stránke zobrazovať obdobie v akom sa projekt nachádza, napr. koľko času ostáva do ukončenia projektu a ďalej úlohy, ktoré je nutné vykonať, aby projekt napredoval. Vzhľadom na to, že názov aplikácie je odvodený od časovej osi, úvodná stránka by tak isto mala obsahovať časovú os celého projektu. Tento pohľad rozdeľuje projekt na menšie časové úseky (napríklad týždne) a popisuje, čo je nutné urobiť v konkrétnom časovom úseku. Cieľom by malo byť ukončenie alebo odovzdanie práce. Používateľ by mal mať možnosť vytvárať si a používať vlastné projekty. Rovnako bude mať možnosť si vybrať z predpripravených projektov, napríklad diplomová práca. Pri prvotnom návrhu niektorých sekcií, ako napríklad aj úvodná stránka, bol využitý voľne šíriteľný nástroj *Balsamiq Mockups*. Vďaka náčrtom z tohto programu sa skrátilo rozhodovanie o umiestňovaní jednotlivých elementov po stránke a ich základný vzhľad. Obrázok 2.3 zobrazuje prvotný návrh hlavnej stránky, ktorý vznikol ešte pred samotným vývojom a vytvorením projektu v repozitári.

Konzultácia je jedným s najdôležitejších udalostí počas vývoju projektu. Vďaka týmto



Obrázok 2.3: Náčrt hlavnej stránky s využitím nástroja Balsamiq Mockups

stretnutiam môže vedúci tohto projektu usmerňovať svojich (v prípade diplomových prác) študentov a naopak študenti vyjadriť svoje myšlienky a získať na ne kvalifikovanú spätnú väzbu. Pomocou sekcie, ktorá správu konzultácií bude obsahovať, je možné termíny konzultácií spravovať. Používateľ v roli profesora pridáva konzultácie pre svojich študentov, aby sa na ne mohli prihlasovať. Dokáže ich zrušiť alebo k danému termínu priradiť niektorého zo zoznamu. Študent bude mať potom vo svojom náhľade možnosť zobrazíť všetky termíny, ktoré pridal jeho vedúci a tak si zvolíť ten, ktorý mu najviac vyhovuje. Vzhľadom na to, že na konzultácie je potrebné chodiť pravidelne, vedúci projektu by mal mať možnosť sledovať túto dochádzku. Vďaka pravidelným stretnutiam je možné vyhnúť sa napríklad aj odkloneniu od témy a včas zistiť vznikajúci problém. Ďalšou dôležitou podmienkou je fakt, že osoba pracujúca na tejto práci by mala prísť pripravená. Teda priniesť potrebné podklady na konzultovanie, mať naštudovanú danú pomôcku alebo knižnicu, ktorú chce do projektu zahrnúť atď., a priniesť vždy niečo nové, aby mohla odprezentovať svoj pokrok od posledného stretnutia svojmu vedúcemu. Tieto skutočnosti by mal byť vedúci rovnako schopný zapísať napríklad formou poznámky.

Z dlhodobjšieho hľadiska by mal byť systém schopný udržiavať celú históriu projektu vzťahujúcu sa k špecifickej osobe. Používateľ disponuje možnosťou spravodlivo ohodnotiť sám seba za predchádzajúce časové obdobie, prípadne odôvodniť toto hodnotenie krátkou poznámkou a logicky sa k tomuto hodnoteniu musí vedieť v budúcnosti vrátiť. Systém teda musí ukladať celý postup jednotlivých užívateľov a dokázať ich zobrazíť pomocou prehľadného grafu, na ktorom je zobrazený vývoj v čase a prípadne kompletnú históriu aj s uloženými poznámkami. Vďaka týmto možnostiam môže používateľ odhaliť včas prípadnú časovú stratu a tú čím skôr dohnať, aby sa vyhol stresu a chybám spojeným s doháňaním práce krátko pred odovzdaním.

Vedúci projektu má prehľad o svojich študentoch. V ideálnom prípade na jednej stránke dokáže zistiť všetky potrebné informácie o danej osobe. Mal by mať možnosť nahliadnuť do histórie postupu v práci danej osoby na projekte, pridať mu novú úlohu, ktorú by mal vykonať a zobrazíť zoznam úloh ktoré už vykonal alebo má naplánované. V prípade nut-

nosti krátkej konzultácie by mal mať vedúci projektu možnosť zaslať tomuto študentovi správu. Vedúci projektu by mal mať možnosť spravovať svojich študentov a bez ohľadu na ich súhlas si ich pridať do svojho zoznamu. Keďže projekt je primárne určený pre Fakultu Informačných Technológií VUT v Brne, identita vedúceho je pri registrácii overená e-mailovou adresou zamestnanca školy. Nevznikne tak situácia, že by študent bol priradený osobou, ktorá sa za vedúceho neprávom vydáva.

Používateľ s právami vedúceho disponuje možnosťou pridávať nové projekty, na ktorých je možné pracovať. Systém obsahuje predpripravený, základný projekt spolu s predpripravenými úlohami na príklade riešenia diplomovej alebo bakalárskej práce, ktorý je priradený každému novo zaregistrovanému používateľovi automaticky. Tento projekt má možnosť každý používateľ zmeniť vo svojich nastaveniach.

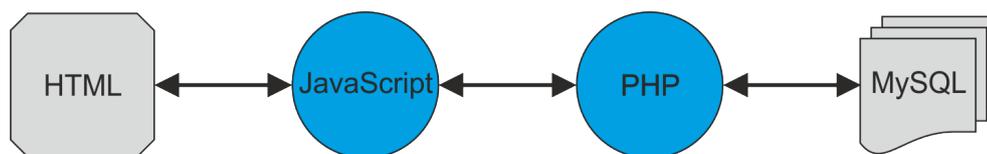
Služba musí byť otestovaná dostatočnou vzorkou používateľov. Čas určený k testovaniu reálnymi návštevníkmi slúži na odhalenie chýb spôsobených autorskou "slepotou". Je vhodné umiestniť do aplikácie niektorý z dostupných sledovacích nástrojov pre zber štatistík a odhalenie správania sa ľudí na jednotlivých stránkach. Takýmto postupom je možné odhaliť to, či návštevník hľadá zamýšľanú funkčnosť na správnom mieste a či jeho pohyb po stránkach nie je príliš zmätený. Táto časť vývoju projektu je jedna z najdôležitejších, preto je dôležité čím skôr ponúknuť k testovaniu aspoň základnú verziu aplikácie, aj keď len s obmedzenou funkčnosťou. Počas tohto obdobia je možné na projekte ďalej pracovať a vytvárať novšie verzie, ktoré budú kvalitnejšie od tých predchádzajúcich.

Ku koncu celej vývojovej, aj testovacej fázy by mal byť projekt pripravený na reálne používanie a zaťaženie. Táto bakalárska práca však uvažuje len o základnej verzii aplikácie ako pomôcky pri vytváraní dlhodobých projektov. Preto by ju bolo nutné prispôbiť a rozšíriť pred použitím za hranicou akademickej pôdy.

Kapitola 3

Návrh implementácie

Na základe cieľových vlastností služby z predchádzajúcej kapitoly bola implementácia tejto aplikácie rozdelená na klientsku a serverovú časť. Prevažná časť implementácie a celková logika sa nachádza na strane serveru. Ten má za úlohu získavať údaje, transformovať a upraviť ich do potrebnej podoby tak, aby s nimi mala klientská časť čo najmenej práce. Jej podstatou je potom zobraziť tieto dáta vo výslednej, používateľovi zrozumiteľnej podobe. Server ďalej k získaniu dát využíva komunikáciu s MySQL databázou, ktorá udržiava všetky informácie o používateľoch a nimi uložené údaje. Celá architektúra aplikácie a spôsob komunikácie jednotlivých komponent je znázornená na obrázku 3.1



Obrázok 3.1: Komunikácia komponent aplikácie

Tento princíp komunikácie a celková stavba jednotlivých komponent bola zvolená vzhľadom na nutnosť použiť JavaScript kvôli animáciám, dynamickému prekresľovaniu obsahu stránok a zároveň schopnosti komunikácie s databázou pomocou SQL požiadavkov, ktoré sa vytvárajú v PHP. Takýto postup vyplýva z riešenia väčšiny moderných webových aplikácií, i keď forma serveru sa v mnohých ohľadoch mení a jazyk PHP sa pre tento účel už takmer vôbec nepoužíva.

