

**Slovenská technická univerzita**

Fakulta informatiky a informačných technológií

Ilkovičova 3, 842 16 Bratislava 4

---

# **Webový editor Texu - Webtex**

**Tímový projekt**

**Dokumentácia k inžinierskemu dielu**

**Tím č. 15 – code-X**

Bc. Daniel Erban

Bc. Igor Hula

Bc. Kazimír Jaroszewicz

Bc. Radoslav Kontúr

Bc. Radovan Kuka

---

Vedúci tímového projektu: Ing. Tomáš Kramár  
Akademický rok: 2011/2012

# Obsah

<b>ÚVOD.....</b>	<b>1</b>
Účel a rozsah dokumentu .....	1
<b>1 PRVÝ ŠPRINT - ALEXANDRA.....</b>	<b>2</b>
<b>2 DRUHÝ ŠPRINT - BOŽENA .....</b>	<b>3</b>
2.1 Autentifikácia používateľov .....	3
2.1.1 Implementácia .....	3
2.1.2 Testovanie .....	5
2.2 Registrácia používateľov .....	6
2.2.1 Implementácia .....	6
2.2.2 Testovanie .....	7
2.3 Vytváranie projektov .....	7
2.3.1 Implementácia .....	8
2.3.2 Testovanie .....	11
2.4 Dashboard .....	12
2.4.1 Implementácia .....	12
2.4.2 Testovanie .....	13
2.5 Autorizácia.....	14
2.5.1 Implementácia .....	14
2.5.2 Testovanie .....	15
2.6 Úprava súborov .....	16
2.7 Verziovanie súborov .....	16
2.8 Dizajn a user experience .....	16
<b>3 TRETÍ ŠPRINT - CECÍLIA.....</b>	<b>17</b>
3.1 Úprava súborov.....	17
3.1.1 Implementácia .....	17
3.1.2 Testovanie .....	18
3.2 Verziovanie súborov .....	19
3.2.1 Implementácia .....	19
3.2.2 Testovanie .....	25
3.3 Správa kolaborantov .....	26
3.3.1 Implementácia .....	27
3.3.2 Testovanie .....	33
3.4 Dizajn a UX .....	35

<b>4</b>	<b>ŠTVRTÝ ŠPRINT - DAGMAR .....</b>	<b>38</b>
4.1	Zobrazit' strom súborov a naštýlovať ho .....	39
4.2	Editor súborov.....	39
4.3	Kompilácia kódu.....	39
<b>5</b>	<b>PROTOTYP PROJEKTU WEBTEX.....</b>	<b>41</b>
5.1	Čo umožňuje náš prototyp .....	41
5.1.1	Registrácia a autentifikácia používateľov .....	42
5.1.2	Manažment projektov.....	42
5.1.3	Manažment súborov .....	42
5.1.4	Revízie súborov.....	42
5.1.5	Kolaborácia používateľov .....	43
5.1.6	Autorizácia .....	43
5.1.7	Editor projektu .....	44
5.1.8	Stromová štruktúra .....	44
5.1.9	Preklad LaTeX súborov .....	44
5.1.10	Grafický dizajn a rozhranie .....	44
5.2	Zhodnotenie a vízia do budúcnosti .....	45
5.2.1	Zhodnotenie prototypu .....	45
5.2.2	Vízia do budúcnosti .....	45

# Úvod

---

## Účel a rozsah dokumentu

Témou nášho tímového projektu je Webový editor pre TeX. Na webe je dostupných len málo editorov podobného zamerania. Problémom mnohých je to, že nazhŕňajú všetko čo by bežný používateľ očakával. Preto sme existujúce riešenia podrobili z analýze z ktorej pre nás vyplynuli užitočné informácie, akých chýb sa vyvarovať a akú hodnotu pridať.

Cieľom nášho tímového projektu je vytvorenie interaktívneho webového editora, ktorý by umožňoval úpravu zdrojových kódov v TeXu (LaTeX, ConTeXt...), ich kompiláciu a prezeranie výsledného PDF súboru. Jadro funkcionality editora by malo spočívať v podpore kolaborácie. Editor by mal podporovať verziovanie dokumentu, sledovanie zmien, zadávanie poznámok a tým umožniť plnohodnotnú kolaboratívnu tvorbu dokumentov. Webový editor by mal umožňovať tvorbu dokumentu aj mimo prostredia webu a poskytovať jednoduchú obojsmernú synchronizáciu s lokálnymi súbormi. Súčasťou projektu je aj vytvorenie API, aby sa dal editor jednoducho zaintegrovať s niektorým z existujúcich desktopových editorov a rozšíriť ich o možnosť prezerania zmien a poznámok.

Riešenie projektu si vyžaduje širokú škálu zručností a technológií: HTML5&CSS3 (rozhranie); JavaScript, CoffeeScript, jQuery, Backbone.js (pre klientsku časť), Ruby, Rails, Node.js a PostgreSQL pre serverovú časť.

Projekt riešime agilnou vývojovou metodikou Scrum s dôrazom na testami riadený vývoj, kde je vývoj rozdelený na dvojtýždňové úseky, tzv. šprinty. V rámci každého šprintu robíme analýzu, návrh, implementáciu a testovanie zvolených funkcionalít, ktoré sú opísané tzv. príbehmi používateľa. Je zvykom šprinty pre ľahšiu identifikáciu pomenúvať podľa nejakého kľúča, my sme si ženské mená zoradené abecedne.

Predkladaný dokument, ktorý predstavuje dokumentáciu k inžinierskemu dielu, odráža toto rozdelenie a zachytáva prácu v jednotlivých šprintoch počas zimného semestra akademického roku 2011/2012.

# 1 Prvý šprint - Alexandra

---

V prvom šprinte sme venovali štúdiu, pre nás nového, programovacieho jazyka, oboznamovaním sa s Rails frameworkom a prípravou prostredia pre aplikáciu a vývoj softvéru. Riešili sme nasledovné úlohy:

Úloha	Plnenie
<b>Pridať projekt do Jenkinsa</b>	Jenkins budeme používať na automatické nasadzovanie zmien do
<b>Vytvoriť repozitár a skupinu na Gitbuse</b>	na školskom Git serveri sme vytvorili skupinu Codex s repozitárom pre náš projekt WebTex, ktorý sa nachádza tu: <a href="http://gitbus.fiit.stuba.sk/+codex">http://gitbus.fiit.stuba.sk/+codex</a>
<b>Rozbehať vývojové prostredie</b>	rozhodli sme sa pre Ubuntu 11.10 distribúciu, ktorú sme nainštalovali ako virtuálny počítač pod VirtualBox.
<b>Nainštalovať a nakonfigurovať PostgreSQL</b>	je nainštalovaný v rámci vývojového prostredia aj produkčného servera
<b>Nainštalovať a nakonfigurovať passenger</b>	nainštalovaný na serveri, slúži na jednoduché nasadzovanie aplikácie
<b>Vygenerovať základnú kostru Rails aplikácie</b>	vygenerovaná príkazom „rails new webtex“
<b>Naštudovať ako funguje SYNCTEX</b>	túto úlohu sme si nechali na neskorší šprint, kedy budeme riešiť zobrazovanie vygenerovaných PDF dokumentov. SyncTeX slúži na synchronizáciu medzi zdrojovým TeX dokumentom a vygenerovaným PDF dokumentom
<b>Naštudovať ako funguje Git note</b>	Naštudovali sme Git note a uvažovali sme výhody a možnosti použitia v našom rpojekte
<b>Porovnať Codemirror a ACE</b>	rozhodli sme sa pre CodeMirror na základe funkcií, ktoré poskytuje
<b>Rozdistribúovať konvencie programovania pre Ruby a Git</b>	rozhodli sme sa ísť podľa štýlu, ktorý nám navrhol vedúci a tento štýl je bližšie špecifikovaný v Dokumentácii k riadeniu

## 2 Druhý šprint - Božena

---

V druhom šprinte sme riešili nasledovné úlohy:

### 2.1 Autentifikácia používateľov

Názov user story	Popis
<b>Autentifikácia používateľov</b>	Ako používateľ sa chcem prihlásiť, aby som mohol mať privátne projekty, ktoré sú viditeľné len mne

V rámci úlohy sme riešili nasledovné podúlohy:

- Nainštalovať a nastudovať DEVISE
- Nakonfigurovať DEVISE
- Cucumber testy

#### 2.1.1 Implementácia

Implementácia má obsahovať formulár, ktorým sa používateľ prihlási do systému. Mal by použiť kombináciu e-mailu a hesla. Pre autentifikáciu sme použili gem Devise. Je to modulárne autentifikačné riešenie pre Rails framework. Po inštalácii samotného gem-u sme vygenerovali model (users.rb), konfiguračný súbor a views pre všetky akcie. Model sme nakonfigurovali na ukladanie šifrovaných hesiel do databázy a validáciu používateľov, registráciu, zapamätanie si používateľov cez cookies, sledovanie prihlásení a obnovenie hesla. Samotný zdrojový kód vyzerá nasledovne:

```
Config/initializers/devise.rb
```

```
Devise.setup do |config|
  config.mailer_sender = "please-change-me-at-config-initializers-devise@example.com"
  require 'devise/orm/active_record'
  config.case_insensitive_keys = [ :email ]
  config.strip_whitespace_keys = [ :email ]
  config.stretches = Rails.env.test? ? 1 : 10

  config.use_salt_as_remember_token = true
  config.reset_password_within = 2.hours
  config.sign_out_via = :get
```

```
End
```

