



**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

# **NÁSTROJ PRO TVORBU A SLEDOVÁNÍ VÝVOJE TECHNICKÉ ZPRÁVY**

TOOL FOR WRITING AND PROGRESS TRACKING OF TECHNICAL REPORT

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**MARTIN PAVELKA**

**VEDOUcí PRÁCE**

SUPERVISOR

**Ing. VÍTĚZSLAV BERAN, 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: **Pavelka Martin**

Obor: Informační technologie

Téma: **Nástroj pro tvorbu a sledování vývoje technické zprávy**

**Tool for Writing and Progress Tracking of Technical Report**

Kategorie: Uživatelská rozhraní

**Pokyny:**

1. Studujte postupy návrhu uživatelských rozhraní a techniky a nástroje pro tvorbu moderních uživatelských rozhraní pro web.
2. Navrhněte systém, který umožní postupnou tvorbu technické zprávy, sledování vývoje tvorby a generování výsledné zprávy. Správce systému bude mít možnost spravovat skupiny projektů a definovat požadované části dokumentu.
3. Vytvořte makety možného UI, testujte na uživateliích a vyhodnoťte. Upravte návrh UI podle výsledků testování.
4. Navržený systém implementujte s využitím moderních nástrojů a technologií pro tvorbu webových aplikací.
5. Navrhněte testy uživatelského rozhraní systému, realizujte testy na uživateliích a naměřené výsledky vyhodnoťte a diskutujte.
6. Vytvořte plakát a krátké demonstrační video reprezentující Vaše řešení.

**Literatura:**

- B. Shneiderman, C. Plaisant. *Designing the user interface: strategies for effective human-computer interaction*, Addison-Wesley Computing, ISBN-10: 0-321-53735-1, 2009
- L. Mathis. *Designed for Use: Create Usable Interfaces for Applications and the Web*, Pragmatic Bookshelf, ISBN-10: 1934356751, 2011
- L. Shklar, R. Rosen. *Web Application Architecture: Principles, Protocols and Practices*, Wiley, ISBN-10: 047051860X, 2009

Pro udělení zápočtu za první semestr je požadováno:

- Body 1, 2, 3 a částečně bod 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: **Beran Vítězslav, 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  
612 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký  
vedoucí ústavu

## Abstrakt

Táto bakalárska práca sa zaoberá návrhom, implementáciou a testovaním modernej webovej aplikácie, ktorá umožňuje priebežné písanie technickej správy a sledovanie procesu jej tvorby. Učiteľ definuje štruktúru práce, ktorú študentovi sprístupní. Študent priebežne píše správu a zaznamenajú sa štatistické údaje o jeho práci, ktoré systém zobrazuje na motiváciu a inšpiráciu.

## Abstract

The bachelor thesis describes the design, implementation and testing of modern web application which allows continual writing of technical documentation and following the whole creation process. A teacher defines the structure of the project and makes it accessible to students. The student continually writes the documentation. The process will be recorded by the system and the statistics will be displayed to students and administrators for their motivation and inspiration.

## Klíčové slová

Web, aplikácia, UX, informačný systém, dokumentácia, technická správa, študent, učiteľ, motivácia, PHP, JavaScript, L<sup>A</sup>T<sub>E</sub>X, štatistiky, DocBeat

## Keywords

Web, application, UX, information system, documentation, technical, report, student, teacher, motivation, PHP, JavaScript, L<sup>A</sup>T<sub>E</sub>X, statistics, DocBeat

## Citácia

PAVELKA, Martin. *Nástroj pro tvorbu a sledování vývoje technické zprávy*. Brno, 2016. Bakalárska práca. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Beran Vítězslav.

# Nástroj pro tvorbu a sledování vývoje technické zprávy

## Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Ing. Vítězoslava Berana, Ph.D.. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....  
Martin Pavelka  
17. mája 2016

## Podakovanie

Rád by som poďakoval pánovi Ing. Vítězoslavovi Beranovi, Ph.D. za odborné vedenie a veľmi aktívny prístup pri písaní tejto bakalárskej práce. Poskytnutých mi bolo množstvo rád a návrhov na zlepšenia, ktoré som zahrnul do výslednej aplikácie.

© Martin Pavelka, 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

<b>1 Úvod</b>	<b>3</b>
<b>2 Teoretické východiská</b>	<b>4</b>
2.1 Existujúce riešenia . . . . .	4
2.2 L <sup>A</sup> T <sub>E</sub> X . . . . .	6
2.3 UX a design zameraný na užívateľa . . . . .	6
2.4 Technológie pre tvorbu webových aplikácií . . . . .	7
<b>3 Ako podporiť písanie kvalitných technických správ?</b>	<b>12</b>
3.1 Špecifikácia požiadaviek . . . . .	12
3.2 Prípady použitia . . . . .	13
3.3 Návrh užívateľského rozhrania . . . . .	13
3.4 Správa projektov . . . . .	16
3.5 Kopírovanie verejného projektu . . . . .	20
3.6 Editácia projektu . . . . .	20
3.7 Pridávanie administrátorov . . . . .	20
3.8 Editor dokumentácie . . . . .	21
3.9 Štatistiky a grafy . . . . .	25
<b>4 Aplikácia DocBeat</b>	<b>29</b>
4.1 Implementácia pomocou CodeIgniter . . . . .	29
4.2 Datový model . . . . .	31
<b>5 Testovanie, analýza a výsledky</b>	<b>32</b>
5.1 Spôsob testovania . . . . .	32
5.2 Problémy pri testovaní . . . . .	32
5.3 Testovanie na subjektoch . . . . .	32
5.4 Zhrnutie testovania . . . . .	34
<b>6 Záver</b>	<b>35</b>
<b>Literatúra</b>	<b>36</b>
<b>Prílohy</b>	<b>37</b>
Zoznam príloh . . . . .	38
<b>A Obsah CD</b>	<b>39</b>
<b>B Inštalácia aplikácie</b>	<b>40</b>

<b>C</b>	<b>Manuál základnej práce s aplikáciou</b>	<b>41</b>
<b>D</b>	<b>Výsledná aplikácia</b>	<b>42</b>

# Kapitola 1

## Úvod

Ak by sme chceli správne pochopiť, ako bol každý produkt či aplikácia navrhnutá a ako presne vo vnútri pracuje, potrebujeme mať poruke nejakú príručku alebo technickú dokumentáciu. Bez nej by bolo pochopenie problému, o ktorom práca pojednáva skoro nemožné. Písanie a čítanie dokumentácií bolo kedysi pomerne nezáživnou činnosťou. Dnes však existujú mnohé prostriedky a postupy, vďaka ktorým je jej čítanie zaujímavé a ponúka nám prehľadne všetky znalosti, ktoré pri práci s projektom potrebujeme.

Prvky ako tabuľky, mockupy, obrázky alebo grafy sú dnes súčasťou skoro každej dokumentácie. Vďaka nim môže byť tvorba takejto dokumentácie pomerne náročná, a preto sa využívajú špeciálne typografické nástroje ako  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ , ktoré umožňujú vytvorenie pekne formátovaného dokumentu. Naučiť sa v takomto prostredí prácu správne písať však chce cvik a nejaké to štúdium navyše.

Moja bakalárska práca ponúka prostriedok, ako zmeniť nudné písanie dokumentácie, ktoré popisovali respondenti v sekcii 5.3, v lepšiu a zábavnejšiu záležitosť pre študenta, ktorý sa chce v písaní zdokonaľiť. Učiteľovi ponúka na druhú stranu možnosť žiakov motivovať a inšpirovať k jej písaniu. Rovnako môže zhodnotiť prácu svojej triedy ako celku. Pomocou týchto znalostí je možné zlepšenie učebného procesu. Cieľom mojej práce nie je žiakov demotivovať zbytočným stresom, ale vybudieť priamo v nich túžbu niečo kvalitné vytvoriť, na čo budú hrdí aj oni sami.

Ďalším problémom spojeným s písaním technickej správy je nedodržiavanie priebežnej činnosti, ktorá sa skladá z určitých overených logických postupov ako návrh, implementácia, testovanie a zhodnotenie. Študenti nad projektom začínajú uvažovať až v prípade implementácie. Moja práca ich motivuje k dodržiavaniu takéhoto procesu a tak ich vedie k lepšie navrhnutému finálnemu produktu.

Prvá kapitola stručne popisuje všetky zdroje, z ktorých som pri tvorbe práce čerpal a z ktorých moja práca vychádza. Nasleduje kapitola informujúca o spôsobe riešenia návrhu. Základné informácie o zdrojovom kóde sú prehľadne zhrnuté v predposlednej kapitole. Záver práce popisuje testovanie a sumarizuje celkové výsledky mojej práce.

## Kapitola 2

# Teoretické východiská

Začiatok kapitoly hodnotí dostupné riešenia, ktoré majú obsahovo blízko k mojej práci a z ktorých som čerpal pri návrhu inšpiráciu. Vyzdvihuje ich kladné stránky a popisuje problémy, ktoré sa mi podarilo nájsť ich testovaním. Zvyšok kapitoly popisuje teoretické informácie o jednotlivých programovacích jazykoch, projektoch, frameworkoch a knižniciach, ktoré som využíval počas implementácie výslednej aplikácie.

### 2.1 Existujúce riešenia

Pred tvorbou samotnej aplikácie som vyhľadal a otestoval všetky dostupné riešenia, aby som si vytvoril určitú predstavu o tom, ako by som mal výslednú aplikáciu navrhnuť. Zhodnotil som každé z nich a snažil sa zlúčiť jednotlivé kladné dojmy do spoločnej predstavy o mojej aplikácii. Existuje viacero riešení, ktoré ponúkajú písanie práce vo formáte  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ . Napriek tomu sa mi nepodarilo nájsť žiadny projekt, ktorý by umožňoval tvorbu užívateľskej dokumentácie, ktorá by bola moderovaná prostredníctvom administrátora. Niektoré aplikácie slúžia na tvorbu štúdijských kurzov. Žiadna z nich však nezahŕňa tvorbu technickej správy.

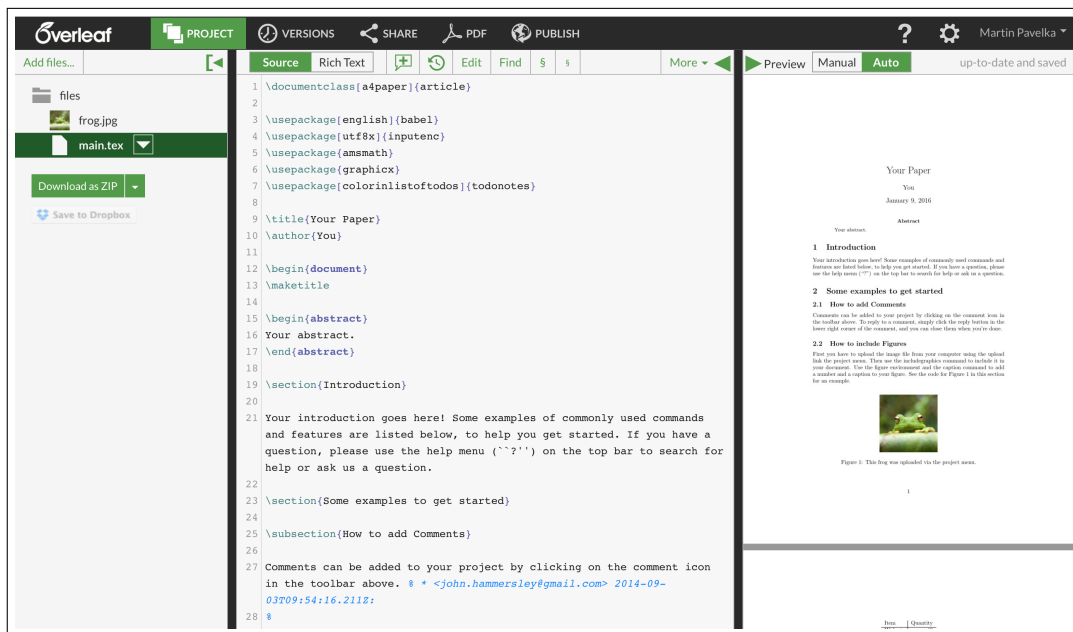
#### Overleaf<sup>1</sup>

Aplikácia patrí medzi najpopulárnejšie nástroje na tvorbu dokumentov vo formáte  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ . Rozhranie je rozdelené do štyroch základných blokov na zoznam súborov, editor kódu, panel nástrojov a náhľad výsledného PDF súboru. Panel súborov poskytuje zoznam všetkých súborov, ktoré užívateľ nahral do systému. Dostupná je aj funkcia, ktorá spočíta základné metriky ako počet slov. Funkcia častokrát nefunguje a vracia chybové hlásenia. Editor zvýrazňuje syntax a podporuje automatické vloženie konštrukcií pre zadanie tučného písma, kurzív atď.. Dostupná je aj funkcia, ktorá umožňuje zadávať už formátovaný text a následne ho prevádza do jazyka  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ . Generovanie náhľadu je možné automaticky alebo manuálne. Panel nástrojov ponúka možnosti spravovať verzie, sťahovať výsledné PDF alebo ho zdieľať prostredníctvom odkazu. Aplikácia má veľmi príjemné užívateľské rozhranie. Bez väčšieho úsilia sa mi podarilo vytvoriť základný dokument. V projekte som sa ňou inšpiroval hlavne pri správe verzií a návrhu panelu nástrojov. Ukážku tvorby projektu v tejto aplikácii zobrazuje obrázok 2.1.

---

<sup>1</sup>Stránka aplikácie Overleaf: <https://www.overleaf.com/>

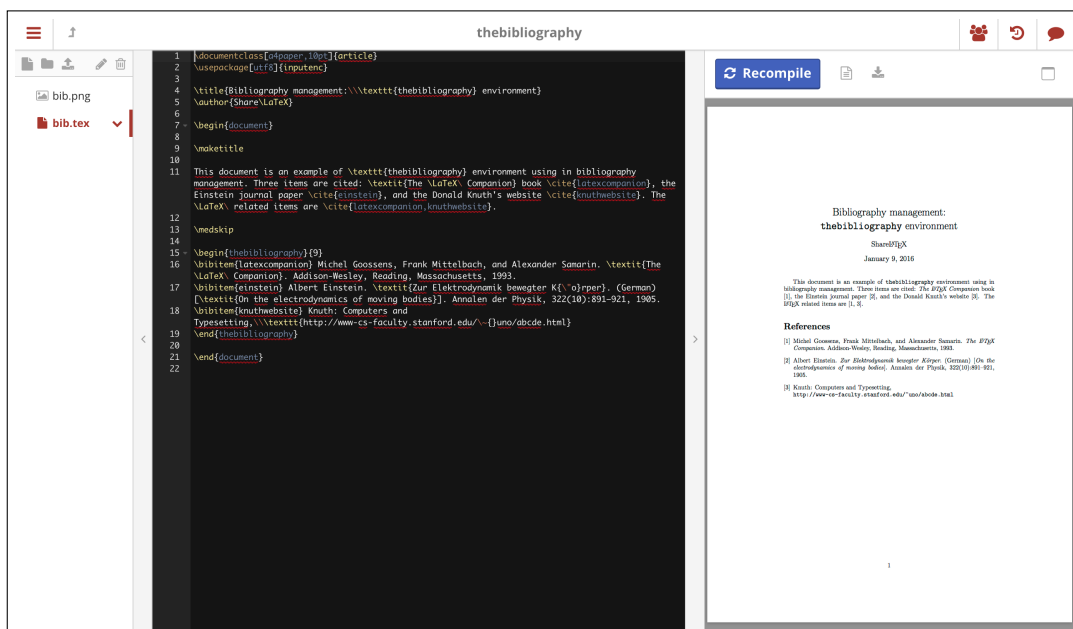




Obr. 2.1: Ukážka aplikácie Overleaf

## ShareLaTeX<sup>2</sup>

Je o niečo jednoduchší ako Overleaf a ponúka menšie množstvo nástrojov na editáciu súborov a tým zvyšuje mieru jednoduchosti pri používaní aplikácie. Design aplikácie je v peknom prevedení a editor poskytuje všetko potrebné pre písanie zdrojového textu. Rozhranie a funkcie ShareLaTeX sú veľmi podobné riešeniu od Overleaf. Užívateľské rozhranie vidno na obrázku 2.2.