3.1 HTML - Bootstrap

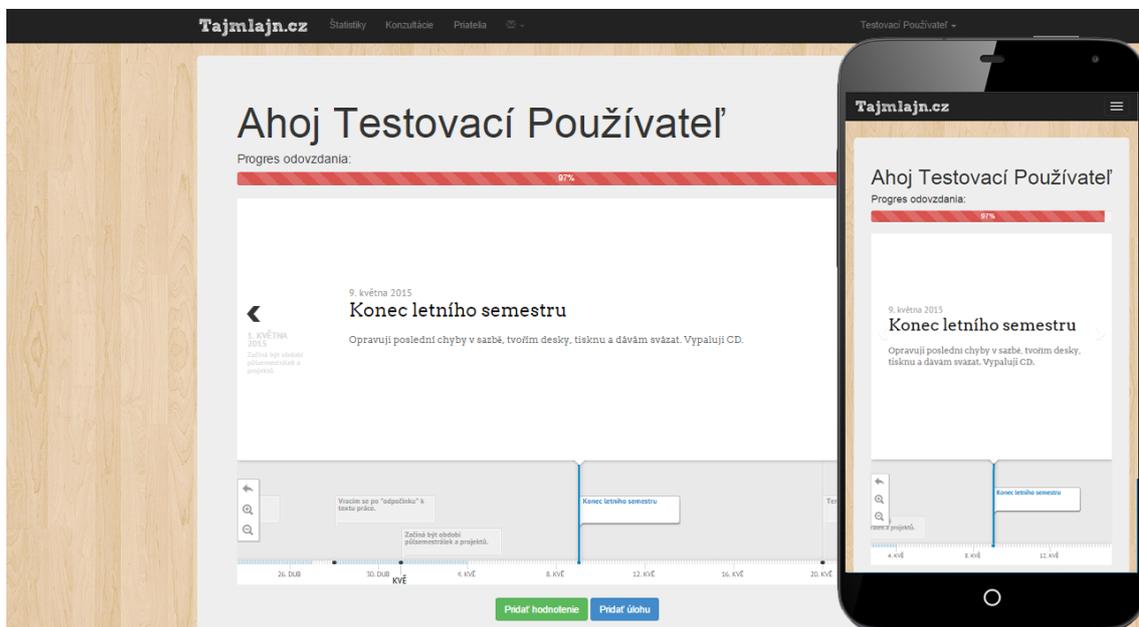
Vizuálna časť je implementovaná pomocou knižnice Bootstrap v3.3.4. Obsahuje väčšinu potrebných objektov a štýlov, ktoré boli plánované a využité v projekte. Tento framework dodáva vzhľad modernej aplikácie a zároveň napomáha k lepšej orientácii a prehľadnosti na stránkach vďaka jeho rozmanitej knižnici prvkov. Je veľmi jednoduchý a voľne dostupný súbor nástrojov pre vytváranie moderného webu a webových aplikácií. Ponúka podporu najrôznejších webových technológií HTML, CSS, JavaScript a mnoho prvkov, ktoré je možné jednoducho implementovať do stránky.

Responzívny dizajn je schopnosť webovej stránky prispôbiť sa šírke obrazovky, v ktorej sa nachádza. Knižnica Bootstrap napomáha pri vytváraní takéhoto typu stránok metódou *mobile first*.

“Je to spôsob vytvárania webových stránok a webových aplikácií, ktorých dizajn je určený pre používateľov, ktorí využívajú mobilné zariadenia a ukazuje, ako títo používatelia interagujú so stránkou. Vyžaduje, aby ste si najprv uvedomili obmedzenia a schopnosti mobilného zariadenia a slovo “najprv” (first) v tomto kontexte znamená považovať mobilné zariadenie za dôležitejšie než nemobilné.” [1]

Hlavným stavebným prvkom je mriežka, ktorá sa v závislosti na šírke displeja skladá do seba a mení šírku a štruktúru elementov. Okrem mriežky je hlavnou zmenou v zobrazení aj hlavné menu, ktoré sa vysúva z hornej časti obrazovky a jeho ovládacie prvky sú väčšie ako pri klasickom zobrazení. Zvýši sa tak pravdepodobnosť, že používateľ trafi svojim prstom alebo dotykovým perom zamýšľaný ovládací prvok na prvýkrát. Touto funkcionalitou je teda možné čiastočne nahradiť mobilnú aplikáciu kvôli prezeraniu webu pomocou smartfónov. Jedná sa o dotykové zariadenia, preto aj štýl ovládania je prispôbovaný dotyku prstom. V tomto projekte sa jedná napríklad o posúvanie časových úsekov na osi.

Na nasledujúcom obrázku 3.2 je možné vidieť vzhľad a rozloženie domovskej stránky a rozdiel medzi zobrazením na veľkej obrazovke a v mobilnom zariadení.



Obrázok 3.2: Domovská stránka a jej responzívny dizajn

3.2 CSS

Knižnica Bootstrap síce ponúka predpripravené šablóny a uľahčuje prácu, ale finálny vzhľad stránok je potrebné pre vlastné účely prispôbiť. Kaskádové štýly, anglicky Cascading Style Sheets predstavujú spôsob, ako ovplyvňovať prezentáciu dokumentu alebo kolekcie dokumentov. CSS umožňuje docieľiť prepracovaný vzhľad dokumentov. Hlavným cieľom tohto jazyka je oddelenie formátovania od obsahu dokumentu čo umožňuje lepšiu údržbu webovej aplikácie. [3] Použitie kaskádových štýlov je dnes bežnou záležitosťou každej webovej stránky alebo aplikácie a rovnako je to aj v tomto projekte. Na niektorých miestach sa v tomto projekte využíva aj najnovší štandard CSS3, kvôli vykresleniu gradientu pomocou kaskádových štýlov, transformácii elementov, vytvoreniu animácie a rozmazanie pozadia

pre zablokovanie ovládania počas vykonávania niektorých operácií.

3.3 JavaScript - klient

Klientská strana tvorí spojenie medzi užívateľským rozhraním a serverom. Jej úlohou je napĺňanie a prekresľovanie vizuálnej stránky aplikácie, ovládanie jednotlivých objektov a formulárov z užívateľského rozhrania, odosielanie požiadavok na server, prijímanie odpovedí a ich spracovanie. K realizácii funkčnosti a dynamického prekresľovania stránky bol využitý *AngularJS*. Ostatné operácie ako napríklad animácie, práca s reťazcami a poľami sú realizované pomocou knižnice *jQuery*, ktorá s *AngularJS* efektívne spolupracuje. Hašovanie hesla je realizované pomocou hašovacej funkcie MD5, ktorá aspoň čiastočne ochraňuje heslá, ktoré sú odosielané na server. Všetky dáta medzi klientom a serverom sú preposielané vo forme objektu notácie JSON. Ďalšou použitou JavaScriptovou knižnicou v projekte je *Timeline JS*¹ pre zobrazenie časovej osi. Táto knižnica ponúka dve rôzne možnosti získavania údajov k zobrazeniu. Jednou je predpripravená šablóna cez *Dokumenty Google* a druhou, ktorá je využitá aj v tomto projekte, je súbor typu JSON. Táto varianta bola vybraná preto, že súbor je možné jednoducho meniť alebo vytvárať nové šablóny programovo priamo v službe. *Timeline JS* sa automaticky prispôbuje šírke obrazovky a pri menších zariadeniach podporuje posunutie prstom namiesto klikania, čo uľahčuje jeho používanie.