## app/models/devise.rb

```
1 class User < ActiveRecord::Base
2   has_many :projects
3
4   # Include default devise modules. Others available are:
5   # :token_authenticatable, :encryptable, :confirmable, :lockable, :timeoutable and
   # :omniauthable
6   devise :database_authenticatable, :registerable,
7         :recoverable, :rememberable, :trackable, :validatable
8
9   # Setup accessible (or protected) attributes for your model
10  attr_accessible :email, :password, :password_confirmation, :remember_me
11 end
```

Vygenerované views obsahovali viaceré obrazovky, najdôležitejší pre prihlasovanie bol view pre sign-in:

## views/devise/sessions/new.html.erb

```
1 <h2>Sign in</h2>
2
3 <%= form_for(resource, :as => resource_name, :url => session_path(resource_name)) do
   |f| %>
4   <div><%= f.label :email %><br />
5   <%= f.email_field :email %></div>
6
7   <div><%= f.label :password %><br />
8   <%= f.password_field :password %></div>
9
```

```

10 <% if devise_mapping.rememberable? -%>
11   <div><%= f.check_box :remember_me %> <%= f.label :remember_me %></div>
12 <% end -%>
13
14 <div><%= f.submit "Sign in" %></div>
15 <% end %>
16
17 <%= render :partial => "devise/shared/links" %>

```

## 2.1.2 Testovanie

Testovali sme pomocou Cucumber testov, na preverenie autentifikácie boli použité nasledovné scenáre:

Kroky scenára	Akcia scenára
<b>Logging in with non-existing account</b>	
Given I am logged in And no user with login <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> exists When I login as <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> with password „password“ Then I should not be logged in And I should see that i have entered wrong account information	Po navigovaní na prihlasovaciu stránku a vyplnení údajov pre neexistujúce konto sa zobrazí upozornenie „Invalid email or password“
<b>Logging in with existing account</b>	
Given i am not logged in And a user with pin <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> and password „secret“ exists When i login as <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> with password „secret“ Then i should be logged in	Po navigovaní na prihlasovaciu stránku a vyplnení formulára s platnými údajmi sa používateľ prihlási a stránka ho presmeruje na domovskú stránku aplikácie



## 2.2 Registrácia používateľov

Názov user story	Popis
<b>Registrácia používateľov</b>	Ako používateľ sa chcem registrovať, aby som sa mohol prihlásiť a mať výhody z toho vyplývajúce

### 2.2.1 Implementácia

Registrácia používateľa bola implementovaná pomocou rovnakého gem-u ako autentifikácia – Devise. Bol nakonfigurovaný aby prijímal ako meno validný e-mail a heslo s minimálne šiestimi znakmi.

Vygenerovaný bol nasledovný view:

#### views/devise/registrations/new.html.erb:

```
<h2>Sign up</h2>

<%= form_for(resource, :as => resource_name, :url =>
  registration_path(resource_name)) do |f| %>
  <%= devise_error_messages! %>

  <div><%= f.label :email %><br />
  <%= f.email_field :email %></div>

  <div><%= f.label :password %><br />
  <%= f.password_field :password %></div>

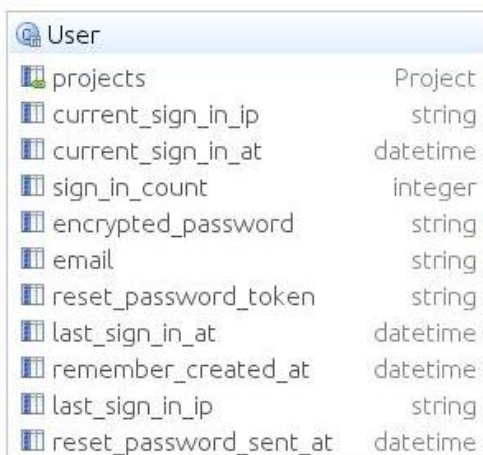
  <div><%= f.label :password_confirmation %><br />
  <%= f.password_field :password_confirmation %></div>

  <div><%= f.submit "Sign up" %></div>

<% end %>

<%= render :partial => "devise/shared/links" %>
```

Vytváral sa tu aj model pre user-a, ktorý vyzerá nasledovne:



Attribute	Type
projects	Project
current_sign_in_ip	string
current_sign_in_at	datetime
sign_in_count	integer
encrypted_password	string
email	string
reset_password_token	string
last_sign_in_at	datetime
remember_created_at	datetime
last_sign_in_ip	string
reset_password_sent_at	datetime

## 2.2.2 Testovanie

Testovali sme pomocou Cucumber testov, na preverenie registrácie boli použité nasledovné scenáre:

Kroky scénara	Akcia scénára
<b>Registering</b>	
Given no user with login <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> exists When I register as <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> and choose my password to be secret And I login as <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> with password "secret" Then I should be logged in	Po navigovaní na Sign-up stránku a vyplnení formulára údajmi sa zobrazí informácia o úspešnej registrácii a automaticky sa používateľ prihlási.
<b>Registering with already taken account name</b>	
Given a user with login <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> already exists When I register as <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> and choose my password to be "secret" Then I should not be registered And I should see that the username <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> is already taken	Po registrovaní s údajmi používateľa, ktorý už existuje v systéme sa zobrazí chybové upozornenie.

## 2.3 Vytváranie projektov

Názov user story	Popis
<b>Vytváranie projektov</b>	Ako používateľ si chcem vytvoriť projekt, aby som mohol logicky zhlukovať dokumenty, ktoré budem písať

Vytváranie projektov v sebe zahŕňa tieto podúlohy:

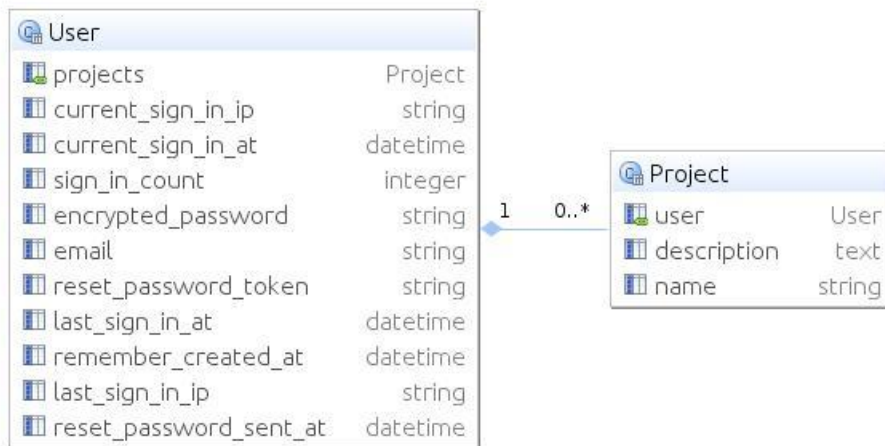
- Navrhnuť a vytvoriť tabuľky pre vytváranie projektov
- Inštalácia GRIT gemu

- Ruby interakcia s Gritom
- Cucumber testy

Táto feature je úzko prepojená s Dashboardom, kde sa zobrazujú samotné projekty.

### 2.3.1 Implementácia

Bola vytvorená tabuľka pre projekty (viď obrázok), v ktorej sa ukladajú potrebné dáta.



Ďalej bol nainštalovaný gem Grit, ktorý dáva objektovo-orientovaný prístup do Git repozitárov pomocou Ruby. Pri vytváraní projektu sa prostredníctvom tohto gemu vytvorí na súborovom systéme repozitár s cestou: `git_path/ID_USER/ID_Project`, pričom `git_path` je nastavený v `config.yml` súbore a `id_user` je ID aktuálne prihláseného používateľa. Celú funkcionlitu má na starosti controller `projects_controller.rb`:

#### app/controlles/projects\_controller:

```

1 class ProjectsController < ApplicationController
2   include ApplicationHelper
3
4   before_filter :find_project, :only => [:show, :edit, :update, :destroy]
5
6   def show
  
```

```
7   authorize! :read, @project
8   end
9
10  def new
11    @project = Project.new
12  end
13
14  def create
15    @project = Project.create(params[:project].merge(:owner => current_user))
16
17    if @project.save
18      flash[:notice] = t('project.created', :name => @project.name)
19      redirect_to dashboard_user_path
20    else
21      render :action => :new
22    end
23  end
24
25  def edit
26    authorize! :edit, @project
27  end
28
29  def update
30    if @project.update_attributes(params[:project])
31      flash[:notice] = t('project.updated', :name => @project.name)
```

```
32     redirect_to dashboard_user_path
33   else
34     render :action => :edit
35   end
36   authorize! :edit, @project
37 end
38
39 def destroy
40   if @project.destroy
41     flash[:notice] = t('project.deleted', :name => @project.name)
42     redirect_to dashboard_user_path
43   else
44     flash[:alert] = t('project.not_deleted', :name => @project.name)
45     redirect_to dashboard_user_path
46   end
47   authorize! :edit, @project
48 end
49
50 private
51
52 def find_project
53   @project = Project.find(params[:id])
54 end
55 end
```

Pomocou metód create, show, edit a destroy sa vytvorí nový projekt, zobrazia existujúce projekty pre aktuálne prihláseného používateľa, zmení sa názov projektu a zmaže sa celý projekt.