Obr. 2.2: Ukážka aplikácie ShareLaTeX

<sup>2</sup>Stránka aplikácie ShareLaTeX: <https://www.sharelatex.com>

## 2.2 L<sup>A</sup>T<sub>E</sub>X

T<sub>E</sub>X je program určený pre vytváranie elektronických dokumentov. Vznikol ako reakcia autora na nesprávnu tlač matematických vzorcov v jeho skriptách. Práve v jednoduchosti ich zápisu vyniká. [2] V tomto programe je možné umiestniť ľubovoľný znak na ľubovoľné miesto v danom súbore. Z T<sub>E</sub>Xu sa nakoniec vyvinul L<sup>A</sup>T<sub>E</sub>X. Na zapisovanie myšlienok autora dnes slúži rozšírená a elegantná séria príkazov. Pre písanie samotného textu dokumentácie som zvolil jazyk L<sup>A</sup>T<sub>E</sub>X, ktorý považujem za jediné vhodné riešenie pre môj projekt aj z hľadiska rozšírenosti na fakulte.

L<sup>A</sup>T<sub>E</sub>X sa inštaluje pomocou jednotlivých programov, ktoré slúžia na preklad zdrojových súborov alebo ako hotová distribúcia, ktorá obsahuje všetky podprogramy a rozšírenia v sebe. Základná kostra súboru obsahuje definíciu začiatku dokumentu, použité balíčky, definície príkazov a základné nastavenie stránok dokumentu. Telo dokumentu tvorí zdrojový text a makrá. Špeciálnym makrom určíme koniec dokumentu. Zdrojové súbory majú príponu \*.tex. Balíčkami je možné rozširovať funkcie. Implementované sú súbormi \*.sty, ktoré sa priložia do zložky projektu. Na tvorbu bibliografických citácií slúži program BibTeX. Definície citovaných diel sú uložené v súboroch \*.bib. Tieto súbory sa vložia do zdrojového súboru a pri citovaní sa vytvorí bibliografický zápis.

### pdfTeX<sup>3</sup>

Je konzolový program, ktorý som využil na konverziu L<sup>A</sup>T<sub>E</sub>X zdrojových súborov do výstupného formátu PDF. Existuje viacero dostupných riešení. Konkrétne sa jednalo o knižnice pdfTeX, XeTeX a LuaTeX. Každú variantu som otestoval na testovacích súboroch. Podľa kvality výsledného PDF dokumentu som sa nakoniec rozhodol do svojho projektu použiť práve pdfTeX, ktorý jednoznačne najlepšie vykonával požadovanú konverziu.

## 2.3 UX a design zameraný na užívateľa

Užívateľská skúsenosť (UX), je celková skúsenosť používateľa s používaním nejakej aplikácie. Táto veličina je nemerateľná, ale jej nedostatočnosť môže viesť k obmedzeniu používania užívateľmi. Pri tvorbe užívateľského rozhrania sa využíva viacero rôznych metodík ako napríklad User centered design (UCE).

UCE sa pozerá na rozhranie z pohľadu konkrétnych osôb. Zohľadňuje ich požiadavky a spôsoby, akým budú s danou aplikáciou pracovať. Pri návrhu sa pýtame, aký užívatelia budú produkt používať, aké ciele sledujú, aké funkcie potrebujú, alebo ako si predstavujú, že aplikácia bude vo výsledku fungovať. Základnými prvkami tohto návrhu sú preto osoba, scenár a diagram prípadov použitia.

### Prístupnosť

Každá funkcia, ktorá sa vyskytuje v aplikácii musí byť užívateľovi prístupná v čo najkratšom možnom čase a tým najjednoduchším spôsobom. Všetky zobrazenia stránky by mali byť čo najviac konzistentné, aby pri prechode z jedného na iné mohol užívateľ využiť skúsenosti, ktoré nadobudol pri používaní toho predchádzajúceho.

---

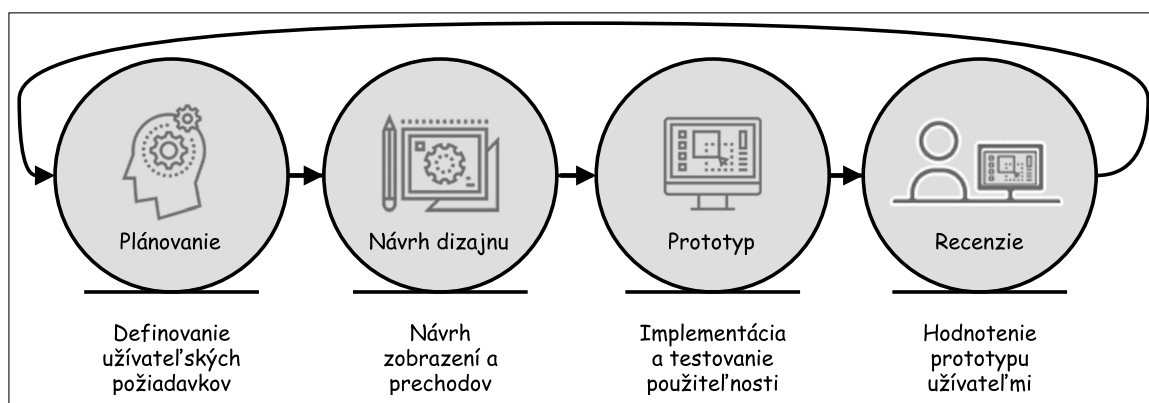
<sup>3</sup>Domovská stránka pdfTeX: <https://www.tug.org/applications/pdftex/>

## Viditeľnosť

Je faktor, ktorý hovorí o tom, ktorý element je na stránke významný a ktorý nie. Vzhľadom k tomu musia byť niektoré časti zvýraznené, aby bolo ich umiestnenie ihneď zjavné. Typicky by mali byť viditeľnejšie častejšie používané funkcie aplikácie.

## Čitateľnosť

Všetky texty by mali byť čo najlepšie čitateľné. Žiadne písmená by nemali byť príliš malé, ale ani veľké. Je napríklad známe, že kapitálky sú ťažko čitateľné. Lepšia čitateľnosť sa tiež dosiahne zvýšením kontrastu medzi pozadím a písmom.



Obr. 2.3: Spôsob návrhu aplikácie

## 2.4 Technológie pre tvorbu webových aplikácií

Počas implementácie webových aplikácií je často nutné použiť viacero programovacích jazykov, knižníc a dostupných riešení, ktoré umožnia implementovať všetky potrebné časti. Nasledujúca kapitola uvedie ich zoznam. Popisuje ich základné vlastnosti, dôvod výberu a vysvetľuje ich význam.

### PHP

Je open-source programovací jazyk, ktorý umožňuje procedurálne alebo objektové programovanie. Je vhodný na programovanie klient-server aplikácií na strane servera. Má využitie najmä pre programovanie interaktívnych dynamických stránok a aplikácií. Jednotlivé skripty sú vykonávané na strane servera a k užívateľovi je prenášaný až výsledok. [3] Využil som ho vďaka jeho dlhodobej tradícii v skriptovaní webových stránok. Ponúka množstvo funkcií, ktoré uľahčujú jeho tvorbu. Pre verziu 7 som sa rozhodol na základe jej dobrej spätnej kompatibility, pričom prináša výrazné zvýšenie výkonu generovania výsledného zobrazenia. Všetky knižnice, ktoré v projekte boli využité ponúkajú podporu tejto verzie.

### MySQL

Implementuje server databázového úložiska. Je dostupný aj pre menšie projekty vďaka jeho voľnej licencií. MySQL je nutné na cieľovom serveri nainštalovať alebo použiť iný dostupný server. Pri implementácii som využil najnovšiu verziu 5.7, ktorá ponúka výrazné zvýšenie rýchlosti oproti predchádzajúcim verziám.

## HTML

Patrí medzi značkovacie jazyky a slúži k implementácii kostry vizuálnej časti webovej stránky. V minulosti formátoval aj jednotlivé elementy, čo neskôr nahradili kaskádové štýly CSS. Prvky v kóde sa nazývajú elementy a je ich možné implementovať pomocou značiek označovaných ako tagy, ku ktorým je možné pripojiť rôzne atribúty. [5]. V aplikácii sa vykytuje v súboroch šablón, kde popisuje rozloženie prvkov na stránke, čím tvorí jej základnú kostru. Použitie HTML je nevyhnutné pri tvorbe akejkoľvek webovej stránky a tak jeho využitie bolo nutnosťou.

## CSS

Upravuje vizuálnu štylizáciu výstupu kódu HTML. Jazyk bol navrhnutý tak, aby umožnil lepšie vizuálne zobrazenie webstránky. CSS je rozšírenie a na funkcionálnosť nemá žiaden vplyv. Pomocou neho môžeme nastaviť jednotlivé farby, fonty ale aj rozloženie samotných prvkov na stránke. Ponúka centralizované štylizovanie vizuálnej časti stránky. [1]. V zdrojových súboroch sa využíva na prehľadnejšie a lepšie oddelenie kostry od jej štýlu. Jeho použitie vychádza aj z využitia frameworku Bootstrap, ktorý implementuje preddefinované štýly.

## JavaScript

Patrí medzi skriptovacie jazyky. Môže bežať buď na strane klienta alebo servera. Pomocou jazyka je možné pristupovať k rôznym prvkom stránky. Tieto prvky môžeme pomocou skriptov rôzne meniť a ovplyvňovať. Dovoľuje tiež reagovať na rôzne udalosti ako kliknutie myši alebo stlačenie klávesy, čo umožňuje spúšťanie ďalších skriptov. [4]. Pri implementácii bol JavaScript využitý na tvorbu responzívneho webu, pričom bol využitý jeho framework jQuery.

## Ajax

Slúži k tomu, aby bolo možné pomocou JavaScriptu načítať alebo poslať nejaké dáta v pozadí stránky. Iné dáta môžeme od stránky naopak prijať a ihneď ich použiť. Počas získavania dát sa stránka vôbec neobnovuje. [6]. Ajax bol v aplikácii využitý na zasielanie a prijímanie dát z externých kontrolérov. Jedná sa o asynchrónne spojenie.

## PHPMailer<sup>4</sup>

Knížnica PHPMailer nahrádza klasický spôsob posielania mailov v jazyku PHP. Ponúka rozširujúce funkcie a viac možností konfigurácie. Pretože samotný mail nebol postačujúci a potreboval som viac možností, implementuje v aplikácii zasielanie všetkých mailov.

## Bootstrap<sup>5</sup>

Patrí medzi najznámejšie štylizujúce frameworky. Obsahuje preddefinované kaskádové štýly, ktoré implementujú vzhľad jednotlivých prvkov na stránke. Namiesto tvorby vlastného štýlu, prostredníctvom neho využívam už hotové riešenia. Existujú aj iné frameworky, napríklad Materialized design. Ja som sa rozhodol pre Bootstrap vďaka skúsenostiam, rozšírenosti na webe a veľkému počtu definovaných elementov.

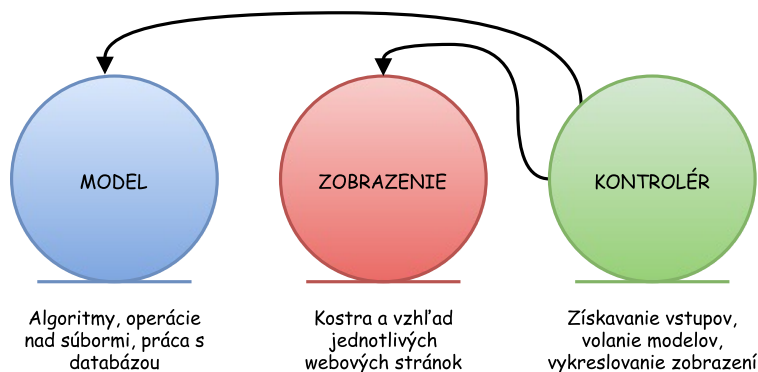
---

<sup>4</sup>Repozitár projektu PHPMailer Bootstrap: <https://github.com/PHPMailer/PHPMailer>

<sup>5</sup>Stránka projektu Bootstrap: <http://getbootstrap.com>

## CodeIgniter<sup>6</sup>

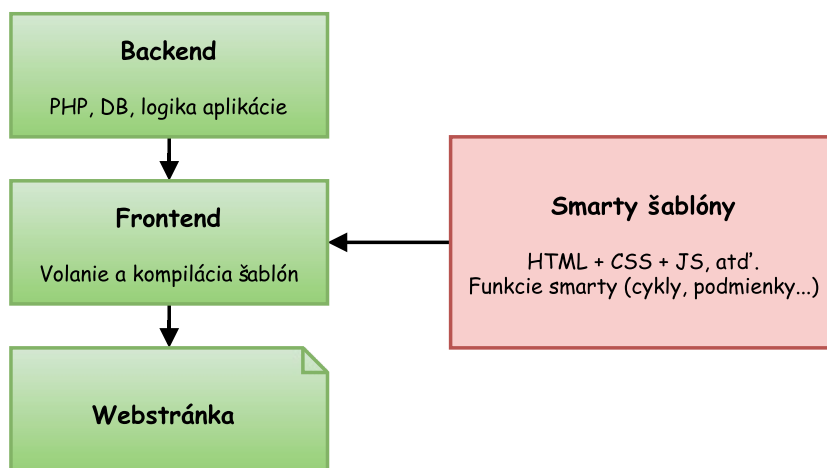
PHP frameworky obsahujú kostru back-endu webovej stránky a poskytujú množstvo hotových funkcií. Jeden z najznámejších je CodeIgniter. Výrazne zvyšuje efektivitu pri tvorbe projektu. Ponúka množstvo pomocných funkcií ako aj možnosť implementácie v návrhovom vzore MVC, ktorý popisuje obrázok 2.4. Existuje viacero PHP frameworkov, napríklad Nette a Zen. S prácou v CodeIgniteri mám už skúsenosti a podľa môjho názoru je jeden z najminimalistickejších.



Obr. 2.4: Ukážka organizácie štruktúry projektu podľa MVC

## Smarty<sup>7</sup>

Patrí medzi najpoužívanejšie šablónovacie knižnice pre aplikácie písané v jazyku PHP. Tvorba šablón je jedným zo základných prvkov moderných webových aplikácií. Pomocou nich je možné dynamicky generovať webovú stránku a vkladať do nej požadované dáta prostredníctvom PHP kódu. Z dostupných riešení, ktoré sa mi podarilo nájsť spomeniem ešte integrované Codeigniter šablónovacie riešenie. Pretože Smarty ponúka prehľadnejší výsledný kód a pokročilejšiu funkcionálnosť, rozhodol som sa ho využiť. Princíp jeho fungovania je popísaný obrázkom 2.5.



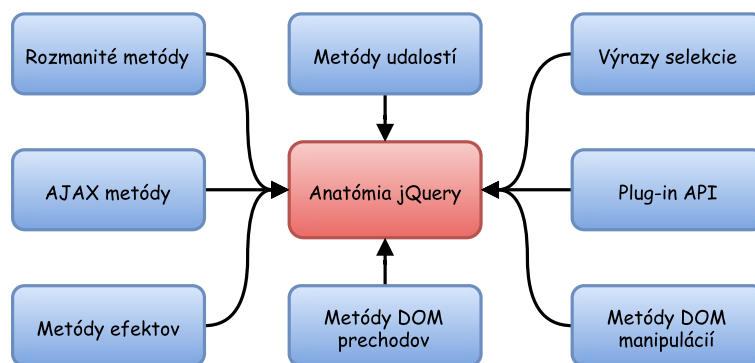
Obr. 2.5: Ukážka organizácie štruktúry projektu s použitím šablón

<sup>6</sup>Stránka projektu CodeIgniter: <https://www.codeigniter.com>

<sup>7</sup>Stránka projektu Smarty: <http://www.smarty.net>

## jQuery<sup>8</sup>

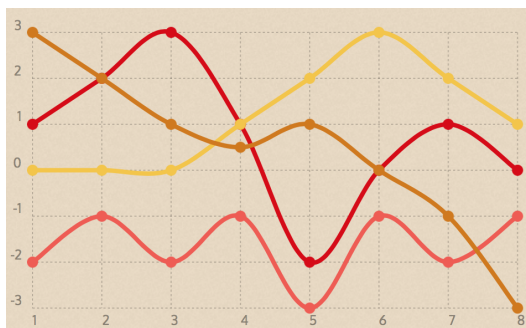
JavaScript framework od firmy Google slúžiaci na tvorbu dynamických stránok. Pomocou neho sú implementované funkcie tranzitívnosti, odosielania požiadavkov POST a GET ale aj dynamické zmeny štýlu stránky. Vďaka nemu je možné prepínať medzi zobrazeniami bez nutnosti obnovenia stránky. Anatómiu frameworku popisuje obrázok 2.6.