3.3.1 AngularJS

“AngularJS je štrukturovaný framework pre dynamické webové aplikácie. Dovoľuje používať HTML ako šablónu a zároveň rozšíriť HTML syntax pre vyjadrenie komponentov aplikácie jasne a stručne. Eliminuje veľké množstvo kódu, ktorý sa v súčasnosti píše prostredníctvom dátovej väzby (data-binding) a vkladania závislostí (dependency injection). Celé sa to deje pomocou JavaScriptu a v rámci prehliadača, čo z neho vytvára ideálneho partnera pre hociaký server.” [2]

Táto knižnica zaisťuje funkčnosť jednostránkovej aplikácie čo znamená, že ku každému pohľadu je pridelená cesta, kontrolér a šablóna (písaná v jazyku HTML), ktoré sú prekresľované v závislosti na zmene ciest užívateľom. Každý kontrolér potom obsluhuje danú šablónu a v nej jednotlivé objekty. Nemusí sa tak prekresľovať celá stránka, ale len jej premenlivý obsah — hlavné menu a pätička ostávajú nezmenené počas celej doby behu aplikácie.

Požiadavky smerom k PHP serveru a prijímané odpovede sú realizované pomocou funkcie *\$http()*. V projekte sú definované 2 typy požiadavkov smerom na server: GET a POST. Každá požiadavka je odosielaná na samostatnú adresu serveru, aby ich pomocou týchto adries vedel zatriediť a požiadavky obslúžiť. Nasledujúci úsek kódu ukazuje spôsob posielania požiadavky POST za účelom prihlásenia. Dáta s odpoveďou alebo chybovou hláškou sú uložené v premennej *data* a návratový kód v premennej *status*.

¹Dostupné na <http://www.timeline.knightlab.com/>

```

$http({
  {
    method: 'POST',
    url: 'index.php/signin',
    data: JSON.stringify(
      {
        'email' : "email@email.sk",
        'password' : "heslo"
      }
    ),
    headers: {
      'Content-Type' : 'application/json'
    }
  }
}).success(function(data, status) {

}).
error(function(data, status) {

});

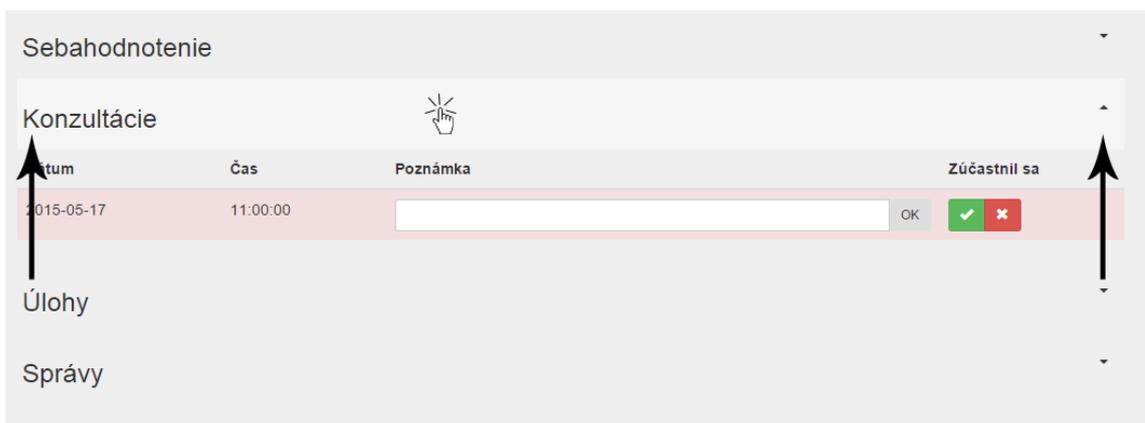
```

3.3.2 jQuery

“jQuery je rýchla, malá a na vlastnosti bohatá JavaScriptová knižnica. Vykonáva operácie ako je prechod v HTML dokumente, manipulácie, spracovanie udalostí, animácie a Ajax oveľa jednoduchšie s ľahko použiteľným API (Application programming interface), ktoré funguje v celej rade prehliadačov.” [4]

Táto knižnica je v projekte zahrnutá kvôli jednoduchosti používania animácií a oproti čistému JavaScriptu rozšírenú a pohodlnejšiu prácu s elementami HTML stránky.

V tomto projekte sa využíva napríklad na rozbaľovanie a zbaľovanie prvkov stránky používateľského profilu. Vďaka tejto funkcii sú zobrazené všetky možnosti sekcie bez nutnosti posúvania stránky a detaily sa zobrazia až na vyžiadanie návštevníka, pomocou kliknutia na požadovaný element, viď obrázok 3.3.



Obrázok 3.3: Rozbaľovací element v profile používateľa

Keďže vstupné pole typu *dátum* a *čas* nie je podporované všetkými prehliadačmi, do projektu bolo potrebné zahrnúť knižnicu, ktorá výber dátumu a času ponúka. Medzi takéto knižnice patrí aj nadstavba jQuery, jQuery UI (z angl. User Interface – používateľské rozhranie). Okrem vizuálneho zobrazenia kalendára slúžiaceho pre výber dátumu táto knižnica obsahuje kontrolu vybranej hodnoty. Nie je teda možné, aby používateľ vybral iný typ hodnoty (v tomto prípade dátum) než sa očakáva.

3.4 MD5

MD5 (message-digest Algorithm) je rozšírená hešovacia funkcia, ktorá vytvára zo vstupných dát výstup (odtlačok) fixnej dĺžky. Jeho hlavnou vlastnosťou je, že malá zmena na vstupe vedie k veľkej zmene na výstupe. Tento spôsob hešovania nezaručuje úplnú bezpečnosť aplikácie, avšak aspoň ukrýva navonok viditeľné heslá pre prípad ich zneužitia pri sieťovej komunikácii (medzi klientskou a serverovou časťou) alebo ukladaniu do lokálneho úložiska. Pri hlbšej analýze dát existuje možnosť rozšifrovať zasielané údaje rôznymi algoritmami a to z dôvodu nepoužitia bezpečnostného protokolu *https*. Avšak táto služba neobsahuje natoľko citlivé údaje používateľov, aby bolo nutné tento protokol použiť.

3.5 PHP - server

Táto časť aplikácie bola implementovaná vo frameworku *Slim*² spolu s využitím objektového programovania v jazyku PHP. Systémové požiadavky tohto frameworku nie sú vysoké - vyžaduje sa PHP s verziou minimálne 5.3.0, čo pre školský server, na ktorom bola aplikácia testovaná plne dostačuje. *Slim* prijíma požiadavky, dokáže ich rozradiť podľa typu a ďalej volať príslušné funkcie. V nasledujúcom príklade je zobrazená požiadavka so špecifickou cestou serveru a volanie funkcie *signin()*.

```
$app->post('/signin', 'signin');

function signin() {
}
```

V tele volanej funkcie je následne možné z požiadavky vytiahnuť prijaté údaje a tak s nimi ďalej vykonávať potrebné operácie. Vo väčšine prípadov sa v tomto projekte k jednoznačnému zaradeniu využíva e-mailová adresa a identifikátor relácie. Na základe týchto údajov je možné zistiť, či je používateľ prihlásený a získať pre neho potrebné informácie. Ďalšie operácie v rámci komunikácie s databázou sú spracovávané PHP knižnicou PDO, ktorá zaisťuje pripojenie k databázovému serveru, pripravenie požiadavkov a ich následné odoslanie a spracovanie odpovedí. Nakoľko odpovede z databázy sú preberané vo forme PHP objektu alebo poľa, je nutné pred odoslaním ďalej klientovi odpoveď spracovať a transformovať do notácie JSON pomocou funkcie *json_encode()*. Celá odpoveď sa vo výslednej forme zapíše pomocou funkcie *echo*.