### 2.3.2 Testovanie

Pri testovaní sa predpokladá, že používateľ je prihlásený

Kroky scenára	Akcia scenára
<b>Creating project</b>	
When I go to my dashboard When I follow link to create new project And I create a new project called "iit.src" Then I should have a project "iit.src" among my projects	Po navigovaní do dashboard, kliknem na tlačítko "new project", vyplním príslušné údaje a kliknem na "Create project". V databáze sa vytvorí záznam a na dashboard sa zobrazí mnou vytvorený projekt
<b>Editing project</b>	
Given I have a project "iit.src" And I edit the project When I change its name to "iit.src 2012" and its description to "iit 2009" Then I should be on my dashboard And I should see that I have a project "iit.src 2012"	Na dashboarde mám vytvorený projekt "iit.src". po kliknutí na link "edit" môžem meniť názov projektu a popis. Keď ho zmením danými údajmi a kliknem na tlačítko "Update Project", v dashboarde vidím, že projekt je zmenený.
<b>Delete project</b>	
Given I have a project "iit.src" And I am on the dashboard When I delete the project Then I should have no project "iit.src"	V dashboarde mám projekt "iit.src". Po kliknutí na tlačítko "delete" sa daný projekt vymaže a už nie je zobrazený v dashboarde.

## 2.4 Dashboard

Názov user story	Popis
<b>Dashboard</b>	Ako používateľ chcem vidieť celú pracovnú plochu na jednom mieste – dashboard, aby som mal rýchly prehľad o mojich projektoch a súboroch

### 2.4.1 Implementácia

Dashboard je východiskový bod po prihlásení používateľa. Sú v ňom prehľadne zobrazené všetky projekty a ich súbory, na ktorých používateľ pracuje.

Implementovaný je vo view `dashboard.html.erb`, využíva funkcionality vytvárania projektov z predošlej úlohy.

**app/views/users/dashboard.html.erb:**

```
1 <h3>Dashboard</h3>
2
3 <table title="dashboard" border="1" cellpadding="3" cellspacing="0" id="dashboard">
4   <tr>
5     <th>Name</th>
6     <th>Owner</th>
7     <th>Last modified</th>
8     <th></th>
9     <th></th>
10  </tr>
11  <% @projects.each do |project| %>
12    <tr>
13      <td><%=link_to project.name, project_path(project)%></td>
14      <td><%= project.owner.email %></td>
15      <td><%= time_ago_in_words (project.updated_at) %> ago</td>
```

```

16 <td><%= link_to 'Edit', edit_project_path(project) %></td>
17 <td><%= link_to 'Delete', project, :confirm => 'Are you sure you want to delete
this project?', :method => :delete %></td>
18 </tr>
19 <% end %>
20 </tr>
21 </table>
22
23 <br>
24 <%= link_to 'New Project', new_project_path %>

```

## 2.4.2 Testovanie

Kroky scenára	Akcia scenára
<b>Seeing dashboard after login</b>	
Given I am logged out When I log in Then I should be on my dashboard page	Po prihlásení sa zobrazí dashboard.
<b>Seeing list of projects and their files in dashboard</b>	
Given I have the following projects and files: Project: tpcup Files: main.tex, overview.jpg Project: iitsrc Files: iitsrc.tex, bibliography.bib When I go to my dashboard Then I shuld see that I have project tpcup And I shoyld see that project tpcup has files main.tex, overview.jpg Then I should see that I have project iit.src	V dasbhoarde mám zoznam mojich projektov spolu so súbormi, ktoré k nim patria.



Seeing last change for a project and each file	
Given I am logged in Given I have project iitsrc And the project has file main.tex which was last modified 1 days ago And the project has file bibliography.bib which was last modified 3 days ago When I go to my dashboard Then I should see that the project iitsrc was last modified 1 day ago And I should see that the file main.tex was last modified 1 day ago And I should see that the file bibliography.bib was last modified 3 days ago	V dashboarde je zoznam projektov so súbormi a údajmi o poslednej zmene súboru/projektu.

## 2.5 Autorizácia

Názov user story	Popis
<b>Autorizácia</b>	Ako používateľ by som mal byť jediný, ktorý vidí moje súbory, aby som obmedzil prístup k projektom, ktoré vytváram.

V rámci tejto úlohy boli riešené nasledovné podúlohy:

- Nainštalovať a naštudovať CanCan
- Definovať CanCan ability
- Zavolanie CanCan pred zobrazením projektu
- Cucumber testy

### 2.5.1 Implementácia

Pri autorizácii sa využíva gem CanCan, ktorý obmedzuje prístup používateľov k prostriedkom. Zabezpečujeme aby používateľ nemohol zmeniť URL a touto zmenou by získal prístup k inému projektu. Všetky prístupy sú riešené cez jednu Ability triedu. Táto trieda je nastavená tak, aby zobrazovala používateľom iba tie projekty, ktorých sú vlastníkom. V `application_controller.rb` musí

byť definovaná akcia, ktorá sa vykoná v prípade zamietnutia CanCanom, riešime to pomocou presmerovania na public stránku forbidden 403. Ďalej v každom controlleri musí byť v príslušnej metóde povedané, že má byť volaný CanCan. To sa zabezpečí napríklad volaním:

```
authorize! :edit, @project
```

Kód hlavnej triedy CanCanu je definovaný nižšie, tento kód však bude nutné pri vývoji produktu prispôbovať požiadavkám.

#### app/models/ability.rb:

```

1 class Ability
2   include CanCan::Ability
3
4   def initialize(user)
5     user ||= User.new
6     can [:read, :edit], Project do |project|
7       project.owner == user
8     end
9   end
10 end

```

## 2.5.2 Testovanie

Testovanie sme vykonávali pomocou Cucumber testov.

Kroky scenára	Akcia scenára
<b>Only author can see his projects</b>	
Given I am logged in as <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> And I have a project "iit.src" And I go to my dashboard Then I should be able to see the project "iit.src" When I relogin as <a href="mailto:mbielik@gmail.com">mbielik@gmail.com</a> And I go to my dashboard	Pri prihlásení sa dashboard zobrazuje iba projekty aktuálne prihláseného používateľa.

Then I should not see the project “iit.src”	
<b>Projects are visible to their authors</b>	
Given I am logged in as <a href="mailto:tkramar@gmail.com">tkramar@gmail.com</a> And I have a project “iit.src” And there is another user and she has a project “papers” When I go to the “iit.src” project page Then I should be able to see the project When I go to the “papers” project page Then I should not be able to see the project	Pri manuálnom zadaní projektu do linku, na ktorý nemá aktuálny používateľ právo, sa zobrazí error 403 – forbidden.

## 2.6 Úprava súborov

Ako používateľ, chcem mať možnosť upravovať súbory, aby som mohol vytvárať dokumenty na webe, bez nutnosti inštalovať TEX na mojom počítači.

V čase písania dokumentácie ešte táto úloha nebola vyriešená z dôvodu personálnych problémov v tíme, presun do ďalšieho šprintu.

## 2.7 Verziovanie súborov

Táto funkcionálna bola presunutá do ďalšieho šprintu z dôvodu priamej závislosti na vytváraných súboroch. Vytváranie súborov bolo tiež riešené v ďalšom šprinte.

## 2.8 Dizajn a user experience

Ako prevádzkovateľ editora chcem, aby editor dobre vyzeral a pohodlne sa používal, aby ľudia za neho boli ochotní platiť.

V čase písania dokumentácie ešte táto úloha nebola vyriešená z dôvodu personálnych problémov v tíme, presun do ďalšieho šprintu.

## 3 Tretí šprint - Cecília

---

### 3.1 Úprava súborov

Názov user story	Popis
Úprava súborov	Ako používateľ chcem mať možnosť upravovať súbory, aby som mohol vytvárať dokumenty na webe pohodlne, bez nutnosti inštalovať Tex na mojom počítači.

Úprava súborov je jedna z ťažísk aplikácie WebTex. V tomto user story bolo treba implementovať funkcionality vytvárania, mazania a úpravy súboru a dbať a spolupracovať s feature Verziovania súborov.

- Cucumber TEST
- Vytvorenie súboru
- Editovanie mena súboru
- Mazanie súboru
- Editovanie obsahu súboru
- Ukladanie a verziovanie

#### 3.1.1 Implementácia

Naša aplikácia je riešená tak že každému používateľovi prislúcha v súborovom systéme jedna zložka nazvaná jeho ID číslom. V zložke má projekty ktorých je vlastníkom, ktoré sú pomenované ID číslom projektu. V každej zložke projektu môžeme mať rôzne súbory.

Ťažisko funkcionality je v dvoch súboroch:

- **app/controllers/documents\_controller.rb**
- **app/models/documents.rb**

Zdrojový kód týchto súborov je umiestnený v kapitole 3.2 Verziovanie súborov. Obe funkcionality boli ako Úprava súborov a Verziovanie dokumentov boli implementované do týchto dvoch súborov a boli implementované v spolupráci oboch zodpovedných programátorov.

### 3.1.2 Testovanie

Testovanie sme vykonávali pomocou Cucumber testov.

Kroky scenára	Akcia scenára
<b>BACKGROUND</b>	
Given I am logged in And I have project "iit.src" And The project "iit.src" has file "main.tex" with content """" This is content for file """"	Existuje projekt „it.src“ so súborom „main.tex“.
<b>Scenario: Creating new file</b>	
When I go to project detail And I create a new file called "overview.jpg" Then I should have a file "overview.jpg" among my files	Keď vytvorím súbor s názvom „overview.jpg“ tak ho neskôr budem vidieť v projekte.
<b>Scenario: Opening the file</b>	
When I open the file "main.tex" Then I should see that the file "main.tex" has content """" This is content for file """"	Keď otvorím súbor, budem vidieť jeho obsah.
<b>Scenario: Editing the file content</b>	
When I open the file "main.tex" And I change the content of file "main.tex" to """" Edited text. """" And I open the file "main.tex" Then I should see that the file "main.tex" has content """" Edited text.	Keď do súboru „main.tex“ vpišem pri editácii text, budem vidieť tento tex aj keď súbor znovu otvorím.