Obr. 2.6: Ekosystém knižnice jQuery

## Chartist<sup>9</sup>

V aplikácii sa nachádzajú grafy, ktoré sú implementované pomocou projektu Chartist. Pomáha rýchlu tvorbu pekných a responzívnych grafov s možnosťou zobrazení viacerých dátových údajov. Ukážka grafu je na obrázku 2.7.



Obr. 2.7: Ukážka grafu v Chartist

## CodeMirror<sup>10</sup> a Prettify<sup>11</sup>

Projekty slúžia na zobrazenie zdrojových kódov so zvýraznením syntaxe podľa programovacieho jazyka. CodeMirror umožňuje zadávanie zdrojového kódu so zobrazením syntakticky zvýrazneného textu. Podporuje odchytyvanie najrôznejších udalostí ako napríklad stisnutie klávesnice. Umožňuje používanie klávesových skratiek a zobrazuje čísla riadkov zdrojového kódu. Prettify zobrazuje zdrojový kód v prehľadnom elemente, tiež s podporou syntaxe.

<sup>8</sup>Domovská stránka projektu jQuery: <http://jquery.com>

<sup>9</sup>Projekt Chartist: <https://gionkunz.github.io/chartist-js/>

<sup>10</sup>Stránka CodeMirror: <https://codemirror.net>

<sup>11</sup>Stránka Prettify: <https://github.com/google/code-prettify>

## Datatables<sup>12</sup>

Na doplnenie pokročilých funkcií do obyčajných HTML tabuliek slúži projekt Datatables. Pretože aplikácia je škálovateľná pre väčší počet študentov a projektov, bolo implementované takéto riešenie jedinou variantou, ako zabezpečiť prehľadné zobrazenie jednotlivých položiek. V každej tabuľke sa nachádza pole na vyhľadávanie hodnôt, možnosť zmeniť počet zobrazených položiek a prepínanie jednotlivých stránok s položkami. Jednou z najdôležitejších funkcií je schopnosť usporiadať položky v stĺpcoch. Ukážku upravenej tabuľky zobrazuje obrázok 2.8.

Show 10 entries Search:

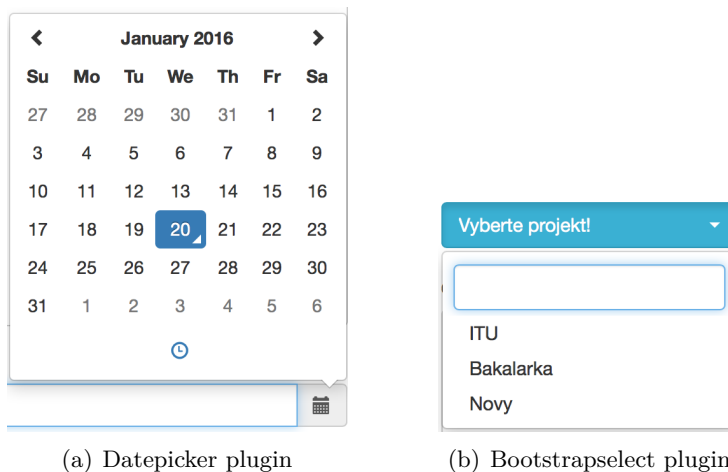
Meno	Vlastnik	Autor	Projekt	Dátum	Editovať
Dvojka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	
Jednicka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	
Stvorka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	
Trojka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	

Showing 1 to 4 of 4 entries Previous 1 Next

Obr. 2.8: Ukážka projektu Datatables

## Datepicker<sup>13</sup> a Bootstrapselect<sup>14</sup>

V samotnom CSS frameworku nie je preddefinovaný štýl pre tzv. selectbox. Tento problém som vyriešil pomocou projektu Bootstrapselect, ktorý implementuje štýl pre tento prvok. Rovnako sa mi nepáčilo riešenie prvku na zadávanie časových údajov a rozhodol som sa ho nahradiť riešením projektu DatePicker, ktoré ponúka intuitívnejšie zadávanie dátumu a času. Ukážky oboch sú zobrazené v obrázku 2.9.



Obr. 2.9: Ukážka pluginu Datatables a BootstrapSelect

<sup>12</sup>Projekt Datatables: <https://datatables.net>

<sup>13</sup>Projekt Datepicker: <http://bootstrap-datepicker.readthedocs.io/>

<sup>14</sup>Projekt Bootstrapselect: <https://silviomoreto.github.io/bootstrap-select/>

## Kapitola 3

# Ako podporiť písanie kvalitných technických správ?

Začiatok kapitoly popisuje prácu z pohľadu diagramu prípadov použitia. Následne vysvetľuje spôsob návrhu užívateľského rozhrania a implementácie jednotlivých funkcií z pohľadu žiaka a učiteľa.

### 3.1 Špecifikácia požiadaviek

Cielová aplikácia by mala obsahovať dva pohľady pre učiteľa a žiaka. Učiteľ bude môcť organizovať štruktúru technickej správy a zadávať termíny odovzdávania. Možné bude aj nastavenie limitov pre citácie a počet slov. Administrácia bude fungovať aj pre väčší počet študentov a projektov. Každý projekt bude organizovaný do menších logických celkov na kapitoly, termíny a triedy. Bude obsahovať jednu triedu a rôzny počet kapitol. Pre každú kapitolu bude zadaný jeden termín odovzdania. Vytvorené projekty budú môcť učiteľia zdieľať. Študent bude tvorcom samotnej technickej správy a bude mu dostupný editor s rôznymi funkciami. Možné bude spravovať verzie a generovať výsledné PDF. Systém bude žiakom poskytovať rôzne motivačné prvky a elementy. Zobrazené budú štatistiky a grafy o práci iných študentov. Rovnako bude pridaná možnosť nahliadnuť do prác spolužiakov, ktorí si budú môcť definovať viditeľnosť svojich prác.

Objekt	Funkcia
<b>Kapitola</b>	Tvorí základnú jednotku štruktúry projektu a obsahuje informácie o konkrétnej kapitole, ktorá je prístupná študentovi
<b>Ročník</b>	Zoskupuje študentov jednej triedy do spoločného objektu ročníka
<b>Inštancia</b>	Jej obsah informuje o čase, kedy začne práca na projekte
<b>Projekt</b>	Hlavný objekt zoskupujúci kapitoly, termíny, triedu a inštanciu
<b>Užívateľ</b>	Definuje účet žiaka alebo administrátora s prístupovými údajmi
<b>Termín</b>	Určuje dátum a čas očakávaného dokončenia práce na konkrétnej kapitole

Tabuľka 3.1: Zoznam logických objektov



## 3.2 Prípady použitia

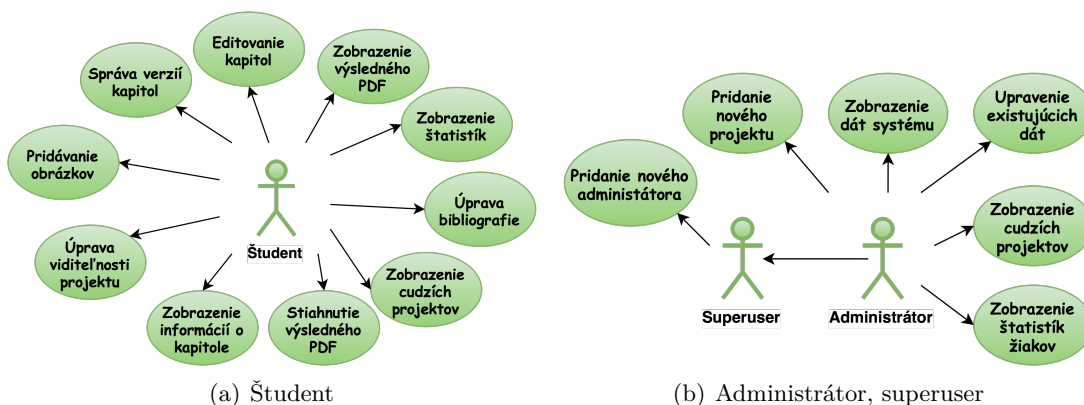
V tejto sekcii sa budem zaoberať tým, akú rolu má pri tvorbe projektu a dokumentácie študent a učiteľ. Každý z nich disponuje zoznamom činností, ktoré mu systém umožňuje vykonávať pri svojej práci.

### Študent

Najdôležitejšími prvkami v zobrazení aplikácie pre študenta je editovanie výslednej technickej dokumentácie. Študent môže prepínať medzi konkrétnymi kapitolami, ktoré následne upravuje. Pri každej kapitole môže zobraziť informácie o termíne odovzdávania a jej kvalite. Pomocou panelu nástrojov je možné generovať náhľad, sťahovať výsledný súbor, pridávať obrázky a spravovať verzie. Podporované je aj zadávanie bibliografických citácií. Moja aplikácia umožňuje inovatívne zobraziť štatistické údaje prostredníctvom tzv. progress barov, tabuliek a grafov. Študenti môžu navzájom zdieľať svoje práce za účelom vzájomnej inšpirácie. Diagram prípadov použitia zobrazuje obrázok 3.1.

### Administrátor - učiteľ

Pomocou administrátorského účtu je možné pridávať nové projekty a zobrazovať projekty iných užívateľov, vďaka čomu sa administrátori môžu navzájom inšpirovať. Dostupná je aj možnosť editácie a mazania objektov v systéme spolu s prehľadom, ktorý zobrazuje všetky položky uložené v systéme. Každý administrátor môže zobraziť základné štatistiky o svojich žiakoch a ich práci. Samotná rola administrátora je ďalej rozdelená na bežného a superuser administrátora, ktorému je navyše povolené pridávať nových administrátorov. Diagram prípadov použitia z pohľadu administrátora je zobrazený na obrázku 3.1.



Obr. 3.1: Diagramy prípadov použitia

## 3.3 Návrh užívateľského rozhrania

Sekcia popisuje požiadavky na užívateľské rozhranie a jeho samotný návrh. Zobrazuje rozdelenie na logické časti pričom pomenováva jednotlivé ovládacie prvky rozhrania s ich vizuálnou implementáciou.

## Úvodná stránka administrátora

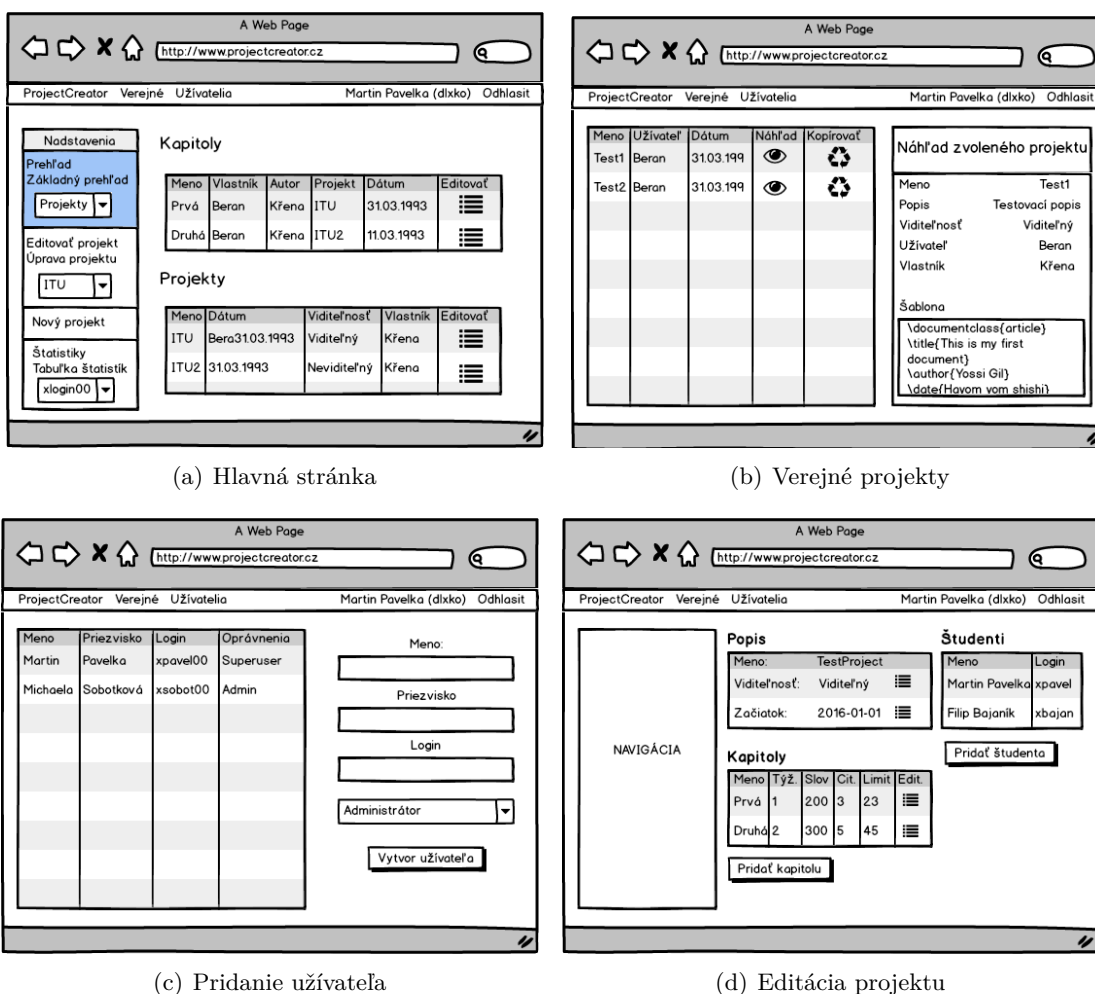
Ľavú časť stránky tvorí ovládací panel vo funkcii navigačného menu. Každá položka panela má krátky popis, aby bolo z prvého pohľadu jasné, o akú funkciu sa jedná. Ľavá časť je informačná, alebo je v nej užívateľ vyzvaný zadať všetky potrebné údaje. Zoznam položiek v databáze je prehľadne zobrazený v tabuľkách s možnosťou stránkovania. Prostredníctvom navigačného panela je ďalej možné editovať projekt, alebo zobraziť štatistiky študenta.

## Verejné projekty učiteľov

Rozdelenie stránky je opäť na dve časti. V ľavej časti sa nachádza tabuľka, v ktorej je zoznam všetkých projektov. Odtiaľ je možné nahliadnuť na detaily zvoleného projektu, alebo ho priamo celý skopírovať medzi vlastné. V časti napravo sa pod sebou zobrazujú detaily o konkrétnom projekte.

## Pridávanie administrátorov

Rozloženie elementov je opäť identické s možnosťou zobrazenia zoznamu administrátorov spolu s ich oprávneniami a formulárom pre pridanie nového administrátora. Všetky pohľady administrátora sú ukázané na obrázku 3.2.



Obr. 3.2: Mockupy administrátorského rozhrania

## Editor dokumentácie

Prvky na stránke sú umiestnené podobne ako pri projekte ShareL<sup>A</sup>T<sub>E</sub>X viď. sekciiu 2.1. V pravom stĺpci sa nachádza zoznam editovaných kapitol. Pod nimi je pridaná položka umožňujúca upravenie bibliografických údajov. V druhej časti sa nachádza panel nástrojov a miesto pre editáciu zdrojového súboru zvolenej kapitoly. Po stlačení ovládacieho prvku panela nástrojov sa vykonanie funkcie indikuje priamo zmenou textu a farby v paneli. Nástroje sú rozdelené podľa ikony a popisu funkcie na vkladanie obrázkov, generovanie náhľadu, uloženie a stiahnutie dokumentu. Pravá časť zobrazuje aktuálnu podobu výstupného súboru po jeho vygenerovaní. V prípade chyby generovania sa na rovnakom mieste zobrazí chybové hlásenie, aby užívateľ popri editácii kapitoly mohol mať neustále na očiach o akú chybu sa jednalo.