Serverová časť – API tejto aplikácie vykonáva celú logiku a pripravuje všetky údaje, ktoré bude klientska časť vyžadovať. Ak je prijatá požiadavka závislá na súkromných užívateľských dátach, server prevedie kontrolu prihláseného konta, ktoré o toto žiada. Na základe e-mailovej adresy, ktorú obdrží a identifikátoru relácie (session) skontroluje, či daný použí-

²Dostupné na <http://www.slimframework.com/>

vateľ má oprávnenie na túto operáciu a patrične zareaguje. To ako klient zisťuje akým spôsobom má na túto situáciu reagovať bude popísané neskôr.

3.6 API

Serverová časť služby *Tajmlajn.cz* ponúka množstvo webových služieb na základe ktorých klient dokáže zobrazíť používateľovi potrebné informácie. S ohľadom na znovupoužiteľnosť a prípadné zdieľanie API s mobilnou alternatívou tejto aplikácie bola vytvorená obecná štruktúra návratových dát. V prípade, že v komunikácii medzi klientom a serverom nenastala chyba a skončila s HTTP stavovým kódom 200, server odoslal objekt typu JSON s ďalšími údajmi, o ktoré klient žiadal. Ak nastala interná chyba, server vloží do odpovede stavový kód, na základe ktorého klient dokáže správne zareagovať na danú situáciu. V niektorých prípadoch server vloží do odpovede aj presné znenie chybovej hlášky, ktorá sa zobrazí v užívateľskom rozhraní formou chybového dialógového okna. Tabuľka 3.6 zobrazuje všetky stavové kódy, ktoré sa v aplikácii vyskytujú a ich výklad. Chybové stavy obsahujú prefix "err_" a notifikácie "notf_".

Chybové kódy	Význam
err_1	nezaregistrovaný používateľ
err_2	neplatné heslo
err_3	vypršanie času relácie
err_4	Všeobecná chyba servera
Notifikácie	Význam
notf_1	úspešne predaný obsah
notf_2	registrácia / prihlásenie úspešné
notf_3	žiadna notifikácia
notf_4	profil nenájdený

Všeobecná forma odpovede tak podľa určenej špecifikácie vyzerá nasledovne:

```
{
    "resp" : {
        ...obsah odpovede...
    },
    "status" : "notf_x/err_x"
}
```

Na strane servera vzniklo niekoľko webových služieb, z ktorých každá má inú funkcionality. Každá z týchto služieb má pridelenú vlastnú cestu pre ich rozlíšenie. Požiadavky smerom na server sú zasielané na adresu podliehajúcu nasledujúcemu formátu: *http://*ADRESA_SERVERA*/index.php/CESTA_K_SLUŽBE* a sú buď typu GET alebo POST. Musia obsahovať aj potrebné údaje, aby sa stal požiadavok platným a klient tak dostal patričné informácie, o ktoré žiada. Pri vyplňovaní formulárov kontroluje neprázdnosť povinných hodnôt klientska časť a ich správnosť časť serverová. Tento postup bol zvolený kvôli šetreniu časom a množstvom dát odosielaných na server, avšak kvôli architektúre projektu a rozhodnutiu odľahčiť klientskú časť od operácií s dátami sa ich správnosť overuje až na strane serveru a klientská časť je o výsledku informovaná. V niektorých prípadoch, ako napríklad súhlasnosť e-mailovej adresy a hesla by to na strane klienta z bezpečnostných dôvodov ani nebolo možné.

Nasledujúci príklad zobrazuje údaje s ktorými klientske zariadenie žiada o prihlásenie a odpovede, ktoré môže na túto žiadosť obdržať. Z tabuľky chybových kódov vyplýva, že okrem správneho priebehu procesu prihlasovania, môže na túto požiadavku prísť chybový kód *err_1* v prípade, že daná e-mailová adresa sa nenachádza v databáze alebo *err_2* ak sa zadaná e-mailová adresa nezhoduje s heslom.

Pr.: Klient odošle požiadavku typu POST na adresu `http://*ADRESA_SERVERA*/index.php/signin` spolu s e-mailovou adresou a heslom v zahešovanej podobe. Obsah požiadavku typu POST:

```
{
    "email" : "test@test.sk",
    "password" : "81dc9bdb52d04dc20036dbd8313ed055"
}
```

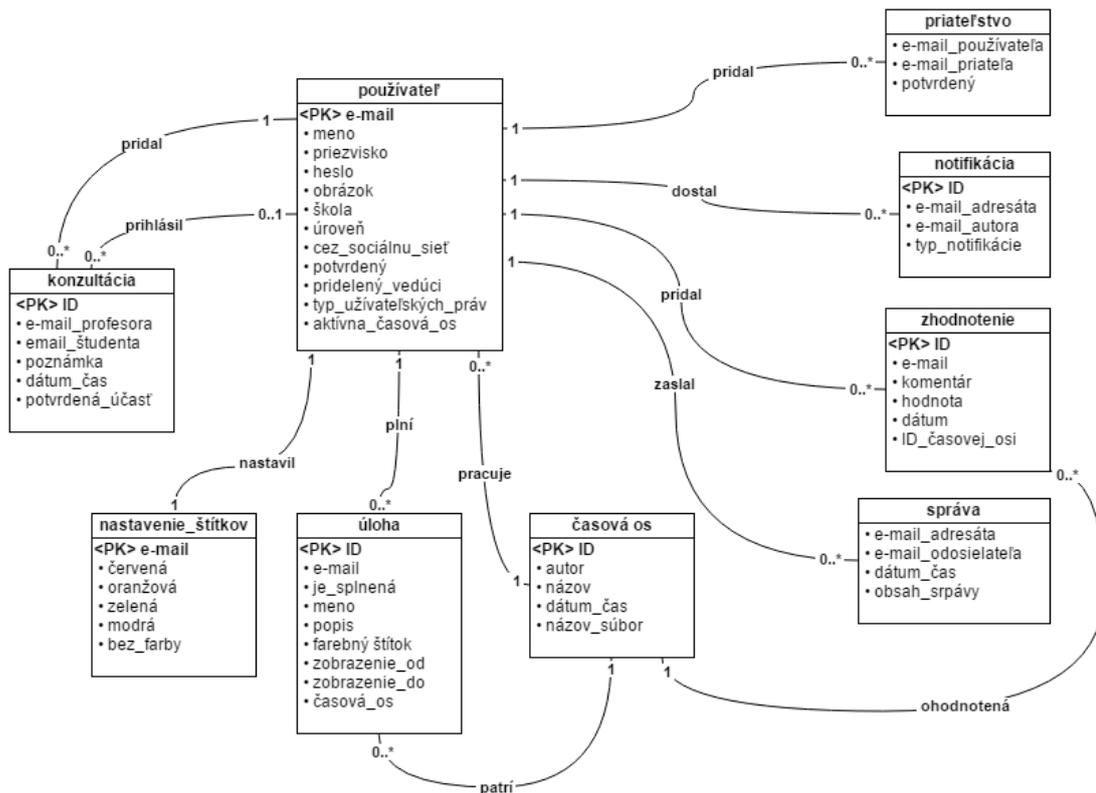
Odpoveď:

```
{
    "status" : "notf_1",
    "res" : {
        "name" : "Test",
        "surname" : "Test",
        "email" : "test@test.sk",
        "image" : "./img/photos/unknow.jpg",
        "school" : "FIT VUT Brno",
        "level" : "Profesor",
        "isSocial" : false,
        "leader" : null,
        "confirmed" : true,
        "role" : "profesor",
        "active_timeline" : "1"
    },
    "s_id" : "55570dc370944-55570dc3710d4-55570dc3718b7"
}
```

V odpovedi sa nachádza informácia o tom, s akým záverom bola ukončená – *notf_1* v tomto prípade znamená, že všetko prebehlo v poriadku a používateľ bol úspešne prihlásený do systému. Odpoveď obsahuje aj všetky základné informácie o prihlásenom používateľovi a nakoniec identifikátor relácie *s_id*. Tento identifikátor je unikátny a od momentu prihlásenia posielať s každou novou požiadavkou, aby bol server schopný zistiť, či je táto relácia ešte stále platná a môžu mu byť poskytnuté patričné informácie.