""	
<b>Scenario: Delete file</b>	
When I delete the file "main.tex" Then I should have no file "main.tex"	Keď zmažem súbor, potom ho už nebudem vidieť medzi mojimi súbormi.
<b>Scenario: Edit file name</b>	
When I rename the file "main.tex" to "iitsrc.tex" Then I should have no file "main.tex" But I should have file "iitsrc.tex"	Keď premenujem súbor, tak sa zmení jeho meno.

## 3.2 Verziovanie súborov

Názov user story	Popis
<b>Verziovanie súborov</b>	Ako používateľ chcem vidieť verzie dokumentu, aby som videl ako sa dokument vyvíjal v čase

V rámci úlohy sme riešili nasledovné podúlohy:

- Zobrazenie zoznamu revízií
- Zobrazenie konkrétnej revízie
- Cucumber testy

Verziovanie dokumentov projektu je jeho dôležitá súčasť v kontexte vývoja dokumentu čase. Jednotlivé verzie dokumentu sa vytvoria v prípade, že používateľ, ktorý je na tomto dokumente oprávnený pracovať, upraví tento dokument a uloží ho. Úlohou verziovania je pre daný projekt zobrazit' všetky k nemu prislúchajúce verzie a takisto zobrazit' aj konkrétnu verziu dokumentu. Túto záležitosť sme sa rozhodli riešiť pomocou verziovacieho nástroja Git, konkrétne v projekte WebTex bude využitý gem Grit.

### 3.2.1 Implementácia

Naša aplikácia je riešená tak že každému používateľovi prislúcha v súborovom systéme jedna zložka nazvaná jeho ID číslom. V zložke má projekty ktorých je vlastníkom, ktoré sú pomenované

ID číslom projektu. V každej zložke projektu môžeme mať rôzne súbory. Revízie týchto súborov sú uložené v gite takže žiadne fyzické kópie na disku. Práve týmito revíziami sa zaoberá táto implementácia. Hlavné ťažisko funkcionality je `documents_controller.rb` a v modeli `documents.rb`

**app/controllers/documents\_controller.rb:**

```
1 class DocumentsController < ApplicationController
2   before_filter :find_document
3
4
5   def revisions
6
7   end
8
9   def show
10
11 end
12
13 def revision
14   @document = @project.find_file_with_sha(params[:file_name], params[:revision_id])
15   @document.at_revision(params[:revision_id])
16 end
17
18 def save
19   @document.save(params[:content])
20   redirect_to show_document_path(params[:project_id], params[:project_name],
params[:file_name])
21 end
22
```

```

23 def destroy
24   @document.delete
25   redirect_to project_path(params[:project_id])
26 end
27
28 def update
29   @document.update(params[:new_file_name])
30   redirect_to show_document_path(params[:project_id], params[:project_name],
31     params[:new_file_name])
31 end
32
33 def find_document
34   @project = Project.find(params[:project_id])
35   @document = @project.find_file_with_sha(params[:file_name], params[:revision_id])
36 end
37 End

```

Práca s dátami je realizovaná pomocou modelu pre dokumenty.

#### **app/models/documents.rb:**

```

1 class Document
2
3   def initialize(options)
4     if(options.length == 3)
5       @file_name = options[:file_name]
6       @repository = options[:repository]
7       @file_path = options[:git_project_path]

```



```
8     else
9         @file_name = options[:file_name]
10        @repository = options[:repository]
11        @sha = options[:sha]
12        @file_path = options[:git_project_path]
13    end
14 end
15 attr_reader :file_name
16
17 def at_revision(sha)
18     Document.new :file_name => @file_name, :repository => @repository, :sha => sha,
19 :git_project_path => @file_path
20
21 def revision_content
22     tree = @repository.tree(@sha)
23     file = tree/"#{@file_name}"
24     file.data
25 end
26
27 def revisions
28     @repository.log("master", @file_name).reverse
29 end
30
31 def name
32     return @file_name
```

```
33 end
34
35 def content
36   file_path_with_name = File.join(@file_path, @file_name)
37   file = File.open(file_path_with_name, "r")
38   return file.read
39 end
40
41 def save(contents)
42   commit(@file_name, "Add file #{@file_name}") do |file_path|
43     File.open(file_path, "w") do |f|
44       f << contents
45     end
46   end
47 end
48
49 def update(new_file_name)
50   old_file_path = File.join(@file_path, @file_name)
51
52   commit(new_file_name, "Rename file #{@file_name} to #{new_file_name}") do
|new_file_name|
53     File.rename(old_file_path, new_file_name)
54   end
55 end
56
57 def delete
```

```

58   commit(@file_name, "Delete file #{@file_name}") do |file_path|
59     File.delete(file_path)
60   end
61 end
62
63 def last_modified_date
64   @last_modified_date ||= Time.now - @repository.log("master", @file_name,
:max_count => 1).last.committed_date
65 end
66
67 private
68 def commit(file, message)
69   file_path = File.join(@file_path, file)
70
71   yield(file_path)
72
73   Dir.chdir(@file_path) do
74     @repository.add file
75     @repository.commit_all "#{message}"
76   end
77 end
78 end

```

### Popis najdôležitejších častí kódu:

```

class DocumentsController < ApplicationController
  before_filter :find_document
  #toto vytvori instanciu Document s nazvom @document pomocou metodu modelu
Project
  #find_file_with_sha(params[:file_name], params[:revision_id])

```

```

def revisions
  #tuto je prepojenie view document/revisions.html.erb s modelom, ktory posle
  premennu @document. to je instancia
  #modelu Document.ako mozes vidiet v tom viewe sa potom deje cely vypis
  zoznamu revizii s tym kto commitoval, kedy...
end

def show

end

def revision
  @document = @project.find_file_with_sha(params[:file_name],
  params[:revision_id])
  @document.at_revision(params[:revision_id])
  #toto komunikuje s view document/revision.html.erb, at_revision je fcia
  modelu , ktora nastavi aktualnu sha instancii @document
  #a obsah suboru sa vypise @document.revision_content
end

```

Revízie súborov sú závislé na modeli projektov, ktorý pacuje s dátami a pri vytváraní projektu vytvára príslušný repozitár. Contoller revízií teda musí byť dobre prepojený s týmto modelom. Potom už je potrebný iba view na zobrazovanie týchto revízií v prehliadači.

### 3.2.2 Testovanie

Testovanie sme vykonávali pomocou Cucumber testov.

Kroky scenára	Akcia scenára
<b>BACKGROUND</b>	
Given I am logged in as "tkramar@gmail.com" And I have project "iit.src" And The project "iit.src" has file "main.tex" with content """" This is the first version of the document """" And I open the file "main.tex" And I change the content of file "main.tex" to """" This is the second version of the document """"	Zmením 3 krát dokukemnt „main.tex“

<p>And I change the content of file "main.tex" to</p> <p>""""</p> <p>This is the final version of the document</p> <p>""""    """"</p> <p>    This is the final version of the document</p>	
<p><b>Scenario: Browsing document revisions</b></p>	
<p>When I go to "main.tex" revisions page</p> <p>Then I should see that "main.tex" has 3 revisions</p>	<p>Keď pôjdem na stránku revízií súboru uvidím 3 revízie</p>
<p><b>Scenario: Viewing document's older revision</b></p>	
<p>When I go to "main.tex" revisions page</p> <p>And I open the first revision of file "main.tex"</p> <p>Then I should see that the contents of that file at that revision were</p> <p>""""</p> <p>This is the first version of the document</p> <p>""""</p>	<p>Keď som na revíziách súboru „main.tex“ a kliknem na prvú revíziu obrazí sa mi dokument v tom čase, čiže jeho prvá verzia.</p>

### 3.3 Správa kolaborantov

Názov user story	Popis
<b>Správa kolaborantov</b>	Ako používateľ chcem spravovať kolaborantov projektu, ktorí k nemu budú mať prístup, aby som na projekte mohol pracovať s inými používateľmi.

V rámci úlohy sme riešili nasledovné podúlohy:

- Cucumber TEST
- Rozposielanie emailov kolaborantom
- Vytvorenie prepojovacej tabuľky medzi users a projects
- Úprava DASHBOARDU
- Pridanie kolaborantov
- Výpis kolaborantov
- Mazanie kolaborantov

Správa kolaborantov a vôbec kolaborácia samotná je jedným zo základných pilierov našej aplikácie WebTex. Správa kolaborantov by mala umožňovať pridávať jednotlivých spolupracovníkov ku konkrétnym projektom. Samozrejme by malo byť umožnené kolaboranta aj zmazať. Táto funkcionality v sebe nezahŕňa len prácu s nejakou tabuľkou, je potrebné aj autorizovať prístup ku konkrétnym projektom pomocou CanCan. Dôležité je aby pridávanie emailov bolo jednoduché a rýchle. Vyriešili sme to pomocou zoznamu emailových adries. Používateľ jednoducho zadá emailové adresy používateľov, ktorých chce zadať medzi spolupracovníkov na projekte.