## Pohľad na triedu

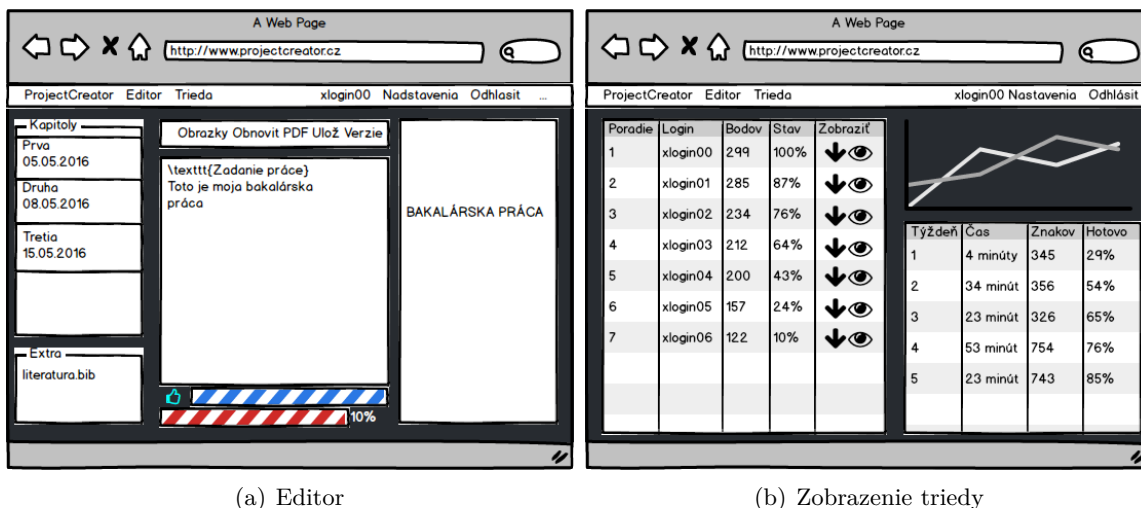
Stránka triedy sa delí na zoznam žiakov s tlačítkami, ktoré zobrazia zdieľaný PDF dokument alebo graf spolu so štatistickými údajmi informujúcimi o priebežnej činnosti študenta na projekte. Pod grafom sa nachádza tabuľka s detailným vysvetlením hodnôt grafu.

## Pridávanie obrázkov

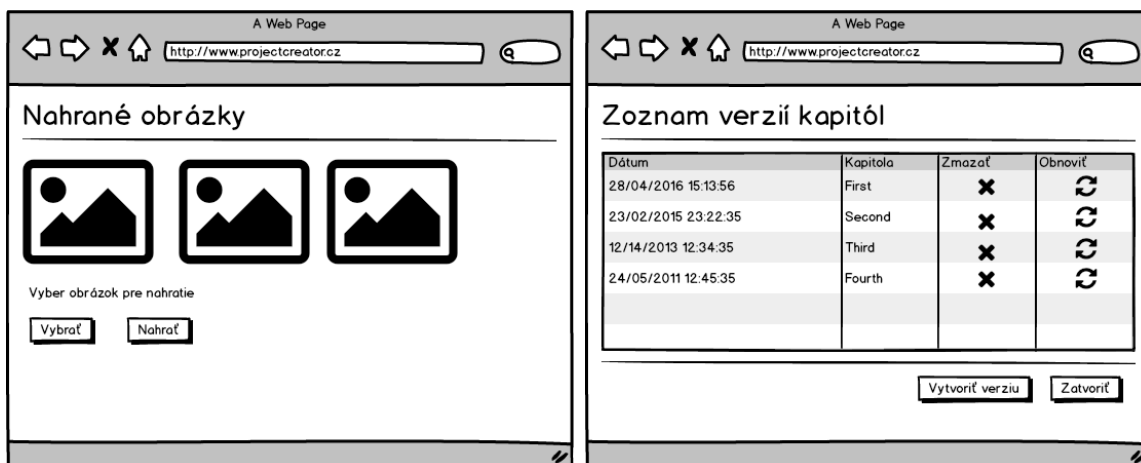
Obrázky je možné pridávať prostredníctvom pop-up okna, ktoré sa otvorí po kliknutí na tlačítko obrázkov z panela nástrojov. Vo vrchnej časti sa nachádza titulok a pod ním zoznam jednotlivých obrázkov, ktoré študent pridal do systému. Pridanie nového obrázku je vykonávané cez formulár pod zoznamom obrázkov.

## Správa verzií

Verzie sú implementované pomocou modálneho okna. Priamo pod titulkom je umiestnená tabuľka s časovými údajmi a zoznamom verzií. Obnovenie alebo zmazanie verzie je riadené ovládacími prvkami, ktoré sú umiestnené v pravej časti tabuľky. Vytvorenie novej verzie a zatvorenie okna je možné tlačítkami pod tabuľkou. Návrhy zobrazení študenta sú na obrázkoch 3.3 a 3.4.



Obr. 3.3: Mockupy rozhrania študenta



(a) Pridávanie obrázkov

(b) Pridávanie verzií

Obr. 3.4: Ďalšie mockupy rozhrania študenta

### 3.4 Správa projektov

Sekcia informuje o procesoch tvorby, editácie a kopírovaní projektov administrátormi systému. Pomenováva jednotlivé prvky využité v projekte a vysvetľuje ich funkciu. Zložitejšie vstupy sú popísané podrobnejšie, aby bolo jasné aké dáta očakávajú a ako presne je implementovaná ich funkčnosť.

#### Tvorba nového projektu

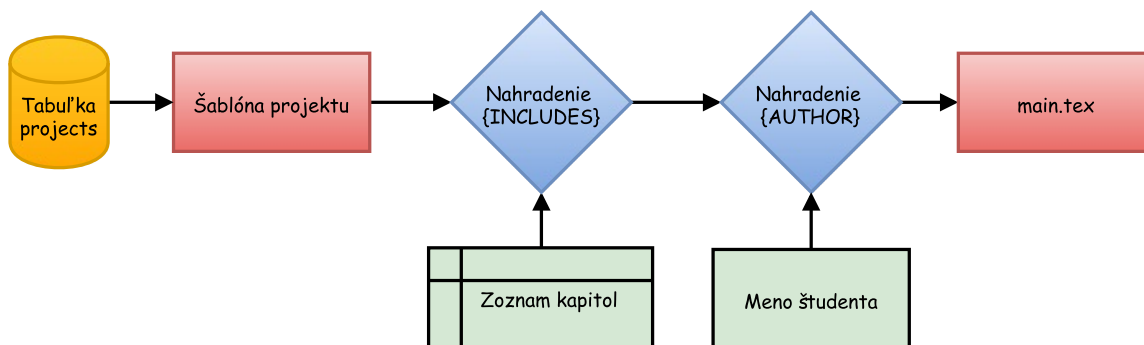
Projekt je možné vytvoriť po prihlásení užívateľa, ktorý má oprávnenia definované ako administrátor alebo „superuser“. Z navigačného menu je potrebné vybrať možnosť „Nový projekt“. Po kliknutí na položku menu sa zobrazí formulár s požadovanými vstupnými údajmi, ktorých zoznam je vymenovaný v tabuľke 3.2. V prípade odoslania chybných údajov, sa na začiatku formulára zobrazí hlásenie o chybe a všetky informácie, ktoré sme zadali, zostanú naďalej vyplnené a ich opätovné vypisovanie nie je už potrebné.

Vstup	Vysvetlenie
Meno projektu	Hlavný názov projektu, pod ktorým bude uložený v systéme
Popis projektu	Stručný popis, o čom projekt pojednáva
Šablóna projektu	Preddefinovaná kostra projektu
Viditeľnosť	Definuje, či sú práce študentov štandardne zdieľané
Začiatok projektu	Dátum začiatku projektu
JSON zoznam žiakov	Zoznam všetkých žiakov vo formáte JSON
Súbory	Súbory, ktoré budú skopírované každému študentovi
Kapitoly	Zoznam kapitol s ich jednotlivými parametrami

Tabuľka 3.2: Zoznam vstupov formulára

## Šablóna projektu

Šablóna definuje základnú kostru celého projektu v jazyku  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ , popísanom v kapitole 2.2. Štandardným vstupom je hotová kostra, ktorá definuje úvodnú stranu a použité balíčky. Pre vytvorenie správnej šablóny je nutné do nej vložiť kľúčové slová  $\{\text{INCLUDES}\}$  a  $\{\text{AUTHOR}\}$  spolu so zátvorkami. Na ich miesta sa dosadí zoznam jednotlivých zdrojových súborov kapitol v tvare  $\backslash\text{include}\{\text{ID\_KAPITOLY.tex}\}$  a autor práce v tvare  $\backslash\text{author}\{\text{Meno Priezvisko}\}$ . Šablóna sa v dátovej zložke študenta uloží pod menom `main.tex`. Príklad spracovania a samotný vstup je uvedený na obrázkoch 3.6 a 3.5. Príklad šablóny je predvyplnený v elemente formulára.



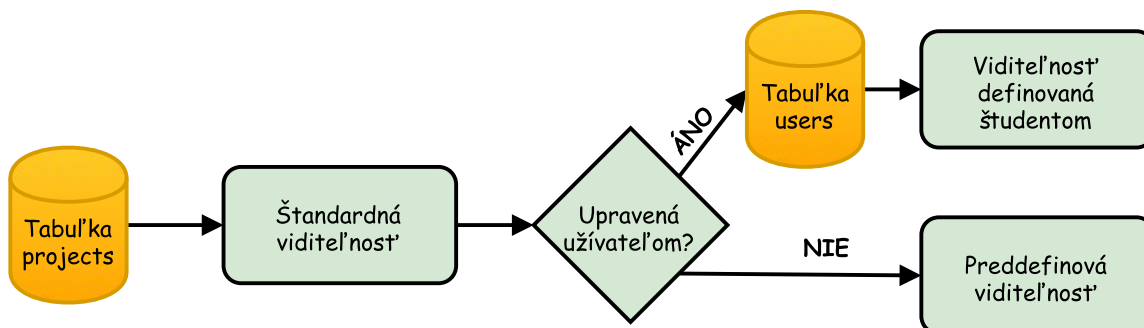
Obr. 3.5: Spracovanie šablóny

```
\documentclass{article}\title{First document}
\author{Yossi Gil}\date{\today}\begin{document}
\maketitle {INCLUDES} \end{document}
```

Obr. 3.6: Príklad vstupu šablóny

## Viditeľnosť

Po vytvorení projektu a inicializácii potrebných dát existuje v zobrazení študenta možnosť nahliadnuť do prác iných študentov. Pre každý projekt môžeme definovať východziu hodnotu viditeľnosti, ktorú si však môže každý žiak upraviť podľa svojich potrieb. Zobrazenie postupu určenia výslednej viditeľnosti je vidno na obrázku 3.7.



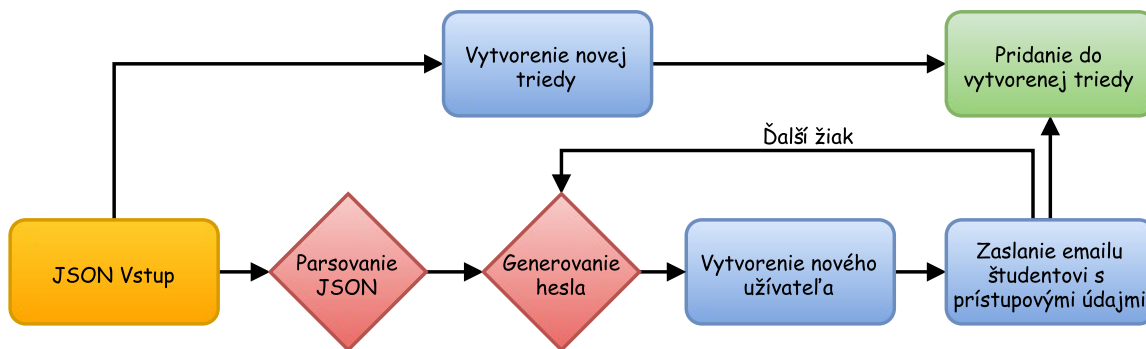
Obr. 3.7: Princíp určenia viditeľnosti

## Začiatok projektu

Pri vytváraní každého projektu je nutné definovať jeho začiatok, aby systém mohol správne priradiť informácie o termínoch odovzdávania a vykresliť požadované grafy. Vstupom začiatku projektu je štandardne začiatok semestra.

## Zoznam žiakov

Vstup je zadávaný vo formáte JSON, ktorý je podľa môjho názoru jeden z najprehľadnejších a velice úsporných. Dáta sa nachádzajú pod objektom s názvom „student“, pričom každá položka má definované užívateľské meno, krstné meno a priezvisko. Príklad vstupu je zobrazený na obrázku 3.9. Pri vytvorení užívateľa, je mu odoslaný mail, ktorý ho informuje, že bol zaregistrovaný v systéme a zasiela mu informáciu o jeho prístupových údajoch, ktoré môže použiť pri prihlasovaní. Formát a vzhľad mailu je popísaný obrázkom 3.10. Odosielanie emailov je riadené prostredníctvom knižnice PHPMailer, popísanej v sekcii 2.4. Podrobnejší princíp generovania jednotlivých žiakov je zobrazený na obrázku 3.8.



Obr. 3.8: Generovanie nových študentov

```
{
  "students": [
    { "login": "xlogin00", "name": "Name0", "surname": "Surname0" },
    { "login": "xlogin01", "name": "Name1", "surname": "Surname1" },
    { "login": "xlogin02", "name": "Name2", "surname": "Surname2" },
  ]
}
```

Obr. 3.9: Príklad vstupu zoznamu študentov

```
Dobý deň páni Uzivatel3

Boli ste zaregistrovaný do služby ProjectCreator

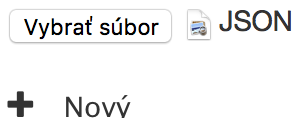
Vaše prihlasovacie meno je test03 a heslo v-qe7r1y
```

Obr. 3.10: Privítací mail

## Súbory

Pridáva možnosť zahrnúť do projektu súbory použité pri kompilácii technickej správy. Štandardne sa prikladá súbor `czechiso.bst`, ktorý implementuje normu pri písaní bibliografických údajov. Súbory je možné pridávať dynamicky po kliknutí na tlačítko **+ Nový**, viď. obrázok 3.11.

### Súbory



Obr. 3.11: Pridávanie súborov

## Kapitoly

Do projektu je možné pridať viacero kapitol. Existujú dva rôzne spôsoby, ako pridať novú kapitolu. V prvom prípade je potrebné stlačiť tlačítko **+ Nová**, pričom sa nám vytvorí nový formulár, ktorý vidno na obrázku 3.12, do ktorého je potreba vyplniť všetky údaje z tabuľky 3.3. V prípade voľby **+ Kópia**, formulár presmeruje užívateľa na zobrazenie verejných projektov. Kapitoly si následne môžeme zobrazit po stlačení ovládacieho prvku v stĺpci „Kapitoly“ viď. obrázok 3.13. Zobrazia sa nám všetky kapitoly zvoleného projektu a po nahliadnutí na informácie môžeme už hotovú kapitolu vložiť do formulára tvorby nového projektu. Nakoniec sa nám vráti starý formulár, kde sa predvyplnia všetky potrebné údaje podľa údajov kapitoly, ktorú sme kopírovali.