3.7 Databáza

Ako už bolo spomenuté, ukladanie a načítanie užívateľských dát sa vykonáva na strane serveru a k ich zhromažďovaniu dochádza na *MySQL* serveri. K vytvoreniu a úprave databázy bol využívaný *phpMyAdmin*. Patrí medzi najobľúbenejšie nástroje pre správu *MySQL*. Túto službu je nutné inštalovať, avšak v tomto projekte bol využitý *phpMyAdmin* na školskom serveri Eva, teda inštalácia nie je potrebná. Pred samotným procesom vytvárania databázových tabuliek bol vytvorený ER diagram, zobrazený na obrázku [3.4](#).



Obrázok 3.4: Model ER diagramu projektu Tajmlajn.cz

3.7.1 Vysvetlenie jednotlivých entít

Entita *Používateľ* zachytáva všetky nutné údaje potrebné pre prezentovanie danej osoby, jej vystupovanie v aplikácii a autorizáciu. Okrem mena a priezviska je potrebné uložiť dvojicu e-mail, heslo pre identifikáciu používateľa pri prihlásovaní. Booleovská hodnota *potvrdený* slúži ako ochrana proti používaniu cudzej e-mailovej adresy. Po potvrdení správnej e-mailovej adresy na základe návštevy vygenerovaného odkazu je táto hodnota zmenená na *true*. Nepovinnými údajmi sú *obrázok*, ak nie je zvolený, vyberie sa prednastavený, a *škola*, ktorú používateľ navštevuje alebo je jej zamestnancom. Na základe nastavenej hodnoty *aktívna_časová_os* je automaticky prezentovaný obsah daného projektu, ku ktorému sa časová os vzťahuje a na ktorom pracuje táto osoba. K tomuto projektu sa potom môžu vzťahovať aj položky v zozname úloh.

Ďalšia entita — *Priateľstvo*, znázorňuje vzťah medzi dvoma používateľmi. Jedná sa o vzťahy medzi študentami, ale priraduje aj študenta k niektorému z registrovaných profesorov. Na základe pravdivej hodnoty booleovskej premennej *potvrdený* vzniká možnosť nazerať do bližších údajov o práci spriateľného používateľa.

Notifikácie a *Správy* spolu úzko súvisia. Komunikácia formou správ je umožnená len medzi rolou študenta a priradeného profesora. Na základe odoslanej správy sa vytvorí nová notifikácia, ktorá má za úlohu upozorniť adresáta na novú správu a tak vzniká možnosť na túto skutočnosť rýchlejšie zareagovať. V prípade, že konto adresáta nie je pri odoslaní správy prihlásené, automaticky sa odošle e-mail používateľovi s textom tejto správy a odkazom, ktorý vedie do aplikácie a zobrazí celú históriu konverzácie.

Pre uloženie *časovej osi* je potrebné nastaviť jej autora, názov a cestu k súboru, v ktorom je uložená. Databáza automaticky doplní jednoznačný identifikátor a čas uloženia, kvôli zisťovaniu jej aktuálnosti.

Entita *Zhodnotenie* predstavuje ohodnotenie vykonanej práce užívateľom samým. Vzťahuje sa k aktuálnemu týždňu a k aktívnemu projektu, ktorý si tento používateľ nastavil. Prihlásená osoba zadá hodnotu na stupnici od 0 po 10 a prípadne môže udať nepovinnú poznámku k svojmu ohodnoteniu. Všetky hodnotenia je potom možné vidieť v štatistikách alebo sa zobrazujú vedúcemu projektu v profile používateľa, prípadne iným priateľom, aby mali prehľad a následne zefektívniili svoj prístup.

Úloha je položkou zoznamu úloh k vypracovaniu. Používateľ môže nastaviť jej pomenovanie a krátky popis. V rámci zoznamu úloh je možné tieto navzájom od seba odlišovať pomocou farebných štítkov, ktorého identifikátor je potrebné sem uložiť. Vzhľadom na niektoré časovo obmedzené povinnosti je možné nastaviť dátum začiatku a dátum konca zobrazovania, ktorým sa tento čas vymedzí. Ak je úloha splnená, zmení sa jej booleovská hodnota *je_splnená* na *true*. Posledným atribútom je ID časovej osi, ku ktorej sa daná úloha vzťahuje, aby bolo možné rozlíšiť úlohy jednotlivých projektov. Na obrázku 3.5 je možné vidieť správcu úloh, ktorý je implementovaný v tejto službe.

	Meno úlohy	Poznámka	Míňiky
Dôležitá	Dôležitá úloha		Farba ✓ ✕
Zelená	Napísať 10 strán práce	Napísať vedúcemu práce a získať podklady	od 2015-05-17 do 2015-05-23 Farba ✓ ✕
Modrá	Dokončená úloha		Farba ✓ ✕

Obrázok 3.5: Správca úloh

Používateľ si v rámci nastavení môže zvoliť pomenovanie jednotlivých štítkov v zozname úloh. Entita *nastavenie_štítkov* zaznamenáva tieto nastavenia u jednotlivých užívateľov. Ku každej z farieb alebo atribútu bez farby sa priradí užívateľom nastavená hodnota.

Poslednou entitou je *Konzultácia*, ktorá zachytáva termín tohto stretnutia. Vedúci projektu — profesor vypíše termín konzultácií na daný dátum a čas. Používateľ v roli študenta sa naň môže prihlásiť, prípadne mu môže byť termín pridelený profesorom. Ten má možnosť pridať k tejto udalosti poznámku a potvrdiť jeho účasť alebo nahlásiť neúčasť. Používateľ má možnosť vyfiltrovať svoje konzultácie alebo zobraziť len termíny vypísané na konkrétny dátum. Obrázok 3.6 znázorňuje skutočnú situáciu prihlásených používateľov a náhľad na správu termínov.

Vedúci projektu: Test Test

Moje

2015-05-17	Prihlásený termin: 11:00	▼
10:00		✓
10:15	admin@admin.sk	
10:30		✓
10:45		✓
11:00	testovaci.pouzivatel@mailinator.com	✘
11:15		✓
11:30		✓
11:45		✓
12:00		✓
2015-05-20	Voľné miesta 9/9	▼
2015-05-27	Voľné miesta 9/9	▼

Obrázok 3.6: Prihlasovanie sa na konzultácie a spravovanie termínov

3.8 Implementácia

Táto sekcia približuje spôsob implementácie niektorých zaujímavých komponent aplikácie.

3.8.1 Registrácia pomocou sociálnej siete

V tomto projekte je zahrnutá registrácia pomocou sociálnej siete *Google+*. Vďaka tejto funkcionalite je možné výraznou mierou skrátiť proces registrácie, ale aj prihlasovania. Pri prvom prihlasovaní musí používateľ súhlasiť s pridelením práv aplikácii, ktorá vyžaduje čítanie základných informácií o profile používateľa a zobrazenie e-mailovej adresy. Údaje sa následne prevezmú z odpovede, ktorá z *Google+* príde a to: meno, obrázok, e-mailová adresa a identifikátor účtu na sociálnej sieti. V databáze sa táto nová položka uloží s príznakom *isSocial*, aby systém dokázal rozpoznať, že pri ďalšom prihlasovaní nemusí vyžadovať heslo a k autorizácii stačí úspešná odpoveď na žiadosť o prihlásenie zo strany sociálnej siete. Pri vstupe do aplikácie pomocou *Google+* teda stačí, aby používateľ bol prihlásený v tejto sociálnej sieti a do služby *Tajmlajn.cz* sa prihlási jediným tlačidlom.