Samozrejme tu sa hneď vynára niekoľko špecifik pri implementácii. Správanie používateľa totiž treba ošetriť. Ak by náhodou zadal zlú adresu, aby sa mu zobrazila chybová správa. Kolaborant by nemal mať možnosť odstrániť vlastníka projektu. Taktiež by nemalo byť možné pridať 2x toho istého kolaboranta k tomu istému projektu. Je to teda množstvo ošetrení ktoré treba zohľadniť.

Významným faktom pri pridávaní kolaborantov, je aj to aby boli informovaný o tom že boli pridaný. Na to využívame gem Mailer, spolu s definovaným telom emailovej správy kde sú vložené informácie ako email kolaboranta a projekt na ktorý bol pridaný ako kolaborant. Bolo treba nakonfigurovať aj smtp. Zatiaľ využívame na rozosielanie emailov účet Gmail s názvom [webtexmailer@gmail.com](mailto:webtexmailer@gmail.com). Keď bude projekt umiestnený na vlastnú doménu, nie je problematické zmeniť konfiguračný súbor.

### 3.3.1 Implementácia

Implementácia pozostávala z niekoľkých častí. Najskôr bolo treba vytvoriť prepojovaciu tabuľku medzi používateľmi a projektmi. Toto bolo zabezpečené novou migráciou.

#### db/migrate/20111119220709\_projects\_users.rb

```
1 class ProjectsUsers < ActiveRecord::Migration
2   def change
3     create_table :projects_users, :id => false do |t|
4       t.integer :project_id, :null => false
5       t.integer :user_id, :null => false
6     end
  end
end
```

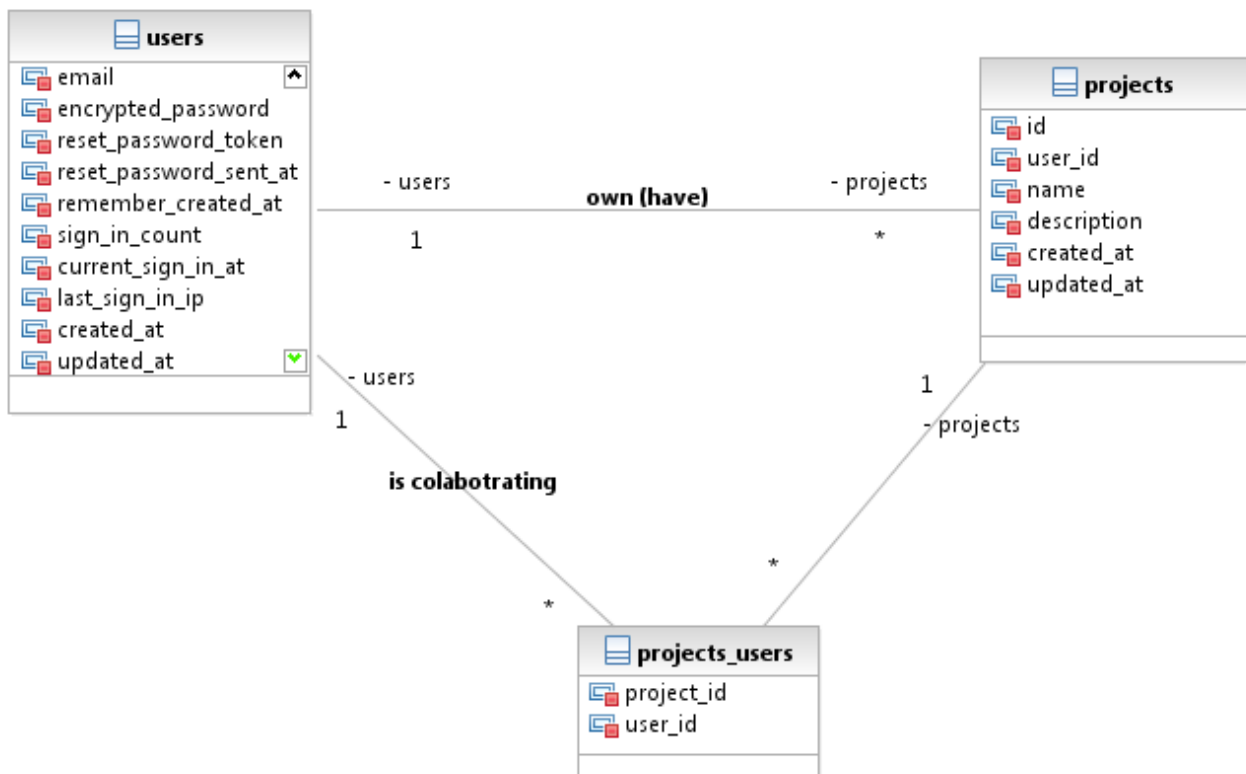
7

```
8   add_index :projects_users, [:project_id, :user_id], :unique => true, :name =>
9   'by_project_and_user'
```

```
9   end
```

```
10 end
```

Index je tabuľke pridelený na riadok práve preto aby používateľ nemohol dvakrát kolaborovať na tom istom projekte, to by nedávalo zmysel.



Obrázok 1 – súčasný logický dátový model projektu WebTex

Dátový model sa možno zdá zatiaľ jednoduchý, ale množstvo záležitostí je riešené Gitom a samotné projekty a súbory používateľov sú uložené v súborovom systéme či už lokálneho stroja alebo pri reálnom nasadení aplikácie servera.

Ďalej bolo treba v implementácii nainštalovať gem Mailer a nastaviť mailového klienta. Nastavili sme gmailový účet. Túto konfiguráciu bolo treba realizovať súbore **/config/enviroments/development.rb** . Tu bolo potrebné pridať nasledovné riadky:

```

17 config.action_mailer.raise_delivery_errors = true
19 config.action_mailer.delivery_method = :smtp
20 config.action_mailer.smtp_settings = {
21   :address          => "smtp.gmail.com",
22   :port             => 587,
23   :domain           => 'gmail.com',
24   :user_name        => 'webtexmailer',
25   :password         => 'aaabbb111',
26   :authentication   => 'plain',
27   :enable_starttls_auto => true }

```

Po pridávaní kolaboranta do prepojujacej tabuľky sa volá funkcia zo súboru **app/mailers/user\_mailer.rb** , tu sá nasetujú premenné, ktoré budú vystupovať v uvítacom maile.

```

1 class UserMailer < ActionMailer::Base
2   default from: "webtexmailer@gmail.com"
3
4   def welcome_email(user, project)
5     @user = user
6     @project = project
7     mail(:to => user.email, :subject => "You was add as collaborator")
8   end
9 end

```

Šablóna pre email je vlastne naštýlovaním správy do HTML. Je umiestnená v súbore: **App/views/user\_mailer/wellcome\_email.html.erb**:

```

1 <!DOCTYPE html>

```



```

2 <html>
3   <head>
4     <meta content="text/html; charset=UTF-8" http-equiv="Content-Type" />
5   </head>
6   <body>
7     <h2>Hello <%= @user.email %> you were added as collaborator for project named
" <%= @project.name %>" </h2>
8     <p>Thanks for joining and have a great day! </p>
9   </body>
10 </html>

```

Hlavné ťažisko vytvorenia kolaboranta je umiestnené v modeli project:

#### **app/models/project.rb:**

```

26 def add_collaborators_by_email(emails)
27   all_emails = emails.gsub(/\s+/, "").split(",")
28   registered_users = User.where(:email => all_emails)
29   registered_users.each do |user|
30     if user
31       users << user
32       UserMailer.welcome_email(user, self).deliver
33     end
34   end
35   valid_emails = registered_users.collect(&:email)
36   invalid_emails = all_emails - valid_emails
37
38   errors.add(:base, :email_not_found, :emails => invalid_emails.join(", "), :count
=> invalid_emails.length) if invalid_emails.any?

```

```
39
40     [valid_emails, invalid_emails]
41 end
```

Ďalšia funkcionálna je prenesená do controllera.

### **app/controllers/collaborators\_controller.rb**

```
1 class CollaboratorsController < ApplicationController
2     include ApplicationHelper
3
4     before_filter :find_project, :find_collaborators
5
6     def create
7         authorize! :edit, @project
8
9         valid_emails, invalid_emails =
@project.add_collaborators_by_email(params[:collaborators])
10
11         flash[:notice] = t('collaborator.created', :emails => valid_emails.join(", "),
:project => @project.name, :count => valid_emails.length) if valid_emails.any?
12
13         if invalid_emails.any?
14             @active_tab = :collaborators
15             render 'projects/edit'
16         else
17             redirect_to dashboard_user_path
18         end
19     end
end
```

```
20
21 def edit
22   authorize! :edit, @project
23 end
24
25 def destroy
26   @user=User.find(params[:format])
27   if @user != @project.owner
28     if @user
29       @project.users.delete(@user)
30       flash[:notice] = t('collaborator.deleted', :name => @user.email)
31       if @user == current_user
32         redirect_to dashboard_user_path and return
33       end
34     else
35       flash[:alert] = t('collaborator.not_deleted', :name => @user.email)
36     end
37   else
38     flash[:alert] = t('collaborator.not_deleted_owner', :name => @user.email)
39   end
40   @active_tab = :collaborators
41   render 'projects/edit'
42 end
43
44 private
```

```

45
46 def find_collaborators
47   @collaborators = Project.find(params[:project_id]).users
48 end
49
50 def find_project
51   @project = Project.find(params[:project_id])
52 end
53 end

```

Chybové správy sú umiestnené v súbore **config/locales/en.yml**.

K celej tejto funkcionalite je vytvorený príslušný view (**app/views/collaborators/\_edit.html.erb**), ktorý zbiera parametre a posielajú ich do príslušného modelu a controllera.