Vstup	Vysvetlenie
Meno kapitoly	Meno, pod ktorým je kapitola zobrazená študentovi
Rozsah - slov	Počet slov chýbajúcich na dokončenie kapitoly 100%
Rozsah - citácie	Počet citácii pre dokončenie kapitoly
Percent pre limit	Limit pre slová a citácie pre zobrazenie nekvalitnej kapitoly
Termín odovzdania	Počet týždňov od začiatku pre dokončenie kapitoly

Tabuľka 3.3: Zoznam vstupov kapitoly



Obr. 3.12: Vytvorenie novej kapitoly

Meno	Užívateľ	Dátum	Info	Kapitoly
TestProject	Martin Pavelka	2016-04-20 11:51:00		

Náhľad zvoleného projektu		
Prva		
Druha		

Obr. 3.13: Kopírovanie existujúcej kapitoly

### 3.5 Kopírovanie verejného projektu

Pre kopírovanie projektu od iného administrátora je nutné zobrazíť stránku pomocou odkazu v hornom navigačnom menu s textom „Verejné“. Po zadaní mena skopírovaného projektu a zoznamu vlastných žiakov sa vytvorí duplicitný projekt pod zvoleným menom s rovnakými kapitolami, termínmi, informáciami a šablónou so súbormi.

### 3.6 Editácia projektu

Editovať projekt je možné po voľbe „Editovať projekt“ a zadaní projektu na editáciu prostredníctvom „selectboxu“. Zobrazenie je prehľadne rozdelené na úpravu základných informácií, kapitol a študentov. Po stlačení ovládacieho prvku na editáciu je možné postupne editovať aj detailnejšie informácie o projekte alebo kapitole. Ukážka hlavného zobrazenia editácie je vidieť na obrázku 3.14.

Popis						
Meno:	TestProject					
Viditeľnosť	Viditeľný					
Začiatok	2016-01-01 00:00:00					
Kapitoly						
Meno	Týždeň	Slov	Citácií	Varovaci limit	Editovať	Odstrániť
Prva	1	100	10	54		
Druha	2	200	20	33		

Študenti					
Show	10	entries	Search: <input type="text"/>		
Meno	↓	Login	↑	Zmazať	↑
Name0 Surname0		xlogin00			
Name1 Surname1		xlogin01			
Name5 Surname5		xlogin05			
Name6 Surname6		xlogin06			

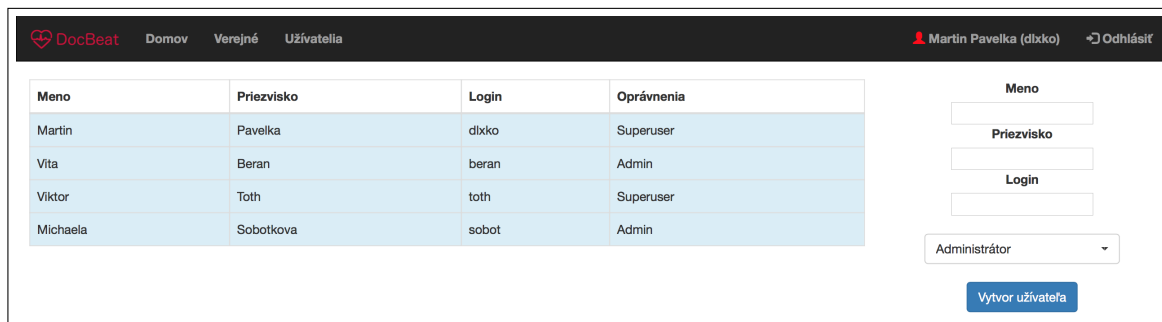
Showing 1 to 4 of 4 entries Previous **1** Next

Obr. 3.14: Editácia projektu

### 3.7 Pridávanie administrátorov

Do systému je možné pridávať nových administrátorov užívateľom, ktorý má oprávnenia „superuser“. Ak je funkcia povolená, je možné pristúpiť na zobrazenie administrátorov pomocou odkazu „Užívateľa“. V pravej časti sa nachádza tabuľka obsahujúca všetkých administrátorov v systéme spolu s ich oprávneniami. Pridanie nového užívateľa je možné prostredníctvom formulára napravo, po zadaní mena, priezviska, loginu a oprávnení. Ako rozhranie vyzerá, zobrazuje obrázok 3.15.





Obr. 3.15: Editácia administrátorov

### 3.8 Editor dokumentácie

Sekcia popisuje implementácie editora dokumentácie v zobrazení študenta. Zobrazenie je logicky členené na zoznam kapitol a literatúru, panel nástrojov, editor kódu a náhľad výsledného súboru.

#### Panel nástrojov

Nachádza sa nad editorom kódu. Dostupných je viacero bežných funkcií využívaných pri tvorbe dokumentácie. Stav vykonávania funkcie je zobrazovaný priamo v paneli, zmenou farby textu a ikonky. Panel je možno vidieť na obrázku 3.16.



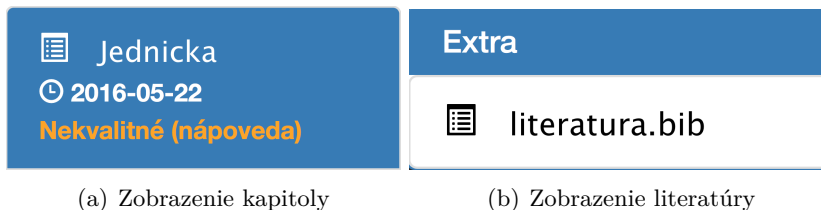
Obr. 3.16: Panel nástrojov

#### Náhľad

Na pravej strane editora sa nachádza náhľad výsledného dokumentu. Jeho obnovovanie je riadené z ovládacieho panelu. V prípade chyby generovania sa zobrazí chybové hlásenie popisujúce chybu a tak je študentovi umožnené bez nutnosti skryť hlásenie, chybu opraviť.

#### Zoznam kapitol a literatúry

Zoznam sa nachádza v ľavej časti stránky. Každé kapitole je priradený názov, pod ktorým je čas očakávaného odovzdania. V prípade, že bol prekročený limit pre varovanie, zobrazí sa text s odkazom na pomocníka vyzývajúceho na pridanie slov alebo citácii vid. obrázok 3.17. Po zvolení príslušnej kapitoly sa jej obsah zobrazí v editore kódu. Implementácia načítania obsahu je vysvetlená na obrázku 3.18.



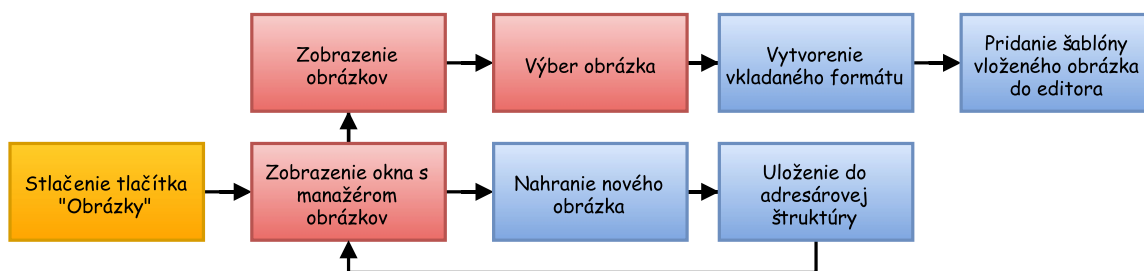
Obr. 3.17: Zoznam kapitol a literatúry



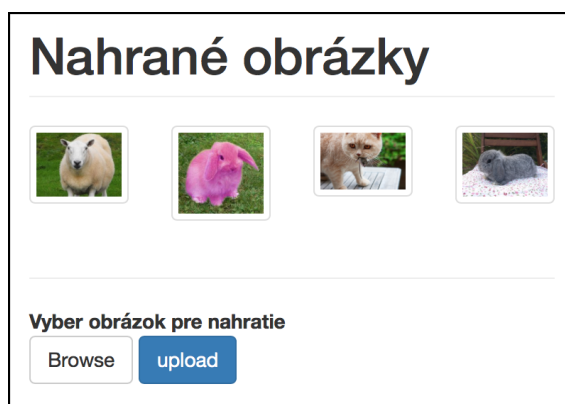
Obr. 3.18: Stavový graf pre načítanie a výber kapitol

### Obrázky

Po stlačení tlačítka „Obrázky“ sa zobrazí nové okno so zoznamom obrázkov, ktoré užívateľ nahral do systému, viď. obrázok 3.20. V prípade, že chce nahrať nový obrázok, je mu to umožnené prostredníctvom tlačítka na nahrávanie súborov. PHP skript uloží súbor do adresárovej štruktúry študenta a zobrazí ho. Následne môže používateľ zvoliť súbor a vložiť ho do svojho zdrojového súboru spolu s predpripravenou syntaxou pre prvok obrázka. Implementácia je popísaná obrázkom 3.19.



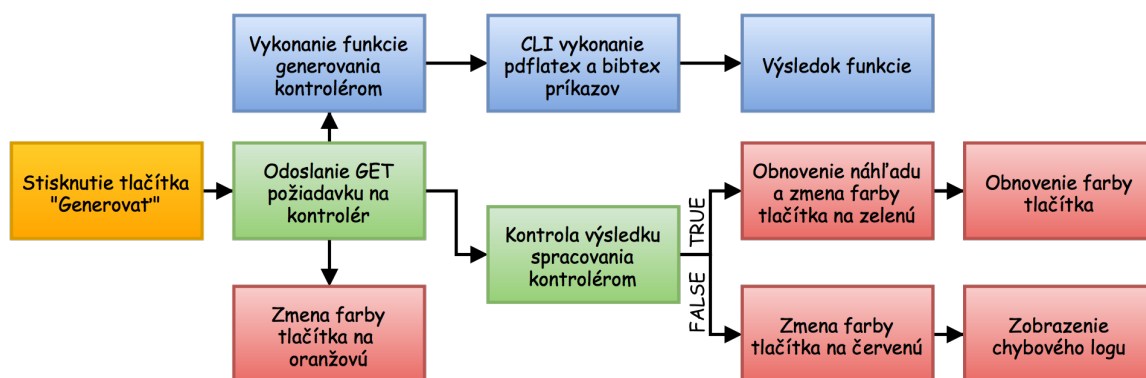
Obr. 3.19: Stavový graf pre nahrávanie obrázkov



Obr. 3.20: Ukážka okna so správcom fotiek

## Generovanie náhľadu

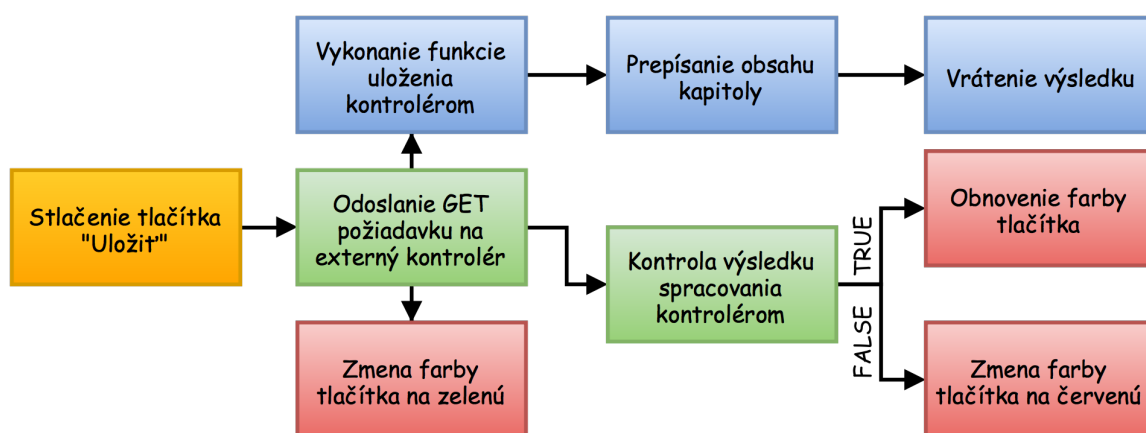
Po stlačení tlačítka „generovanie“, tlačítko zmení farbu na oranžovú, čím informuje používateľa o úspešnom spustení operácie. Prostredníctvom skriptu Ajax sa odošle GET požiadavka na konkrétny kontrolér, ktorá zabezpečí vykonanie príkazu na pozadí a vygeneruje nový náhľad. V prípade úspešného ukončenia tlačítko zmení farbu na zelenú, v opačnom prípade na červenú farbu indikujúcu chybu pri preklade zdrojového súboru. V prípade úspechu stránka automaticky obnoví náhľad PDF súboru s vykonanými zmenami. Chybové hlásenie sa zobrazuje na rovnakom mieste ako samotný náhľad. Podrobnejší popis implementácie je vysvetlený na obrázku 3.21.



Obr. 3.21: Stavový graf pre generovanie

## Ukladanie zmien

Pri zvolení funkcie uloženia sa opäť pošle požiadavka POST prostredníctvom skriptu Ajax na externý kontrolér, ktorý uloží nové zmeny do adresárovej štruktúry. Nahradí sa pôvodný obsah kapitoly, ktorá je aktuálne upravovaná. Implementáciu detailnejšie popisuje schéma na obrázku 3.22.



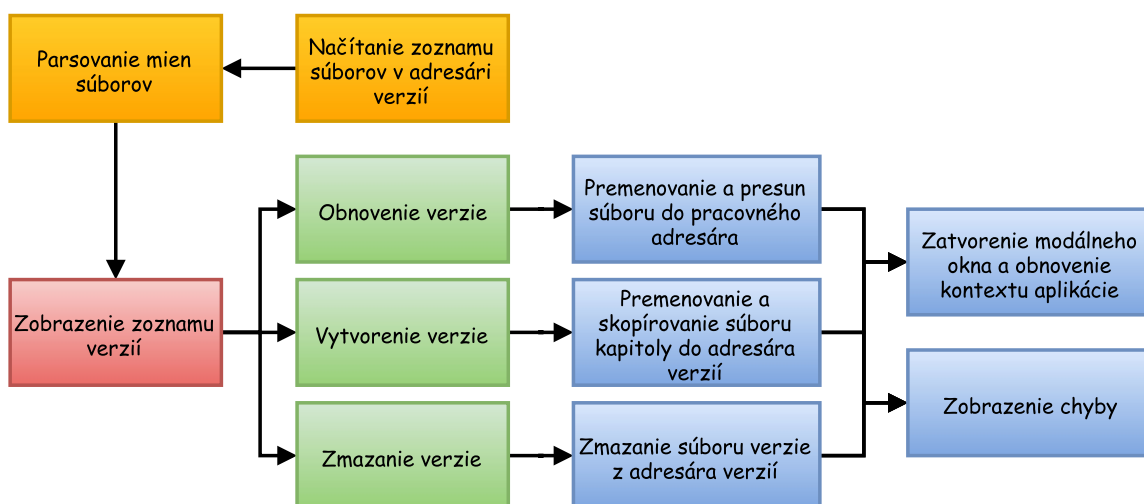
Obr. 3.22: Stavový graf pre ukladanie

## Editor kódu

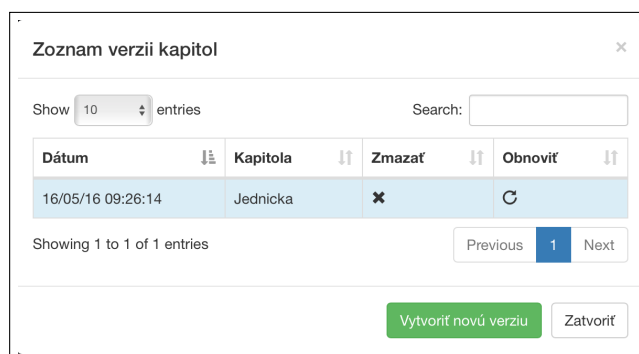
Služi na editovanie obsahu jednotlivých kapitol. Na jeho implementáciu som využil riešenie CodeMirror, ktoré je popísané v kapitole 2.4, vďaka ktorému je pri písaní možné zobrazenie syntaxe a čísiel riadkov kódu.