3.8.2 Udržiavanie relácie

Motivácia udržiavania relácie(sedenia) spočíva u webových aplikáciách v možnosti obnoviť stránku internetových prehliadačov. Po obnovení stránky sa vynúti jej opätovný štart a tým sa vymažú všetky uložené premenné aplikácie. Klientská časť webového portálu *Tajmlajn.cz* ukladá len základné informácie o sedení a to e-mailovú adresu, identifikátor relácie a prípadne heslo, pokiaľ používateľ zvolil pri prihlasovaní možnosť *Pamätať si ma*.

Serverová časť si ukladá zoznam všetkých relácií a sleduje, či jeden účet nie je prihlásený na viacerých zariadeniach alebo prehliadačoch. Naopak v situácii, kedy je jedno konto neaktívne viac ako 5 dní, vyprší platnosť relácie. V oboch prípadoch je používateľ s touto skutočnosťou oboznámený a aplikácia prejde na stránku s prihlasovacím formulárom. Dôjde

teda k automatickému odhláseniu. Dôvodom tejto funkcionality je zaistenie konzistencie údajov a vyhnutie sa sporným situáciám, ktoré by mohli nastať pri použití služby vo viacerých oknách. Ďalším dôvodom je zvýšenie bezpečnosti v záujme ochrany údajov alebo zneužitím účtu treťou osobou.

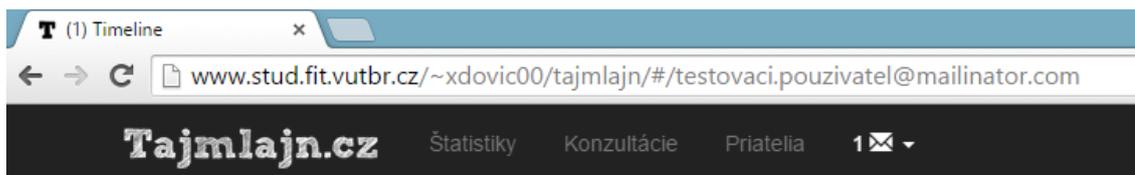
3.8.3 Hlavné menu

Funkcionalita hlavnej ponuky je riešená dynamicky. Počas štartovania aplikácie sa odošle samostatná webová požiadavka, na ktorú serverová časť služby odošle naspäť položky hlavného menu, ktoré určí podľa typu prihláseného účtu. Je teda zrejmé, že účet typu študent bude mať iné možnosti ako profesor alebo administrátor. Ak sa aplikácia nachádza v stave, kedy nie je prihlásený žiadny používateľ, menu ostáva schované.

3.8.4 Funkcionalita notifikácií

Celkovo aplikácia obsahuje 2 druhy notifikácií a 4 udalosti na ktoré má upozorňovať a ktoré sa aktualizujú automaticky v krátkom časovom úseku. Ak vznikla nová udalosť, server odošle popis a obsah notifikácie naspäť ku klientovi. Zo strany klienta nie je implementovaná žiadna zvláštnosť. Odošle sa požiadavka o nové notifikácie a v prípade, že celá sieťová komunikácia prebehla v poriadku zobrazí sa obsah — v prípade, že nejaký prišiel — a odošle rovnakú požiadavku znovu. Tento proces sa cyklicky opakuje počas celej doby behu aplikácie v prípade, že je používateľ prihlásený. Po rozkliknutí notifikácie pomocou položky v hlavnom menu sa prejde na stránku, na ktorú upozorňuje táto notifikácia a odošle sa žiadosť o jej vymazanie.

Prvým typom upozornení je krátke oznámenie, ktoré je možné vidieť v hlavnom menu ako to znázorňuje obrázok 3.7 a druhým typom dialógové okno, ktoré zobrazuje dôležité upozornenie, na ktoré by používateľ nemal zabudnúť alebo ho prehliadnúť. V takomto okne sa zobrazuje napríklad upozornenie na nevyplnené hodnotenie v aktuálnom týždni, ktorý čoskoro uplynie a ďalej k nemu nie je možné sa vrátiť. Okrem hlavnej ponuky sa počet nových notifikácií zobrazuje aj v titulke karty prehliadača. Používateľ je takto oboznámený s novou udalosťou aj keď momentálne nepracuje s aplikáciou.



Obrázok 3.7: Upozornenie na 1 novú notifikáciu

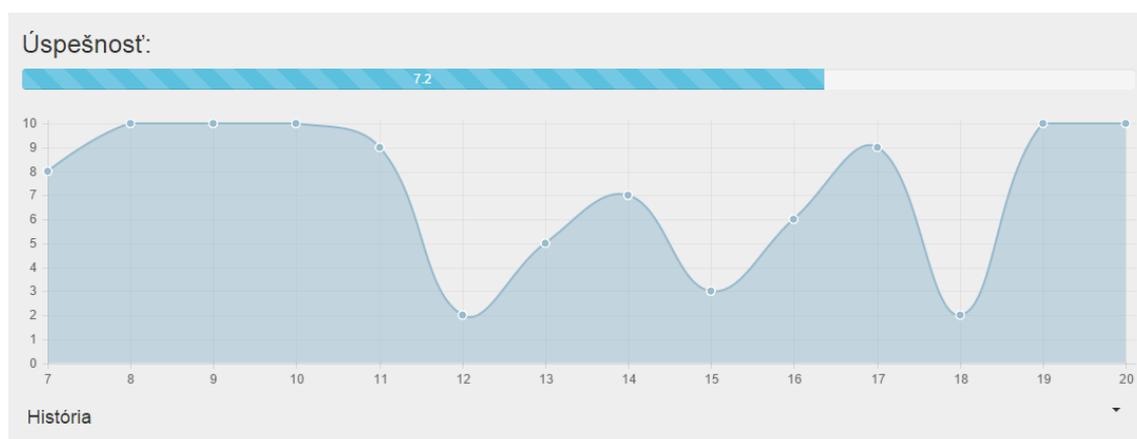
Na strane serveru sa vykonáva celá logika upozornení. V momente ako serverová časť obdrží požiadavku, pozrie sa do databázy a zistí, či sa v nej nachádzajú nové udalosti pre prihlásenú osobu. Ak nie, počká 1 sekundu a vyhľadáva v databáze znova. Tento proces sa opakuje pokiaľ systém nenájde novú notifikáciu, avšak maximálne 5krát kvôli možnému vypršaniu limitu *http* požiadavku. Maximálna doba čakania na odpoveď teda zodpovedá približne 5 sekundám. Naopak minimálna doba činí približne 1 sekundu v prípade, že sa nová udalosť nájde už v prvom cykle. Pokiaľ nebola nájdená žiadna notifikácia, odpoveď pozostáva len z prázdneho JSON objektu kvôli šetreniu prenesených dát.

Typy notifikácií v aplikácii:

- správa: Oznámenie o obdržaní novej správy
- vzťah: Oznámenie o novej žiadosti o priateľstvo
- vymazaná konzultácia: Oznámenie o vymazaní niektorého z termínov, na ktorý bol študent prihlásený
- úloha: Oznámenie o novej úlohe pridanej vedúcim projektu

3.8.5 Zobrazovanie grafov

Knižnica *ChartJS*³ ponúka vykreslenie grafov, ktoré je v tejto službe využité pre zobrazenie štatistík, aby používateľ získal lepší prehľad o svojej snahe v závislosti na čase. Táto knižnica opäť ponúka zobrazenie, ktoré prispôbujuje svoj vzhľad šírke obrazovky alebo elementu, v ktorom je umiestnená. Aj práve preto bol zvolený práve tento doplnok. Príklad grafu priebehu sebahodnotení na obrázku 3.8 ukazuje hodnotu ohodnotenia od 0–10 v závislosti na časovom úseku, v tomto prípade v rozmedzí 7.–20. týždňa.