### 3.3.2 Testovanie

Testovanie sme vykonávali pomocou Cucumber testov, ktoré sme postavili rozsiahle.

Kroky scenára	Akcia scenára
<b>BACKGROUND</b>	
Given the following users exists: "webtex1test@gmail.com", "webtex2test@gmail.com", "webtex3test@gmail.com", "webtex4test@gmail.com" Given I login as "webtex1test@gmail.com" And I have a project "iitsrc" And I want to edit collaborators on project "iitsrc"	Existujú vymenovaní používatelia. Som prihlásený ako používateľ <a href="mailto:webtex1test@gmail.com">webtex1test@gmail.com</a> a mám projekt „iitsrc“. A idem editovať kolaborantov k tomuto projektu.
<b>Scenario: Collaborator can see project on dashboard</b>	
And I go to collaborators tab And I add new collaborators "webtex2test@gmail.com,	Keď pridám nových kolaborantov, tak budú po

<p>webtex3test@gmail.com"</p> <p>When I login as "webtex2test@gmail.com"</p> <p>Then I should see that I have a project "iitsrc"</p> <p>When I login as "webtex3test@gmail.com"</p> <p>Then I should see that I have a project "iitsrc"</p>	<p>svojom prihlásení vidieť daný projekt na svojom dashboarde.</p>
<p><b>Scenario: User which is NOT between collaborators can not see project on dashboard</b></p>	
<p>And I go to collaborators tab</p> <p>And I add new collaborators "webtex2test@gmail.com, webtex3test@gmail.com"</p> <p>When I login as "webtex4test@gmail.com"</p> <p>Then I should not see the project "iitsrc"</p>	<p>Keď používateľ nie je v zozname kolaborantov projektu, nemôže vidieť projekt na svojom dashboarde.</p>
<p><b>Scenario: Collaborator can delete other collaborator</b></p>	
<p>And I go to collaborators tab</p> <p>And I add new collaborators "webtex2test@gmail.com, webtex3test@gmail.com"</p> <p>When I delete collaborator "webtex2test@gmail.com" of project "iitsrc"</p> <p>And I login as "webtex2test@gmail.com"</p> <p>Then I should not see the project "iitsrc"</p>	<p>Kolaborant môže z tabuľky kolaborantov zmazať iných kolaborantov.</p>
<p><b>Scenario: Collaborator can not delete owner</b></p>	
<p>And I go to collaborators tab</p> <p>And I add new collaborators "webtex2test@gmail.com, webtex3test@gmail.com"</p> <p>When I login as "webtex2test@gmail.com"</p> <p>And I want to edit collaborators on project "iitsrc"</p> <p>And I delete collaborator "webtex1test@gmail.com" of project "iitsrc"</p> <p>Then I should still see owner "webtex1test@gmail.com" in table of collaborator for project "iitsrc"</p> <p>When I login as "webtex1test@gmail.com"</p> <p>Then I should see that I have a project "iitsrc"</p>	<p>Kolaborant nemôže zmazať vlastníka projektu zo zoznamu kolaborantov.</p>
<p><b>Scenario: User which is NOT between collaborators can not see collaborators, when change url</b></p>	

And I add new collaborators "webtex2test@gmail.com, webtex3test@gmail.com" When I login as "webtex4test@gmail.com" And I want to edit collaborators on project "iitsrc" Then I should not be able to see collaborators	Tu je testovaná autorizácia, aby nemohol pristupovať k zmene kolaborantov nekolaborant.
<b>Scenario: If bad email is input, will be not added to table of collaborators</b>	
And I go to collaborators tab When I add new collaborators "blbost-webtech@gmil.com" Then I should not see collaborators "blbost-webtech@gmil.com" in table of collaborators for project "iitsrc"	Nie je možné pridať používateľov, ktorí nie sú registrovaný s platným emailom.
<b>Scenario: Sending emails with message, that user was added as collaborator for project</b>	
And I go to collaborators tab When I add new collaborators "webtex4test@gmail.com" Then "webtex4test@gmail.com" should receive an email	Kolaborantovi ktorí bol pridaní k spolupráci na projekte príde informačný email.

### 3.4 Dizajn a UX

Názov user story	Popis
<b>Dizajn a UX</b>	Ako používateľ chcem aby rozhranie bolo pre mňa zaujímavé vizuálne príťažlivé a zároveň prehľadné

Kvalitný návrh a grafický dizajn bývajú tým prvým prvkom, ktorým si aplikácia môže získať používateľa. Najskôr musí byť rozhranie teda príťažlivé a potom sa používateľ odhodlá vyskúšať funkcionality. Keďže ide o webovú aplikáciu je vhodné použiť technológie ako CSS3, compass-style, SCSS, HTML5, JAVAScript.

V rámci úlohy boli riešené nasledovné podúlohy:

- Návrh obrazoviek
- Štúdium CSS, SCSS, compass, HTML5
- Naštýľovanie zatiaľ implementovaných obrazoviek (login, dashboard, úprava projektov, editácia kolaborátorov, editácia súborov a revízie)

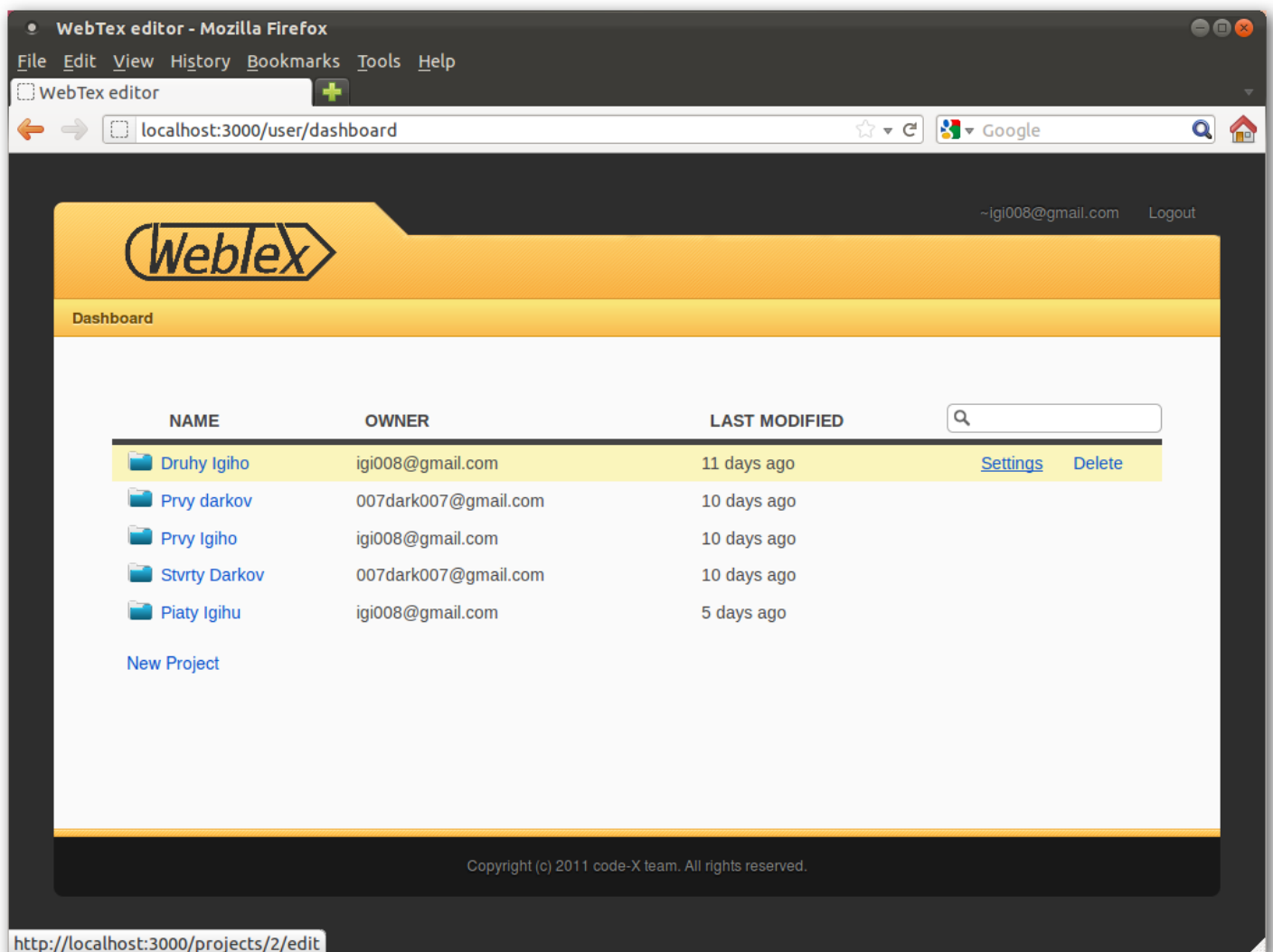
Na koľko ide o rozsiahlu a pomerne náročnú úlohu, jej realizácia prebieha naprieč viacerými šprintmi.

Pre realizácii rozhraní používame Compass ide o open-source framework pre CSS. Mnohé veci sú tu riešenie prehľadnejšie a pre programátora sa to používa lepšie lebo sa nemusí príliš opakovať. Pre dynamické prvky bol použitý najmä Javascript a jQuery.

Celý dizajn je zameraný na súlad farieb vizuálnu príťažlivosť, jednoduchosť a intuitívnosť. Logo bolo navrhnuté v programe Adobe Illustrator CS5 pomocou vektorovej grafiky, čo tiež značí o profesionálnom prístupe k dizajnu.