## Správa verzií

Zobrazenie zoznamu verzií je dostupné prostredníctvom tlačítka na paneli nástrojov žiaka. Po jeho stlačení sa zobrazí modálne okno s tabuľkou, ktorá zobrazuje jednotlivé položky verzií v riadkoch, viď. obrázok 3.24. Pri každej verzii je dostupný dátum jej vytvorenia, názov kapitoly, pre ktorú bola vytvorená a možnosť jej zmazania či načítania. V prípade potreby vytvorenia novej verzie je táto funkcia implementovaná rovnako v modálnom okne prostredníctvom zeleného tlačítka. Načítanie zoznamu verzií je riešené získaním zoznamu súborov v adresári, ktorý sa nachádza v súborovej štruktúre v špeciálnom priečinku pre verzie. V prípade zmazania a obnovenia verzie dochádza k operáciám nad touto súborovou štruktúrou a to kopírovanie a mazanie jednotlivých súborov. Po úspešnom obnovení je zabezpečená okamžitá obnova všetkých prvkov stránky do aktuálneho stavu. V prípade chyby je o nej užívateľ informovaný zobrazením červenej chybovej hlášky na mieste tlačítka pre verzie. Podrobnejší popis funkcie ponúka obrázok 3.23



Obr. 3.23: Stavový graf pre správu verzií



Obr. 3.24: Ukážka okna spravujúceho verzie

### 3.9 Štatistiky a grafy

Nasledujúca sekcia je venovaná štatistikám a prehľadom využitým v mojom projekte. Popisuje význam a funkciu jednotlivých štatistických prvkov a zobrazuje ich implementáciu.

#### Bodovací systém

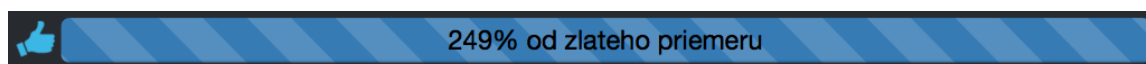
Za aktivitu pri tvorbe dokumentácie a úspešné splnenie požadovaných parametrov dostáva žiak bodové hodnotenie. Hodnotenie je rozdelené na základné časti podľa tabuľky 3.4.

Názov	Vysvetlenie
Char points	Body za písanie. V prípade, že žiak napíše určený počet znakov, dostane bod. Kopírovaním sa počet napísaných znakov nezvyšuje.
Time points	Body za aktivitu. Ak má žiak otvorenú stránku a prejavuje aktivitu po učení čase sa mu pripočíta bod.
Done points	Vyjadrujú percentuálne pre každú kapitolu a pre celý projekt kompletnosť požiadaviek. Prepočítavajú sa pri uložení kapitoly.

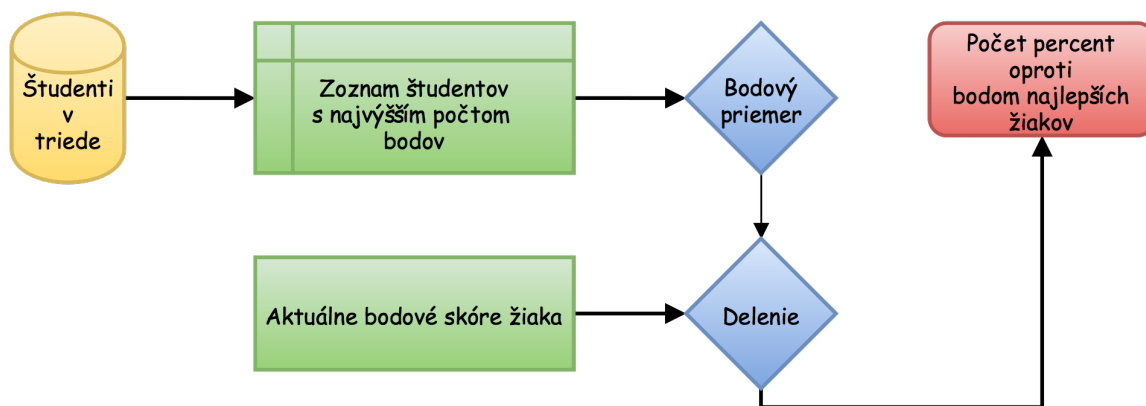
Tabuľka 3.4: Zoznam bodovania

#### Lead ukazovateľ

Funguje na princípe vzájomnej súťaživosti. Ukazovateľ zobrazuje počet všetkých bodov oproti „špičke ročníka“. Špička je skupina žiakov s najväčším počtom bodov v triede. Počet žiakov špičky sa dá nastaviť. Z priemeru bodov špičky a bodov žiaka sa nakoniec odvodí výsledné percentuálne číslo. Pomocou takéhoto procesu sa nemôže stať, že by jeden žiak bol schopný výrazne ovplyvňovať celkovú hranicu bodov. V prípade, že má žiak aspoň polovicu bodov skupiny najlepších žiakov, je mu zobrazený emotikon s prstom, ktorý ukazuje hore, v opačnom prípade prst smeruje dole. Ukážka je na obrázku 3.25 a výpočet na 3.26.



Obr. 3.25: Lead ukazovateľ



Obr. 3.26: Stavový graf pre lead ukazovateľ

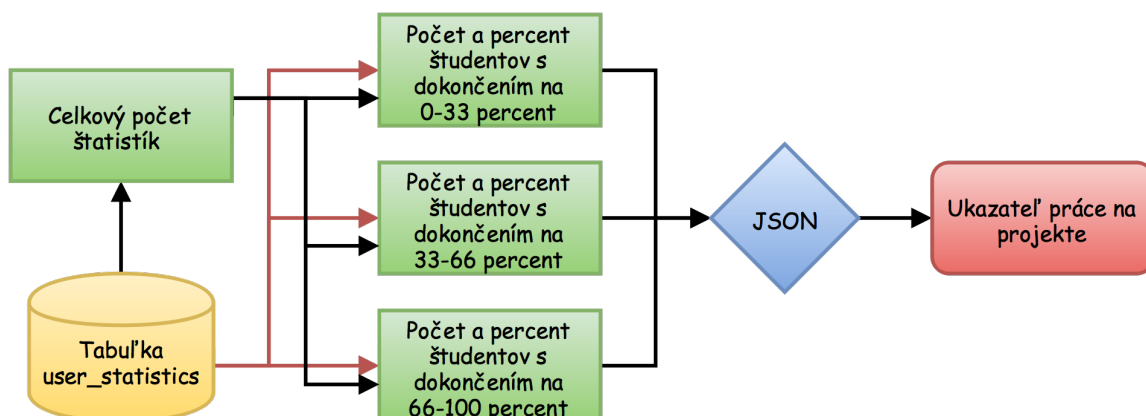
## Ukazovateľ práce v triede

Zobrazuje mieru splnenia požiadaviek, stanovených pre projekt v rámci triedy. Rozdelený je na tri časti, ktoré sú farebne rozlíšené. Pre každú farbu platí isté kritérium dokončenia. Červená farba značí dokončenie na 0–33%, oranžová 33–66% a zelená 66–100%. Nulový počet percent a plný počet percent by osamote pôsobili skôr demotivačne, preto som tieto hodnoty pričlenil k hlavnému rozdeleniu. Prvok opäť slúži ako motivácia študentov. Percento za grafom zobrazuje aktuálny stav splnenia pre žiaka, ktorý je prihlásený do systému. Pri získavaní dát sa pošle požiadavka na databázu, ktorá vráti pre každé kritérium počet žiakov. Pomocou Ajax požiadavky sa datá načítajú a zobrazia na stránke. Ukážka grafu je na obrázku 3.27. Spôsob načítavania dát popisuje schéma 3.28

Obrázok zobrazuje stav, keď šesťnásť študentov nemá hotovú ani jednu tretinu práce, dvaja majú viac ako jednu tretinu a zvyšní dvaja majú dokončené viac ako dve tretiny práce, a tak sa blížia k dokončeniu písania technickej správy.



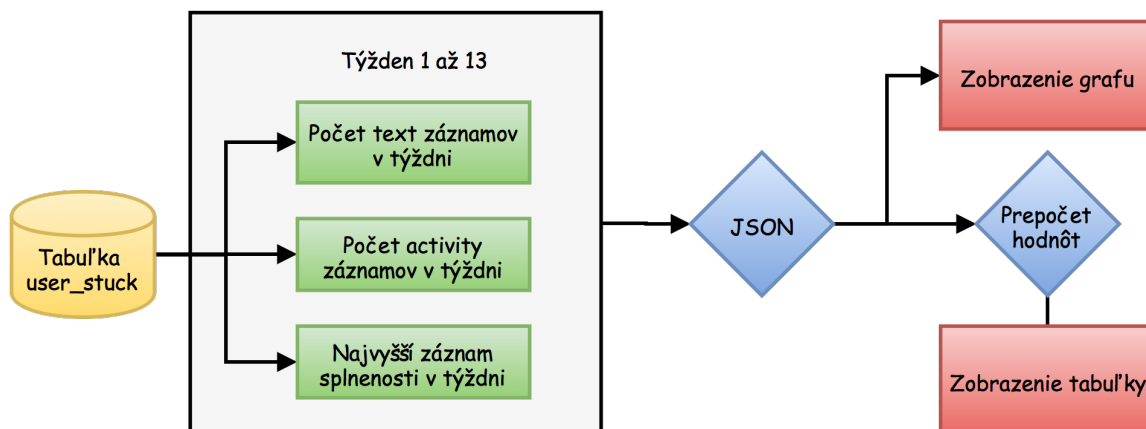
Obr. 3.27: Ukazovateľ práce v triede



Obr. 3.28: Stavový graf pre ukazovateľa práce na projekte

## Priebežná práca na projekte

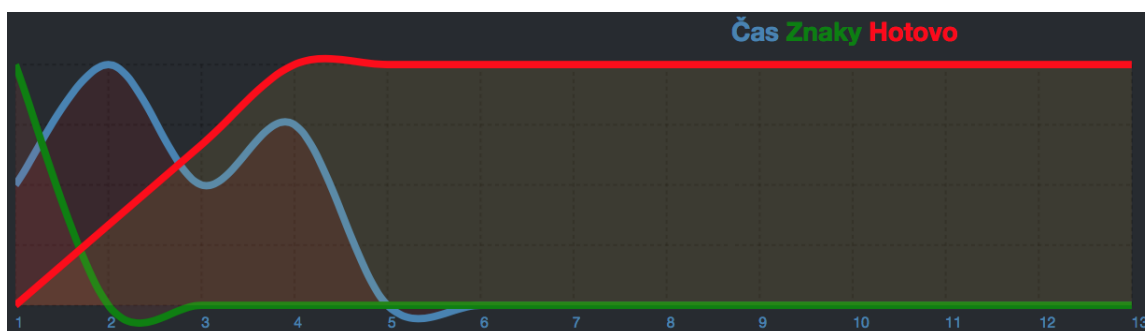
Prvkom, ktorý môžu študenti využiť k inšpirácii od spolužiakov je graf 3.30 a tabuľka 3.31, ktoré informujú o priebežnej činnosti konkrétneho žiaka na technickej správe. V grafe sú zobrazené hodnoty nadobudnutých bodov za jednotlivé študijné týždne. Pre každý typ bodov je zobrazený priebeh zvlášť. Nad grafom sa nachádza legenda informujúca o význame kriviek. **Modrá** zobrazuje čas, po ktorý bol študent aktívny v systéme. **Zelená** krivka popisuje intenzitu písania textu. **Červená** vypovedá o úrovni dokončenia kritérií všetkých kapitol projektu. Všetky hodnoty v grafe sú normalizované na hodnoty od nula po jedna. Pod grafom sa nachádza tabuľka, ktorá detailne popisuje hodnoty a prepočítava ich na minúty a počty znakov. Percentá sú do tabuľky doplnené bez prepočtu priamo. Každý riadok zobrazuje konkrétny týždeň semestra. Graf je implementovaný projektom Chartist viď. 2.4. Doručovanie dát do zobrazenia prebieha volaním Ajax skriptu.



Obr. 3.29: Stavový graf priebežnej práce na projekte

Z grafu na obrázku 3.30 je možné zhodnotiť, že študent v prvom týždni aktívne písal text pričom ale v nasledujúcom nevykonával už žiadnu úpravu. Počas prvých týždňov prejavoval aktivitu v systéme. Projekt dokončil v štvrtom týždni.

Na základe podobných analýz je možné vyhodnotiť chovanie jednotlivých žiakov a vytvoriť si predstavu o práci na ich projekte. Ak študenti pracovali na kapitole len v jednom týždni, je možné upraviť termín odovzdávania a pod.. Podobnými úpravami má učiteľ možnosť zlepšiť návrh a viesť študentov k lepším výsledkom.



Obr. 3.30: Graf priebežnej práce na projekte

Týždeň	Čas	Znak	Hotovo
1	0 minut	100 znakov	0%
2	1 minut	0 znakov	10%
3	0 minut	0 znakov	20%
4	1 minut	0 znakov	30%

Obr. 3.31: Tabuľka priebežnej práce na projekte

## Úpravy po termíne

V prostredí administrátora je možné pristúpiť z menu na zobrazenie štatistík konkrétneho žiaka. Prvá tabuľka obsahuje počet jednotlivých bodov a číslo, ktoré udáva ich počet spolu. V tabuľke nižšie sa nachádza zoznam jednotlivých týždňov. Ku každému termínu je doplnený dátum a zoznam kapitol. Jednotlivé číselné údaje patriace konkrétnemu termínu a kapitole udávajú počet vykonaných zmien riadkov, vid. obrázok 3.32.

Tabuľka je určená pre učiteľa, ktorý vďaka nej môže vidieť, ako žiak dodržiaval termíny a či spätne upravoval kapitoly alebo naopak nevykonával žiadnu zmenu. Funkcia slúži na odhalenie študentov, ktorí vytvoria nekvalitný text a na koniec celú prácu prerobia.

Implementácia je riešená pomocou lokálnych GIT repozitárov. V každom termíne sa zavolá v adresári študenta `git commit`. Pri zobrazení štatistík potom aplikácia vygeneruje log pomocou `git log --stat` a uloží ho do dočasného súboru. Súbor následne vyhodnotí a vypočíta všetky požadované hodnoty pre každú kapitolu. Nakoniec vygeneruje výslednú tabuľku. Proces zobrazuje obrázok 3.33.

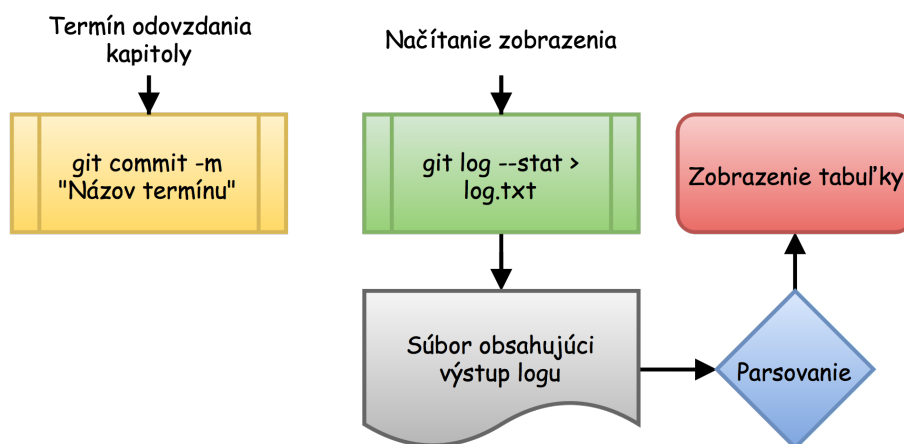
## Štatistiky užívateľa xlogin00

Časové body	Znakové body	Body aktivity	Body spolu
1	200	0	9

Termín	Dátum	Jednicka	Dvojka	Trojka	Stvorka
Deviaty	Mon May 16 14:04:17 2016 +0200	0	0	0	7
Osmý	Mon May 16 14:04:07 2016 +0200	0	0	0	6
Siedmy	Mon May 16 14:03:57 2016 +0200	0	0	0	5
Siesty	Mon May 16 14:03:42 2016 +0200	0	0	11	0
Piaty	Mon May 16 14:03:26 2016 +0200	0	0	10	0
Štvrtý	Mon May 16 14:03:11 2016 +0200	0	0	11	0
Tretí	Mon May 16 14:02:56 2016 +0200	0	12	0	0

Obr. 3.32: Tabuľka priebežnej práce na projekte



Obr. 3.33: Tabuľka priebežnej práce na projekte



## Kapitola 4

# Aplikácia DocBeat

Projekt bol implementovaný pomocou návrhového vzoru MVC a frameworku CodeIgniter riešeného v sekcii 2.4. Kapitola stručne popisuje back-end aplikácie, členenie zdrojových súborov a ich význam. Zobrazená je aj štruktúra databázy prostredníctvom tabuliek.