Obrázok 3.8: Graf priebehu sebahodnotení používateľa

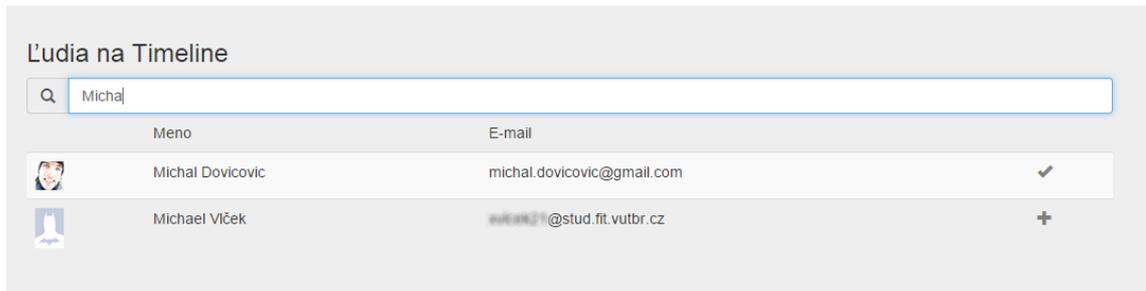
3.8.6 Filtrovanie položiek zoznamu

Portál *Tajmlajn.cz* obsahuje niekoľko zoznamov, ktoré môžu obsahovať množstvo položiek. Medzi ne patrí napríklad zoznam používateľov v sieti alebo zoznam termínov konzultácií. Pre lepšiu orientáciu v nich bolo potrebné použiť filtrovanie položiek. Zoznam konzultácií ponúka filtrovanie položiek na základe dátumu, ale je schopný zobrazíť aj termíny, na ktoré je priradený prihlásený používateľ. Zoznam ľudí v sieti obsahuje textové pole, pomocou ktorého je možné filtrovať položky na základe mena alebo e-mailovej adresy. Vďaka použitiu JavaScriptového frameworku *AngularJS* je práca s filtrami zjednodušená. Nasledujúci príklad ukazuje použitie filtru v praxi. *ng-repeat* slúži na vykreslenie rovnakých položiek v zozname, pričom údaje berie z poľa objektov *members*. Tento príklad by vypísal zoznam mien členov, ktoré by bolo možné vyfiltrovať na základe hodnoty uloženej v premennej *filterQuery*. *AngularJS* automaticky po zmene tejto hodnoty aktualizuje a prekreslí zoznam položiek.

³Dostupné na <http://www.chartjs.org/>

```
<ul ng-repeat="member in members | filter:filterQuery">
  <li>{{member.name}}</li>
</ul>
```

Príklad dôsledku použitia filtru v zozname používateľov v sieti na obrázku 3.9 zobrazuje zhodu vyhľadávaného výrazu u dvoch položiek.



Obrázok 3.9: Filtrovanie položiek zoznamu na základe mena

Kapitola 4

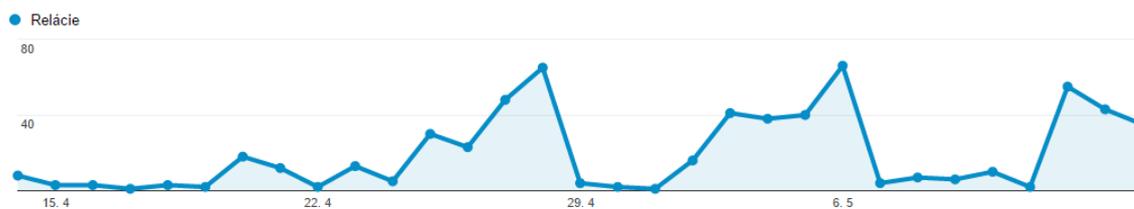
Testovanie

Samotnému vydaniu aplikácie pre použitie reálnymi používateľmi predchádzalo testovanie a s ním spojená oprava chýb. Najskôr sa jednalo o skúšanie aplikácie v úzkom kruhu vybraných osôb, ktoré s danou tematikou nemajú nič spoločné. Osloveným ľuďom bola aplikácia predstavená len zbežne a ich úlohou bolo vypracovať niekoľko jednoduchých postupov bez vysvetlenia navigácie na stránkach. Na základe týchto experimentov bola aplikácia ďalej prispôbovaná.

Veľmi dôležitou fázou vývoja služby *Tajmlajn.cz* bolo opravovanie chýb, ktoré vznikli počas vývoja aplikácie a rovnako aj tých, ktoré sa vyskytujú len v určitých prípadoch. Práve kvôli tomuto prípadu bol do aplikácie zavedený kontaktný formulár. Ten slúži pre nahlásenie chyby objavenej priamo používateľom. Tento proces sa osvedčil najmä počas paralelného vývoja spolu s používaním aplikácie. Takto postupne vznikali verzie a aktualizácie, ktoré nahradzovali tie aktuálne používané.

Počas testovacej fázy boli využité aj dva sledovacie nástroje. *Google Analytics* sleduje množstvo návštev na webovej doméne a na rôznych podstránkach aplikácie. Tieto fakty zdanlivo nie sú potrebné k samotnému vývoju, avšak vďaka nim je možné hodnotiť aplikáciu ako celok. Programátor má možnosť vidieť ktoré stránky sú najnavštevovanejšie a tým pádom aj potrebné a naopak nenavštevované stránky, ktoré sú možno redundantné. Obsah, či funkcionality takýchto stránok je tak buď nepotrebná a odstránená alebo presunutá a zlúčená s inou sekciou. V tejto aplikácii boli po dlhodobejšom sledovaní presunuté napríklad štatistiky z hodnotení vykonanej práce do osobitnej sekcie, aby pri spustení hlavnej stránky nebol divák presýtný množstvom informácií. Na druhej strane, z pohľadu vedúceho projektu, sa štatistiky presunuli do profilu študenta, avšak do rozbaľovacieho elementu, aby bola stránka profilu prehľadnejšia.

Celkovo bola stránka od vydania na testovacie účely 25.2.2015 zobrazená doposiaľ v 505 reláciách. Graf 4.1 zobrazuje rozloženie návštevnosti počas jednotlivých dní od 14.4.2015 do 14.5.2015.



Obrázok 4.1: Graf návštevnosti používateľov

Z informácií a štatistík tohto sledovacieho nástroja je možné vyčítať koľkokrát bola daná stránka zobrazená, aká je priemerná doba trvania relácie alebo to, z akého prehliadača a krajiny daná osoba pristupuje. Z analyzovaných grafov vyplýva, že rozloženie zobrazení stránok je vyvážené a ani jedna zo sekcií v tejto aplikácii nemá takú malú návštevnosť, aby sa dalo uvažovať o jej zrušení alebo presunutí jej funkcionality na iné miesto. Avšak z týchto údajov nie je možné vyčítať detailné správanie návštevníkov na stránkach.