Každý jeden view musel byť naštýlovaný do finálnej podoby. Potrebný kód teda môžeme vidieť v jednotlivých viewoch.

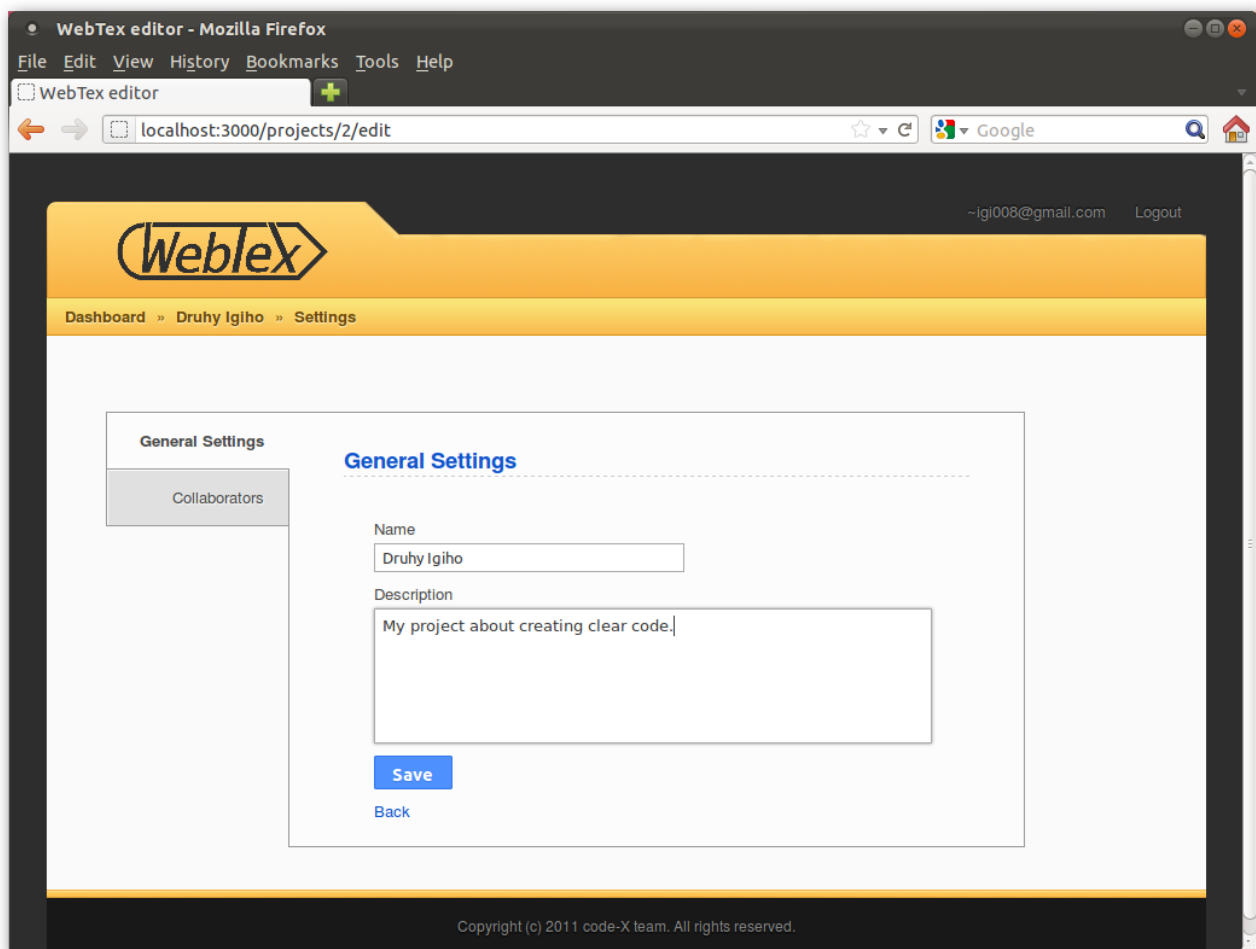
Tu ponúkame ukážku ako vyzerajú niektoré obrazovky nášho editora WebTex.



**Obrázok 2 – Obrazovka Dashboardu editora Webtex**

Na dashboarde používateľ môže vidieť svoje projekty a projekty na ktorých kolaboruje, to jednoducho zistí pomocou položky owner. Pri pohybe kurzora po dashborde sa zvýrazňuje konkrétny projekt a objavujú sa položky napravo nazvané Setting a Delete. Delete slúži na

jednoduché zmazanie projektu, pričom môžem zmazať iba svoj projekt, nie cudzí. Významná je však položka Settings, ktorá nám sprístupní nastavenie projektu.



**Obrázok 3 – Obrazovka nastavení projektu editora Webtex**

V nastaveniach projektu je nám umožnené zmeniť meno a opis projektu. Na karte Collaborator nám je umožnené plnohodnotne pridávať a odstraňovať spolupracovníkov projektu.

Na Dashboarde máme možnosť kliknúť na názov projektu, ktorý nás odnaviguje práve k súborom tohto projektu, kde ich môžeme jednoducho editovať. Môžeme meniť ich meno, ale aj obsah. Tu už pracuje aj verziovanie ktoré je založené n nástroji Git. Potom si môžem pozrieť históriu vývoja súboru pri kliknutí na Revisions, ktoré je dostupné pri každom súbore.



## 4 Štvrtý šprint - Dagmar

---

Tu sme si definovali úlohy hlavne charakteru štylovania a prezentácie tímu ako aj tvorbu dokumentácie. Nezabudli sme ani na dôležitú funkciu a začali sme pracovať na Editore a Kompilácii kódu.

Úloha	Plnenie
<b>Naštylovať kolaborantov</b>	Upraviť view kolaborantov do takej podoby, aby bol v súlade so štýlom projektu.
<b>Naštylovať strom súborov a zobrazit' ho</b>	Toto si vyžaduje implementáciu, aby sme mohli súbory zobrazovať v stromovej štruktúre, preto bude tomuto venovaná samostatná <b>kapitola (4.1)</b> .
<b>Naštylovať revízie súborov</b>	Upraviť view revízií do takej podoby, aby bol v súlade so štýlom projektu.
<b>Editor súborov</b>	Editor súborov je dôležitým elementom najmä kvoli prehľadnosti pri editovaní, číslovanie riadkov, zvýrazňovanie syntaxe a iné. Budeme transformovať riešenie CODEMIRROR pre naše účely. Vyžaduje si implementáciu, a preto sa tomu bližšie venujeme v samostatnej <b>kapitole (4.2)</b> .
<b>Kompilácia kódu</b>	Významný prvok aplikácie, v tomto šprinte sa venujeme iba čistému prekladu súboru. Vyžaduje si implementáciu, preto je bližšie popísaná v samostatnej <b>kapitole (4.3)</b> .
<b>Nová webová stránka tímu</b>	Nová webová stránka bola implementovaná za použitia CSS, HTML a JAVAscriptu. Stará webová stránka nebola najšťastnejšia a programátor, ktorý ju programoval náš tím opustil. Preto sme sa rozhodli vytvoriť novú, zaujímavejšiu a prehľadnejšiu.
<b>Preinštalovať server</b>	Server je nutné preinštalovať nastali určité problémy pri inštalácii Capistrano. Prechádzame na Fedoru a inštalujeme všetky potrebné veci, ktoré budú nutné na nasadenie našej aplikácie.
<b>Dokumentácia k projektu</b>	Ide o zdokumentovanie implementácie ktorá prebiehala

	v jednotlivých šprintoch. Niektoré šprinty boli dlhšie ako 2 týždne, ale bolo to najmä kôli odchodom členov tímu a rozsah funkcionality implementovaný v týchto šprintoch bol dosť široký.
<b>Prezentácia projektu</b>	Príprava materiálov na prezentáciu prototypu projektu.

## 4.1 Zobrazit' strom súborov a naštýlovať ho

Názov user story	Popis
<b>Zobrazit' strom súborov a naštýlovať ho</b>	Ako používateľ chcem aby som mal možnosť vidieť súbory v stromovej štruktúre

Realizácia nie je v čase odovzdávania dokumentácia dokončená. Úloha je v PROGRESE.

## 4.2 Editor súborov

Názov user story	Popis
<b>Editor súborov</b>	Ako používateľ chcem aby som mal možnosť editovať súbory v prehľadnom editore s číslovaním riadkov a zvýrazňovaním syntaxe

Realizácia nie je v čase odovzdávania dokumentácia dokončená. Úloha je v PROGRESE.

## 4.3 Kompilácia kódu

Názov user story	Popis
<b>Kompilácia kódu</b>	Ako používateľ chcem aby som mal možnosť súbory LaTeXu kompilovať do PDF.

Úlohou kompilácie je prekompilovať zadaný kód napísaný v Latexu do súboru formátu „pdf“ a uložiť tento súbor do adresára .../public/pdf/

Na kompiláciu kódu sa využíva nástroj Texlive.

Tu sú najdôležitejšie riadky kódu:

```
out = "#{Rails.root}/public/pdf/"
system("pdflatex -output-directory #{out} #{@document.file_path}")
redirect_to show_document_path(params[:project_id],
params[:project_name], params[:file_name])
```

V kóde je cesta k cieľovému adresáru zapísaná v premennej `out`. Následne sa využije príkaz Ruby systém, ktorému ako parameter vložíme príkaz Shellu `pdflatex`. Tento príkaz má prepínač `-output-directory`, to znamená, že musíme zadať aj priečinok kde sa pdf súbor uloží, to je v premennej `out`. Druhý parameter je cesta k súboru, ktorý obsahuje kód latexu. Tento súbor nemusí mať príponu `tex` alebo iné prípony špecifikujúce latexovské súbory.