### 4.1 Implementácia pomocou CodeIgniter

#### Kontroléry

Spracovávajú vstupné požiadavky a generujú výslednú stránku. V projekte sa nachádzajú v priečinku `application/controllers`. Ich zoznam je popísaný v tabuľke 4.1.

Názov	Vysvetlenie
<code>admin</code>	Hlavná stránka pohľadu administrátora
<code>admin_edit</code>	Jednotlivé stránky editovania kapitol, ročníkov atď.
<code>admin_public</code>	Generovanie stránky obsahujúcej prehľad projektov
<code>admin_super</code>	Stránka umožňujúca pridávanie nových administrátorov
<code>index</code>	Prihlasovacia stránka po načítaní aplikácie
<code>student</code>	Editor dokumentácie v zobrazení študenta
<code>student_lead</code>	Zobrazenie triedy a zoznamu prác študentov
<code>student_pictures</code>	Okno pre pridávanie obrázkov

Tabuľka 4.1: Zoznam kontrolérov

#### Modely

Uchovávajú všetkú funkcionality aplikácie. Nachádzajú sa v priečinku `application/models`. Volajú sa z kontrolérov pričom hlavne zabezpečujú operácie nad databázou a adresárovou štruktúrou. Základné rozdelenie modelov aplikácie sa nachádza v tabuľke 4.2.

Názov	Vysvetlenie
<b>controller</b>	Sú modely, ktoré dopĺňajú funkcie kontroléru, nezasahujú však do databázy
<b>database</b>	Pridávajú, upravujú a načítavajú akékoľvek dáta z databázy.
<b>functions</b>	Špeciálne funkcie pre emaily, JSON, git atď.

Tabuľka 4.2: Rozdelenie modelov

## Zobrazenia

Nachádzajú sa v adresári `application/views`. Sú načítavané prostredníctvom kontrolérov a obsahujú HTML kostru výsledných stránok. Vkladajú sa do nich dáta načítané z modelov a tak dopĺňajú MVC trojicu. Ich základné rozdelenie je popísané v tabuľke 4.3.

Názov	Vysvetlenie
<b>admin</b>	Zoskupujú všetky zobrazenia na hlavnej stránke administrátora
<b>admin_edit</b>	Obsahujú zobrazenia všetkých stránok na editáciu objektov
<b>includes</b>	Vkladajú sa do iných zobrazení a obsahujú odkazy na knižnice
<b>mails</b>	Obsahujú kostru pre emaily rozposielané študentom a administrátorom
<b>public</b>	Obsahuje štruktúru modálneho okna pri kopírovaní projektu
<b>student</b>	Zoskupujú všetky zobrazenia v editore pri pohľade študenta
<b>*.tpl</b>	Sú hlavné zobrazenia volané priamo z kontrolérov

Tabuľka 4.3: Rozdelenie zobrazení

## Assets

Obsahujú dáta, ktoré sú načítavané spolu so stránkou. Pristupuje sa k nim priamo cez URL adresu. Uložené sú v priečinku `/assets`. Medzi súbory patria jednotlivé knižnice, štýly ale aj samotné užívateľské dáta. Prehľadný zoznam položiek sa nachádza v tabuľke 4.4.

Názov	Vysvetlenie	Názov	Vysvetlenie
<b>images</b>	Obrázky a ikonky	<b>js</b>	Vlastné skripty
<b>css</b>	Kaskádové štýly	<b>jquery</b>	jQuery
<b>bootstrap</b>	Bootstrap	<b>bootstrapselect</b>	Bootstrapselect
<b>chartist</b>	Chartist	<b>codemirror</b>	CodeMirror
<b>datepicker</b>	DatePicker	<b>datatables</b>	DataTables
<b>additional</b>	Ostatné skripty	<b>dáta</b>	Užívateľské dáta

Tabuľka 4.4: Rozdelenie assets

## 4.2 Datový model

Na obrázku 4.1 a tabuľke 4.5 je zobrazený zoznam databázových tabuliek spoločne s ich parametrami. Každý parameter obsahuje vysvetlenie. Primárne, obyčajné a cudzie kľúče sú v tabuľkách oddelené čiarou.

Chapters	Description
id : int	Identifikátor
name : text	Meno
content : text	Obsah
words : int	Slov
cites : int	Citácií
warning_limit : int	Percent pre varovanie
creation_date : time	Časová značka
project : int	Odkaz na projekt
user : int	Vlastník
origin : int	Autor

Projects	Description
id : int	Identifikátor
name : text	Názov
description : text	Popis
base : text	Šablona
visible : int	Viditeľnosť
creation_date : time	Časová značka
user : int	Vlastník
origin : int	Autor

Classes	Description
id : int	Identifikátor
name : text	Názov
creation_date : time	Časové razítko
user : int	Vlastník
project : int	Projekt

Instances	Description
id : int	Identifikátor
name : text	Názov
start_date : time	Začiatok
creation_date : time	Časové razítko
project : int	Projekt
class : int	Trieda
user : int	Vlastník

User_statistics	Description
id : int	Identifikátor
time_points : int	Body za čas
char_points : int	Body za text
generate_points : int	Body generovaní
percent_done : int	Percent hotovo
user : int	Student
project : int	Projekt

User_donestat	Description
id : int	Identifikátor
done : int	Percent hotových
user : int	Student
chapter : int	Kapitola

User_stucks	Description
id : int	Identifikátor
type : int	Typ štatistiky
value : int	Hodnota
time : time	Časové razítko
user : int	Student

Users	Description
id : int	Identifikátor
username : text	Login
name : text	Meno
surname : text	Priezvisko
password : text	Heslo
permission : int	Oprávnenia
creation_date : time	Časová značka
custom_visibility : int	Viditeľnosť
class : int	Trieda
project : int	Projekt
admin : int	Administrátor

Project_lines	Description
id : int	Identifikátor
weeks : int	Týždňov
creation_date : int	Časové razítko
project : int	Projekt
chapter : int	Kapitola
user : int	Vlastník

Project_chapter	Description
id : int	Identifikátor
creation_date : time	Časové razítko
project	Projekt
chapter	Kapitola
user	Vlastník

Obr. 4.1: Zoznam tabuliek databázy

Tabuľka	Funkcia
Chapters	Tabuľka popisujúca objekt kapitoly
Users	Popisuje študentov a administrátorov v systéme
Projects	Hlavná logická jednotka aplikácie popisujúca projekt
Classes	Objekt triedy, ktorá sa referuje projekt
Instances	Inštancia projektu popisujúca začiatok semestra
Project_chapter	Spojujúca tabuľka 1 ku N projektu a kapitol
Project_lines	Spojujúca tabuľka 1 ku N kapitoly a termínu odovzdania
User_statistics	Obsahuje súhrnné informácie o štatistikách študentov
User_donestats	Informácie o dokončení jednotlivých kapitol študentmi
User_stucks	Granulárne informácie o bodoch s časovým razítkom

Tabuľka 4.5: Zoznam tabuliek v databázy

## Kapitola 5

# Testovanie, analýza a výsledky

Kapitola popisuje proces testovania aplikácie, definuje podporované platformy a poukazuje na komplikácie, ktoré sa vyskytli pri testovaní. Záver kapitoly zhŕňa výsledky a vedie nad nimi diskusiu.

### 5.1 Spôsob testovania

Počas vývoja aplikácie prebiehalo testovanie funkčnosti po dokončení každej väčšej súvislej časti. Nakoniec bola aplikácia otestovaná ako hotový a súvislý celok. Vývoj aplikácie prebiehal na prehliadači **Safari**, avšak testovanie bolo vykonané aj na prehliadačoch **Chrome** a **Firefox**. Problémy ktoré boli odhalené, sú popísané v sekcii 5.2. Po overení funkčnosti som aplikáciu nechal vyskúšať viacerým ľuďom a nechal ich vyplniť jednoduchý dotazník, ktorý zobrazuje ich pohľad na kvalitu spracovania aplikácie. O výsledku testovania na subjektoch pojednáva kapitola 5.3.

### 5.2 Problémy pri testovaní

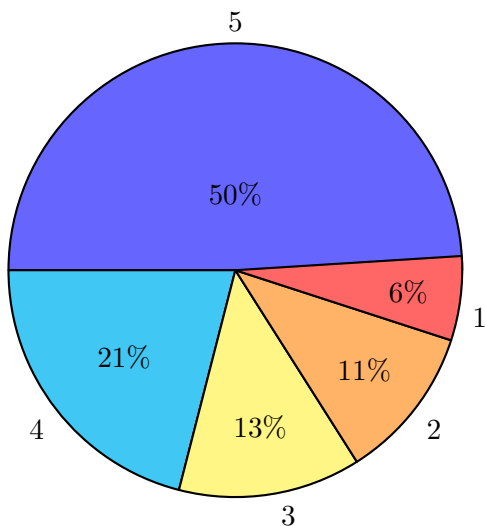
Pri testovaní aplikácie som narazil na problém s vyrovnávacou pamäťou a to hlavne v prehliadači Safari. Problém sa prejavuje pri generovaní náhľadu, pričom nový súbor neobsahuje zmeny vykonané v kapitole. Pri podrobnejšej analýze problému sa mi podarilo zistiť, že prehliadač naďalej uchováva staršiu verziu výsledného dokumentu a pri obnovení nepožaduje zaslanie novej. Obnovovanie náhľadu je riešené pomocou jazyka JavaScript.

### 5.3 Testovanie na subjektoch

Pri niektorých otázkach návrhu bolo potrebné vykonať prieskum v širšej skupine ľudí. Testovanie bolo vykonané prostredníctvom testovacích otázok a jednoduchého známkovania od čísla jedna, ktoré značilo najlepšiu reakciu, po číslo päť, ktorým bola označená najhoršia reakcia. V strede hodnotenia na čísle tri som označil odpoveď ako negatívnu. Výsledky testovania som následne zobrazil pomocou tabuliek a grafov.

### Motivácia písania dokumentácie

Na úvod som zadal študentom otázku, ktorá zisťovala ich nálady pri písaní technickej správy. Vytvoril som jednoduchú anketu, ktorá obsahovala jednu otázku v znení: „Aké sú Vaše nálady pri písaní technickej správy“. Negatívna odpoveď bola označená číslom tri a bola popísaná ako nálada, ktorá je rovnaká ako pri práci na zvyšku projektu. **Počet testovaných subjektov bol 56<sup>1</sup>**. Výsledky som zobrazil v grafe 5.1.



Obr. 5.1: Výsledky ankety písania dokumentácie

### Hodnotenie výslednej aplikácie

Po skončení samotnej implementácie som výslednú aplikáciu nechal otestovať viacerým osobám. Na začiatku som im vysvetlil základné informácie o aplikácii a jej účele. Popísal som proces vytvárania projektov a editovania dokumentácie. Po otestovaní každej funkcie, bol subjekt požiadaný, aby zapísal jeho subjektívne hodnotenie na stupnici od jedna do päť ako v predchádzajúcej ankete. Zo získaných dát som nakoniec vypočítal priemerné hodnoty, ktoré som zapísal do tabuľky 5.1.

Úloha	Hodnotenie	Čiastkové hodnotenia
Vytvorenie nového projektu	4.6	2x4, 3x5
Editácia existujúceho projektu	4.4	3x4, 2x5
Zdieľanie projektov administrátormi	4.4	3x4, 2x5
Prehľad prác spolužiakov	5	5x5
Pridávanie obrázkov	4.4	3x4, 2x5
Správa verzií kapitol	4.8	1x4, 4x5
Editácia kapitol projektu	4.8	1x4, 4x5

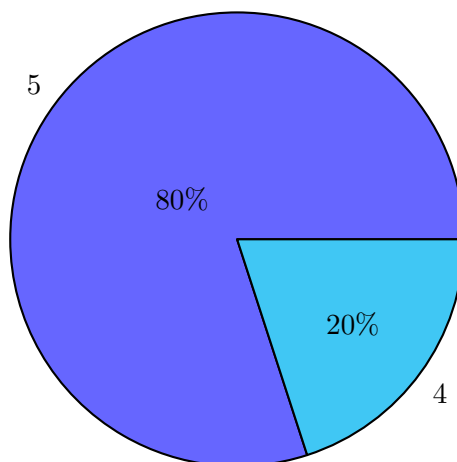
Tabuľka 5.1: Zoznam testovaných úloh

<sup>1</sup>Zdroj prvej otázky: <https://dlxko.typeform.com/to/XQsZrO>

Po otestovaní jednotlivých funkcií som sa zameril na celkový dojem z používania aplikácie. Snažil som sa vyhodnotiť istú hodnotu „user experience“, ktorá je popísaná v sekcii 2.3. Subjektom som položil všeobecné otázky, ktoré sa zameriavajú na rôzne pohľady hodnotenia kvality. Zoznam otázok spolu s priemerným hodnotením je popísaný v tabuľke 5.2. Na koniec celého testovacieho dotazníku som pridal otázku, ktorá zhodnocovala celkový dojem z používania aplikácie. Výsledok je znázornený na obrázku 5.2. Počet testovaných osôb bol 5.

Kritérium	Hodnotenie	Čiastkové hodnotenia
Intuitívnosť	4.8	1x4, 4x5
Design	4.20	4x4, 1x5
Funkčnosť	4.4	3x4, 2x5
Schopnosť motivácie	4.4	3x4, 2x5

Tabuľka 5.2: Zoznam všeobecných otázok



Obr. 5.2: Celkové hodnotenie aplikácie

## 5.4 Zhrnutie testovania

Z výsledkov testovania aplikácie DocBeat je možné vyvodiť záver, že mnou navrhnutá aplikácia má výrazne kladné hodnotenie od testovaných subjektov. Skoro všetci respondenti pri ankete vyjadrovali nechuť pri písaní dokumentácie. Po vyskúšaní mojej aplikácie viaceru z nich zhodnotilo, že by prostredníctvom nej mohlo prísť k zlepšeniu motivácie ako aj výsledného produktu za pomoci dodržiavania stanovených termínov a postupu. Negatívnejšie bol hodnotený návrh užívateľského rozhrania aplikácie, ktorý je jednoduchší oproti iným editorom  $\text{\LaTeX}$  súborov a neponúka tak široké možnosti. Väčšinu funkcií subjekty hodnotili pozitívne a súhlasili s názorom, že ich využitie je pre daný typ aplikácie vhodné a použitie jednoduché. Všetkým testovaným osobám sa rovnako podarilo vytvoriť základnú kostru projektu ako aj vygenerovať výsledný súbor bez potreby pomoci. Napriek tomu som do programu implementoval funkciu zobrazujúcu pomocníka, ktorý informuje užívateľov o jednotlivých funkciách aplikácie.

# Kapitola 6

## Záver

Zadaním mojej bakalárskej práce bolo vytvoriť modernú webovú aplikáciu, ktorá umožňuje vytváranie dokumentácie v nástroji L<sup>A</sup>T<sub>E</sub>X. Aplikácia ponúka všetky potrebné základné nástroje na vytvorenie kompletnej dokumentácie, ktorá splňa všetky požiadavky zadané učiteľom.

Študent je pri písaní práce motivovaný viacerými funkciami, ktoré mu umožňujú sebakritické pozorovanie. Má možnosť sa inšpirovať prácami iných žiakov, pričom sú mu dostupné grafy a informácie, vďaka ktorým môže sám vyhodnotiť usilovnosť iných spolužiakov, na základe ktorej sa môže rozhodnúť čerpať z ich práce. Aplikácia sa snaží v žiakoch vzbudiť hravosť a prirodzenú súťaživosť medzi sebou navzájom.