Ďalším sledovacím prostriedkom použitým v tomto projekte je *Mouseflow.com*. Ten umožňuje sledovať priamo správanie návštevníkov stránok. Jednotlivé návštevy nahráva vo forme videa a ukladá na server, aby ich bolo možné prezeráť a analyzovať. Prínos tohto nástroja je hlavne v tom, že je možné odhaliť zmätené správanie používateľov v aplikácii. Na základe nahratých návštev boli v službe *Tajmlajn.cz* doladené polohy tlačítok a do istej miery aj rozdelenie a návrh jednotlivých podstránok. Portál *Mouseflow.com* ponúka v základnej, neplatenej verzii 100 nahrávok s úložnou dobou maximálne 1 mesiac. Problém tohto sledovacieho nástroja však spočíva v tom, že aplikácia *Tajmlajn.cz* je jednostránková aplikácie a nahrávky neobsahujú meniaci sa obsah, len jej základné zobrazenie (pozadie a hlavnú ponuku). Po hlbšej analýze sa zistilo, že tomuto problému sa dá vyhnúť len povolením podpory relácii – Session support. Táto funkcia ale v bezplatnej verzii povolená nie je. Prvotné analýzy tak boli vykonávané len úpravou videí tak, že pod nahrávku bol vložený obrázok ako náhľad stránky, aby bolo možné aspoň približne odhadnúť polohu klikov a pohybov na stránkach. Tento proces bol ale dosť zdĺhavý a nepresný. Ďalšou nevýhodou bolo spomaľovanie načítavania stránok, ktoré táto služba spôsobovala. Preto bol *Mouseflow.com* nahradený v závere testovania nástrojom *Inspectlet*, ktorý má síce podobné podmienky v základnom balíku, ale nemá problémy so zobrazovaním náhľadov jednostránkových aplikácií. Týmto sa celý problém vyriešil a rozbor návštev značne uľahčil. Rovnako sa vyriešil aj problém s pomalým načítavaním. V niektorých prípadoch bolo zaznamenané zrýchlenie dokonca o takmer 5 sekúnd bez ohľadu na zložitosť a objem prenesených dát súvisiacich s vykreslením danej sekcie.

Z analýzy videí od reálnych používateľov nebolo zistené takmer žiadne zmätené správanie. Všetky polohy elementov a rozloženie sekcií bolo doladené už na základe konzultácií počas vývojovej fázy. Výsledok rozboru videí však nezaručuje úplnú správnosť a aplikácia môže takéto neželané správanie u nových používateľov vyvolávať. Tento fakt môže byť spôsobený nepresnou analýzou počas používania nástroja *Mouseflow.com*, pričom z nových rozborov nemusí byť zreteľný, keďže užívatelia si na toto riešenie už zvykli. Jedným z podnetov, ktorý som obdržal zo strany testujúcich študentov bolo prídanie funkcionality obnovenia hesla, na ktoré sa počas vývoja zabudlo, aj keď zmena hesla je implementovaná a umiestnená v nastaveniach účtu. Pri strate hesla sa sem však používateľ nedostane.

Ako už bolo uvedené, webová služba *Tajmlajn.cz* vykazovala problémy s neprímerane dlhým načítavaním stránky. Tieto problémy spôsobovala aj knižnica použitá pre zobrazenie časovej osi. Jej načítanie bolo preto posunuté až na koniec hlavného dokumentu *index.html*, aby sa stránka načítala rýchlejšie aj za cenu pozdržania vykreslenia časovej osi.

4.1 Dosiahnuté ciele a budúcnosť aplikácie

Stanovené ciele popísané v kapitole 2 sa podarilo vo väčšej miere dosiahnuť a všetky tieto body sú zahrnuté v projekte, no niektoré prispôbené pre použitie špeciálne k projektu diplomových prác. Aktuálna verzia aplikácie je schopná viesť študentov k úspešnému vypracovaniu ich diplomovej práce alebo do istej miery k vlastnému dlhodobému projektu. Je preto potrebné poznamenať, že táto bakalárska práca je len základnou verzou služby a

pre masové používanie je nutné ju poupraviť, prípadne doplniť. Aplikácia však do budúcnosti disponuje s veľkou perspektívou. Prvým a najväčším krokom vpred bude jej nasadenie na komerčný server — mimo školského, na ktorom bola testovaná. Do blízkej budúcnosti existuje možnosť využitia tejto služby na fakulte študentami a profesormi. Je samozrejmé, že všetko závisí od jej prezentácie a od toho, či ju ľudia príjmu a začnú používať. V prípade, že by o túto službu bol obzvlášť veľký záujem, bolo by nutné ju z časti pozmeniť. Napríklad vyhľadávanie ľudí v sieti by malo vrátiť výsledky s používateľmi len z blízkeho okolia, kvôli zobrazovanému množstvu položiek. Rovnako by sa musela zvýšiť bezpečnosť aplikácie a prípadne použiť protokol *https*.

Jedným z hlavných bodov návrhu bolo obecné API, ktoré je pripravené na pripojenie ďalšieho typu zariadení, napríklad k mobilnej aplikácii spustiteľnej na platforme Android alebo iOS. Vzhľadom na veľký záber týchto dvoch platforiem na trhu by sa prístup k údajom stal jednoduchším a prakticky dostupným odkiaľkoľvek s pripojením k internetovej sieti. Ak by sa služba *Tajmlajn.cz* rozšírila k stálemu používaniu, vytvorenie mobilnej aplikácie by bol jeden z ďalších a najhlavnejších cieľov do budúcnosti.

Kapitola 5

Záver

Zadaním tejto práce bolo zoznámiť sa s JavaScriptovými frameworkami, navrhnúť a implementovať základnú verziu služby pre plánovanie dosiahnutia cieľa. Klientska časť aplikácie je implementovaná v jazyku JavaScript, avšak kvôli obecnému API táto práca využíva aj jazyk PHP. Výslednú aplikáciu bolo potrebné otestovať s dostatočnou vzorkou používateľov.

V súčasnosti sa táto webová služba nachádza na adrese <http://www.stud.fit.vutbr.cz/~xdovic00/tajmlajn/>, ale pre účely verejného využitia by musela byť presunutá na inú doménu, čo zatiaľ nebolo nutné.

Práca obsahuje kompletný servis pre použitie pri vývoji diplomových prác, s možnosťou rozšírenia o ďalšie zaujímavé, dlhodobé projekty. Obsahuje náhľad na celý projekt, predpripravené úlohy, správu konzultácií a ďalšie funkcie. Využitie na mobilných zariadeniach je možné zatiaľ len v prehliadačoch vďaka responzívnemu dizajnu, avšak služba ponúka možnosť napojenia mobilných aplikácií vďaka obecnému API.

Služba bola testovaná najmä používateľmi, ktorí pracovali na bakalárskych alebo diplomových prácach, vďaka ktorým bolo objavených a následne opravených niekoľko chýb alebo pridaná funkcionálnosť. Ich návštevnosť však klesla spolu s odovzdaním týchto projektov a v súčasnej dobe *Tajmlajn.cz* nemá aktívnych používateľov. Vďaka snahe vedúcejmu tejto práce, ktorý má v pláne ju využiť u svojich ďalších študentov by mohla v budúcnosti opäť získať používanosť.

Literatura

- [1] Clarissa Peterson: *Learning Responsive Web Design*. O'Reilly Media, Inc., 2014, ISBN 978-1-449-36294-2.
- [2] Google, licencované pod CC BY 3.0: AngularJS.
<http://docs.angularjs.org/guide/introduction>.
- [3] Meyer, E. A.: *Eric Meyer o CSS: kompletní průvodce*. Zoner Press, 2007, ISBN 978-80-86815-64-0.
- [4] The jQuery Foundation: jQuery. <http://jquery.com/>.

Dodatek A

Obsah CD

- adresár *src/* – zdrojový kód aplikácie
- adresár *text/* – zdrojový kód textu práce
- súbor *xdovic00.pdf* – text práce vo formáte pdf
- súbor *README* – návod k inštalácii
- súbor *poster.png* – plagát vo formáte png
- súbor *video.mp4* – plagát vo formáte mp4

Dodatek B

Plagát

Tajmlajn.cz

WEBOVÁ SLUŽBA PRO PLÁNOVÁNÍ DOSAŽENÍ CÍLE
•
WEB SERVICE FOR GOAL PLANNING

Ahoj Testovací Použivatel
Progress odovzdaná

Timeline
Časový plán vašej diplomky

TODO list

Stav	Úloha	Popis	Milky
Completed	Dělat testy	14.05.2015-12.05.2015-18	Fakta
Active	Písně emaily		Fakta

✓ **TODOs**
✓ **konzultácie**
✓ **čo som už spravil?**
✓ **ako sú na tom moji priatelia?**

Michal Dvorník