Administrátor systému môže rovnako zdokonaľovať svoje znalosti v návrhu projektov a využívať projekty alebo samostatné kapitoly od iných užívateľov systému. Rovnako môže definovať štruktúru svojho návrhu, ktorá je rozčlenená na menšie objekty ako termíny, kapitoly, súbory alebo šablóny. Pri hodnotení môže využívať nadobudnuté štatistické údaje k pochopeniu problémov v triede a reflektovaniu nadobudnutých vedomostí do ďalších projektov a požiadaviek, ktoré bude v zadaní projektu definovať.

Z nazbieraných dát sa dá realizovať zaujímavá analýza, o tom, ako študenti pracujú a aké nežiadúce postupy pri tvorbe dokumentácie používajú. Následne môže vzniknúť viacero riešení, ako podobným problémom predchádzať. Moja práca je vďaka týmto prvkom inovatívna a ponúka iný pohľad na bežné a nudné písanie technických správ.

# Literatúra

- [1] Bohuš, P.: *CSS – čo je to a na čo sa používa [online]*. 2015-06-06 [cit. 2016-05-05], [Online; navštíveno 05.05.2016].  
URL <https://cookies.sk/co-je-to-css/>
- [2] Found, J.: *LaTeX 1. – Co to je a proč se ho chtít učit? [online]*. 2013-08-21 [cit. 2016-05-05], [Online; navštíveno 05.05.2016].  
URL <http://programujte.com/clanek/2013080500-latex-1-co-to-je-a-proc-se-ho-chtit-ucit/>
- [3] Lacko, M.: *Čo je PHP a trošku histórie [online]*. 2008-08-14 [cit. 2016-05-05], [Online; navštíveno 05.05.2016].  
URL <http://www.phpblog.sk/clanok/70/co-je-php-a-trosku-historie/>
- [4] Marko, R.: *AJAX - návod pro začátečníky [online]*. 2005-11-06 [cit. 2016-05-07], [Online; navštíveno 07.05.2016].  
URL <http://citron.blueboard.cz/clanek/ajax-navod-pro-zacatecniky>
- [5] Písek, S.: *HTML : začínáme programovat*. Praha : Grada, 2014, ISBN 978-80-247-5059-0.
- [6] Zralý, J.: *JavaScript – 1 – Úvod [online]*. 2004-07-08 [cit. 2016-05-05], [Online; navštíveno 05.05.2016].  
URL [http://www.linuxsoft.cz/article.php?id\\_article=237](http://www.linuxsoft.cz/article.php?id_article=237)



# Prílohy

## Zoznam príloh

<b>A</b>	<b>Obsah CD</b>	<b>39</b>
<b>B</b>	<b>Inštalácia aplikácie</b>	<b>40</b>
<b>C</b>	<b>Manuál základnej práce s aplikáciou</b>	<b>41</b>
<b>D</b>	<b>Výsledná aplikácia</b>	<b>42</b>

# Príloha A

## Obsah CD

- Prezentačný plagát (**Poster.pdf**)
- Prezentačné video (**Advert.mp4**)
- Video z behu programu (**Preview.mp4**)
- Zdrojové súbory aplikácie (**Application.zip**)
- Technická správa (**Documentation.pdf**)

## Príloha B

# Inštalácia aplikácie

1. Nainštalujeme **Apache** a **MySQL** server. Už hotové balíčky obsahujúce oba servery sú dostupné pre Windows - Wamp<sup>1</sup>, OSX - Mamp<sup>2</sup> a Linux - Lamp<sup>3</sup>.
2. V priečinku s Apache serverom nájdeme podpriečinok `htdocs`, kde skopírujeme všetky zdrojové súbory aplikácie.
3. V zdrojovom súbore `application/config/database.php` zadáme údaje slúžiace na pripojenie do databáze.
4. V konfiguračnom súbore `application/config/config.php` upravíme konštantu `base_url`, ktorá bude odkazovať na URL adresu aplikácie. V prípade dodržania postupu bude definovaná ako `http://127.0.0.1:PORT/`. Port vyplníme podľa portu, na ktorom beží Apache server.
5. Do databázy MySQL nahrajeme SQL zdrojový súbor, ktorý sa nachádza v zdrojových súboroch aplikácie v priečinku `database/ProjectCreator.sql`.
6. Do systému je potrebné nainštalovať podporu pre príkazy `pdflatex`, `bibtex` a `texfot`.

---

<sup>1</sup>Balíček Wamp: <https://bitnami.com/stack/wamp>

<sup>2</sup>Balíček Mamp: <https://bitnami.com/stack/mamp>

<sup>3</sup>Balíček Lamp: <https://bitnami.com/stack/lamp>

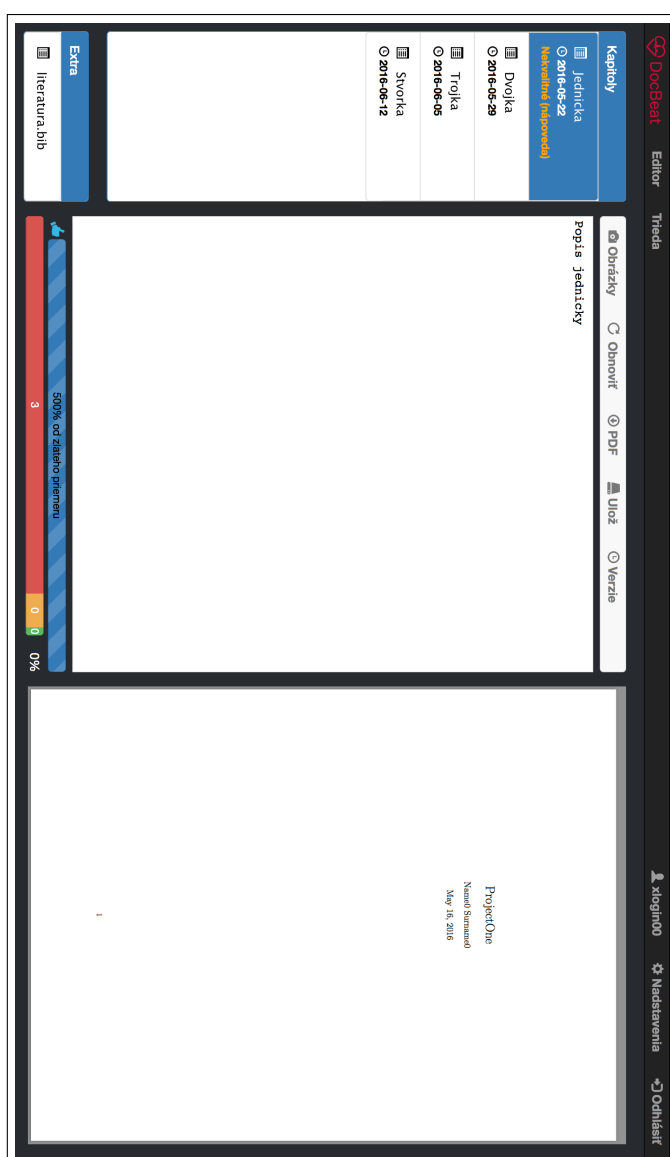
## Príloha C

# Manuál základnej práce s aplikáciou

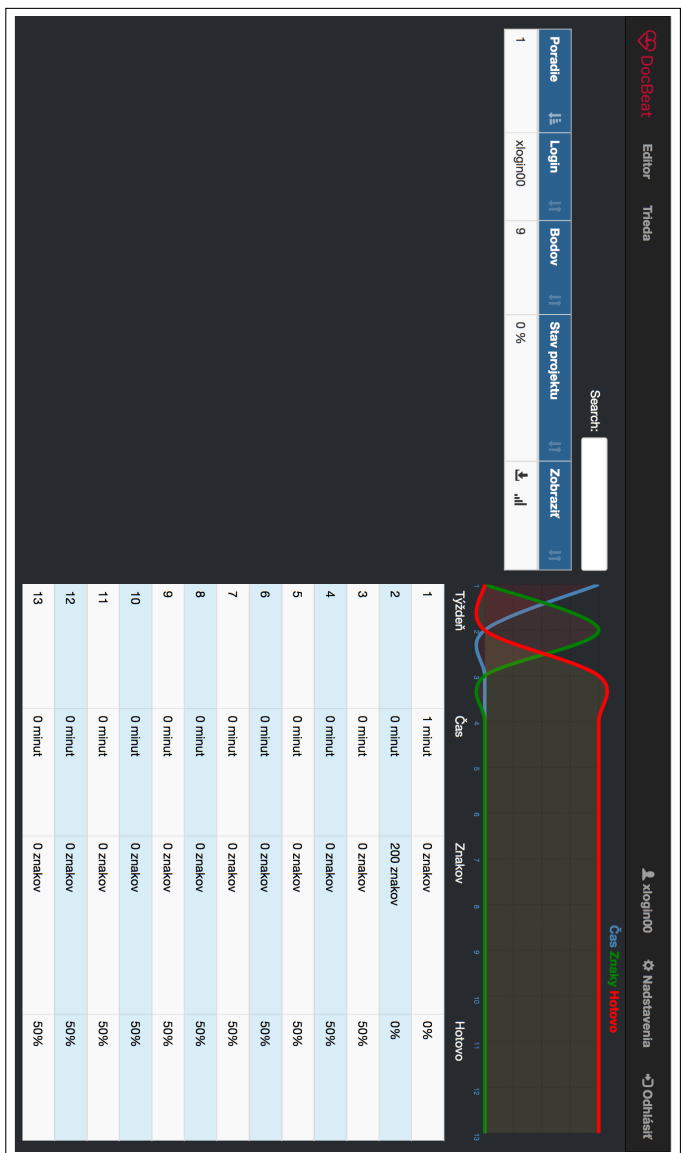
1. Do systému je potrebné sa prihlásiť pod administrátorským účtom (možno použiť už preddefinovaný administrátorský účet `admin:admin`).
2. Z navigačného menu vyberieme položku **Nový projekt**.
  - Do vstupu **Meno projektu** zadáme, meno pod ktorým bude projekt označený
  - **Popis projektu** bude obsahovať stručný popis o projekte.
  - Základná **šablóna projektu** je už preddefinovaná. Môžeme ju upraviť.
  - Zvolíme požadovanú prednastavenú **viditeľnosť** projektu.
  - Kliknutím na tlačítko kalendára zadáme **začiatok projektu**.
  - **JSON zoznam žiakov** obsahuje už predvyplnený formát, ktorý editujeme.
  - Pomocou tlačítka **+** **Nový** vyberieme súbory, ktoré budú priložené k projektu.
  - Kapitoly definujeme za pomoci tlačítka **+** **Nový** pod zoznamom kapitol.
    - **Meno kapitoly** označuje jej názov zobrazený študentovi.
    - **Rozsah (slov)** definuje očakávaný počet slov kapitoly.
    - **Rozsah (citácií)** definuje počet citácií.
    - **Percent pre limit**, hodnota pod ktorou sa zobrazí upozornenie.
  - Vytvorenie projektu prebehne po kliknutí na tlačítko **vytvor nový projekt**.
3. Prihlásime sa pod údajmi, ktoré prišli zadanému žiakovi emailom na jeho štúdiijnú emailovú adresu (`xlogin00@stud.fit.vutbr.cz`). V prípade úspešného vytvorenia projektu a študenta bude umožnené prihlásenie do editora technickej správy.
4. Z panela nástrojov zvolíme funkciu **obnoviť**, čím vytvoríme nový náhľad.
5. Kliknutím na tlačítko **PDF** stiahneme aktuálny súbor.
6. Počas písania textu vidíme **v modrom ukazateli** o koľko percent vedieme od špičiek triedy. **Spodný ukazovateľ** zobrazuje počet žiakov s danou úrovňou dokončenia projektu.
7. V hornom navigačnom paneli zvolíme možnosť **trieda**, kde v prípade, že máme nastavený projekt na verejný vidíme vlastný projekt aj s grafom a tabuľkou.

# Príloha D

## Výsledná aplikácia



Obr. D.1: Editor technickej správy



Obr. D.2: Náhľad prác spolužiakov

DocBee
Domov
Verejně
Uživatelé
1 Martin Pavelka (dixko)

**Nadstavienia**

**Prehľad**  
Prehľad základných informácií o systéme

**Kapitoly a ročníky**

**Editor projekt**  
Editorovanie jednotlivých informácií o projekte

**Vyberte projekt**

**Nový projekt**  
Vytvorenie nového projektu a jeho častí

**Štatistiky**  
Tabuľka štatistík pre každého studenta

**Vyberte užívateľa**

## Kapitoly

Show 10 entries

Meno	Vlasnik	Autor	Projekt	Datum	Upravit
Dvojka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	⋮
Jednicka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	⋮
Stvorka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	⋮
Trojka	Martin Pavelka	Martin Pavelka	ProjectOne	2016-05-16 12:55:41	⋮

Showing 1 to 4 of 4 entries

Previous **1** Next

---

**Ročníky**

Show 10 entries

Meno	Studentov	Vlasnik	Datum	Upravit
Class for ProjectOne	3	Martin Pavelka	2016-05-16 12:55:41	⋮

Showing 1 to 1 of 1 entries

Previous **1** Next

Search:

Obr. D.3: Prehľad položiek v databázy



DocBeet Domov Vneshné Uživatelia Martin Pavetka (dixko) Odišľovať

**Nadstavania**

**Pohľad**  
Přehľad základných informácií o systéme

**Kapitoly a ročníky**

**Editovať projekt**  
Editovanie jednotlivých informácií o projekte

**ProjectOne**

**Nový projekt**  
Vytvorenie nového projektu a jeho častí

**Štatistiky**  
Tabuľka štatistík pre každého študenta

**Výberné uživateľa**

**Popis**

**Meno:** ProjectOne

**Validnosť:** Validný

**Začiatok:** 2016-05-15 00:00:00

**Kapitoly**

Meno	Týždeň	Slow/Chráci	Varovanie	Možnosti
Jednicka	1	100/1	30	☒
Dvojka	2	200/2	20	☒
Trojka	3	300/3	30	☒
Štvorka	4	400/4	40	☒

**Študenti**

Show 10 entries

Search:

Meno	Login	Znazar
Name0 Surname0	xlogjn00	☒
Name1 Surname1	xlogjn01	☒
Name2 Surname2	xlogjn02	☒

Showing 1 to 3 of 3 entries

**Přidat študenta**

Previous **1** Next

Obr. D.4: Editácia existujúceho projektu

DecBeat    Domov    Vernejšie    Uživatelia    **Martin Pavelka (titko)**    Odišľasť

## Štatistiky užívateľa xlogi00

**Nastavenia**

**Prehľad**  
Přehľad základných informácií o systéme

**Kapitoly a ročníky**

**Editovať projekt**  
Editovanie jednotlivých informácií o projekte

**Výberne projekt**

**Nový projekt**  
Vytvorenie nového projektu a jeho časť

**Štatistiky**  
Tabuľka štatistik pre každého študenta

xlogi00

Časové body	Znakové body	Body aktivity	Body spolu
1	200	0	9

Termín	Datum	Jednička	Dvojka	Trojka	Štvorka
Deviaty	Mon May '16 14:04:17 2016 +0200	0	0	0	7
Osmy	Mon May '16 14:04:07 2016 +0200	0	0	0	6
Siedmy	Mon May '16 14:03:57 2016 +0200	0	0	0	5
Šiesty	Mon May '16 14:03:42 2016 +0200	0	0	11	0
Piaty	Mon May '16 14:03:26 2016 +0200	0	0	10	0
Štvrtý	Mon May '16 14:03:11 2016 +0200	0	0	11	0
Tretí	Mon May '16 14:02:56 2016 +0200	0	12	0	0
Druhý	Mon May '16 14:02:38 2016 +0200	0	6	0	0
Prvý	Mon May '16 14:02:13 2016 +0200	1	1	1	1

Obr. D.5: Štatistiky študenta v zobrazení administrátora

DocBeat Domov Verejné užívateľia Martin Pavelka (dixko) -> Odišťať

Meno	Priezvisko	Login	Oprávnenia
Martin	Pavelka	dixko	Superuser
Vita	Baran	baran	Admin
Viktor	Toth	toth	Superuser
Michaela	Sobotkova	sobot	Admin

Meno

Priezvisko

Login

Administrátor

[Vytvor užívateľa](#)

Obr. D.6: Zoznam administrátorov a ich editácia