## 5 Prototyp projektu WebTex

Náš prototyp je v podstate určitým naplnením Backlogu projektu. Je implementovaná značná časť funkcionality. Najvýznamnejšie jadro aplikácie je plne funkčné a dokonca je prototyp použiteľný.



#	Project	Tracker	Status	Priority	Subject	Assignee	Category	Target version	Start date	Due date	% Done
1036	WebTex editor	Feature	In progress	Normal	Neprihlásený používateľ		Backlog	Backlog	13.10.2011	11.12.2011	
1029	WebTex editor	Feature	New	Normal	► Zobrazenie „about“ stránky		Backlog	Backlog	13.10.2011		
1030	WebTex editor	Feature	Closed	Normal	► Prihlásenie používateľa		Backlog	Backlog	13.10.2011	09.11.2011	
1031	WebTex editor	Feature	Closed	Normal	► Registrácia používateľa		Backlog	Backlog	13.10.2011	09.11.2011	
1041	WebTex editor	Feature	Closed	Normal	► Možnosť prekladu LaTeX kódu		Backlog	Backlog	13.10.2011	11.12.2011	
1037	WebTex editor	Feature	In progress	Normal	Prihlásený používateľ		Backlog	Backlog	13.10.2011	30.11.2011	
1035	WebTex editor	Feature	Closed	Normal	► Možnosť vytvárania nových projektov		Backlog	Backlog	13.10.2011	14.11.2011	
1034	WebTex editor	Feature	Closed	Normal	► Odhlásenie používateľa		Backlog	Backlog	13.10.2011	09.11.2011	
1032	WebTex editor	Feature	Closed	Normal	► Zobrazenie zoznamu projektov		Backlog	Backlog	13.10.2011	14.11.2011	
1038	WebTex editor	Feature	Closed	Normal	► Možnosť zmazania projektu		Backlog	Backlog	13.10.2011	14.11.2011	
1039	WebTex editor	Feature	New	Normal	► Možnosť uploadu projektu		Backlog	Backlog	13.10.2011		
1040	WebTex editor	Feature	New	Normal	► Možnosť stiahnuť celý projekt		Backlog	Backlog	13.10.2011		
1042	WebTex editor	Feature	Closed	Normal	► Možnosť zmeny mena projektu		Backlog	Backlog	13.10.2011	14.11.2011	
1043	WebTex editor	Feature	Closed	Normal	► Nastavenie kolaborantov na projekt a príslušných práv		Backlog	Backlog	13.10.2011	30.11.2011	
1044	WebTex editor	Feature	New	Normal	► Možnosť nastavenie hlavného súboru na preklad		Backlog	Backlog	13.10.2011		
1045	WebTex editor	Feature	In progress	Normal	► Verziovanie projektu a pridávanie popisov k revíziám		Backlog	Backlog	13.10.2011		
1046	WebTex editor	Feature	Closed	Normal	► Zobrazenie jednotlivých revízií súborov v projekte		Backlog	Backlog	13.10.2011	30.11.2011	
1047	WebTex editor	Feature	Closed	Normal	► Vytváranie nového prázdneho súboru		Backlog	Backlog	13.10.2011	30.11.2011	
1048	WebTex editor	Feature	New	Normal	► Vytváranie nového súboru z templateov		Backlog	Backlog	13.10.2011		
1049	WebTex editor	Feature	New	Normal	► Vytváranie nového súboru z templateov		Backlog	Backlog	13.10.2011		
1050	WebTex editor	Feature	New	Normal	► Uloženie súboru medzi templaty		Backlog	Backlog	13.10.2011		
1051	WebTex editor	Feature	Closed	Normal	► Možnosť zmazania existujúceho súboru		Backlog	Backlog	13.10.2011	30.11.2011	
1052	WebTex editor	Feature	New	Normal	► Možnosť uploadu súboru		Backlog	Backlog	13.10.2011		
1054	WebTex editor	Feature	New	Normal	GIT - synchronizácia mimo prostredia webu		Backlog	Backlog	13.10.2011		
1055	WebTex editor	Feature	New	Normal	► GIT PUSH		Backlog	Backlog	13.10.2011		
1056	WebTex editor	Feature	New	Normal	► GIT PULL		Backlog	Backlog	13.10.2011		
1057	WebTex editor	Feature	New	Normal	► GIT autentifikácia menom a heslom alebo pomocou ssh kľúča		Backlog	Backlog	13.10.2011		
1058	WebTex editor	Feature	New	Normal	LaTeX editor		Backlog	Backlog	13.10.2011		
1059	WebTex editor	Feature	In progress	Normal	► Syntax highlighting		Backlog	Backlog	13.10.2011		
1060	WebTex editor	Feature	New	Normal	► LaTeX autocompletion		Backlog	Backlog	13.10.2011		
1053	WebTex editor	Feature	New	Normal	► Možnosť pripájania poznámok k LaTeX kódu pomocou breakpointov		Backlog	Backlog	13.10.2011		
1061	WebTex editor	Feature	New	Normal	► Možnosť pdf náhľadu		Backlog	Backlog	13.10.2011		
1062	WebTex editor	Feature	New	Normal	► Highlighting aktuálneho riadku		Backlog	Backlog	13.10.2011		
1063	WebTex editor	Feature	New	Normal	► Search and replace stringov		Backlog	Backlog	13.10.2011		
1064	WebTex editor	Feature	New	Normal	► Full-screen editing		Backlog	Backlog	13.10.2011		
1065	WebTex editor	Feature	New	Normal	► Podpora zmeny temy		Backlog	Backlog	13.10.2011		
1066	WebTex editor	Feature	New	Normal	► Undo a Redo		Backlog	Backlog	13.10.2011		
1067	WebTex editor	Feature	In progress	Normal	► Číslovanie riadkov		Backlog	Backlog	13.10.2011		
1068	WebTex editor	Feature	New	Normal	► Informačný status bar		Backlog	Backlog	13.10.2011		
1121	WebTex editor	Feature	New	Normal	► Tlačítka na doplnenie kódu		Backlog	Backlog	14.10.2011		

Obrázok 3 – Plnenie Backlogu v čase písania dokumentácie

### 5.1 Čo umožňuje náš prototyp

Dá sa povedať že tie najdôležitejšie prvky finálnej aplikácie by mali bez väčších problémov fungovať už v tomto prototypu. Silné jadro aplikácie je teda pripravené. Ďalšia funkcionality, ktorá zostala neimplementovaná môže byť vnímaná ako akési vylepšenia aplikácie. Teraz zhrnieme našu funkcionality do niekoľkých podkapitol. Každá jedna kapitola predstavuje množstvo práce, ktoré by sa dalo popisovať do väčších detailov. Ale pre prehľadnosť budeme výstižní.

### **5.1.1 Registrácia a autentifikácia používateľov**

Pri práci s našou aplikáciou je zatiaľ potrebná registrácia, neskôr bude určitá funkcionálna sprístupnená aj neregistrovaným používateľom. Každý používateľ sa zaregistruje pod svojím emailom a heslom. Pod týmito atribútmi bude v systéme vystupovať. Máme vytvorenú tabuľku používateľov, kde uchováваме aj iné dôležité atribúty ako čas posledného prihlásenia, IP adresa posledného prístupu, token na reset hesla iné. Tieto atribúty nám umožňujú rozširovať ďalej funkcionálnu. Čiže používateľ sa môže plnohodnotne zaregistrovať a potom sa môže samozrejme aj so svojimi údajmi prihlasovať do aplikácie.

### **5.1.2 Manažment projektov**

Aplikácia umožňuje plnohodnotne vytvárať projekty s ich názvom a opisom. Projekt je vlastne základná jednotka, od ktorej sa všetko dovíja. Pri vytvorení projektu je v súborovom systéme vytvorená zložka s ID používateľa, ktorý projekt vytvoril. V tejto zložke používateľa je podzložka s ID projektu, ktorý vytvoril. Táto zložka projektu je prípravou lokality pre strom súborov, ktoré budú v projekte.

Projektom je možné jednoducho meniť meno, alebo ich opis. Projekty je možné jednoducho zmazať. A kedykoľvek môže používateľ jednoducho vytvoriť nový projekt.

### **5.1.3 Manažment súborov**

Používateľovi sme umožnili vytvárať súbory v nejakom projekte. Tieto súbory môže nielen vytvárať, ale aj meniť ich meno, mazávať ich, a čo je asi najdôležitejšie, môže editovať ich obsah. Tieto súbory sú potom zapísané v príslušnom projekte na súborovom systéme. Pamätali sme samozrejme na rôzne ošetrenia problémov, ktoré by mohli nastať. Ide napríklad o súbor bez názvu alebo o súbory s rovnakým názvom atď. Aplikácia pri takýchto konfliktoch používateľa upozorní.

### **5.1.4 Revízie súborov**

Vynikajúcim prvkom našej aplikácie je možnosť sledovať zmeny súborov v čase. Na toto jednoduché verziovanie sme využili nástroj Git, ktorý je zakomponovaný v našej aplikácii. Pri každom súbore si teda jednoducho používateľ môže pozrieť jeho predchádzajúce zmeny v čase. Riešenie pomocou verziovacieho nástroja Git je veľmi sofistikované, pretože sú uložené iba zmeny, ktoré nastali, nemusia sa tak ukladať celé súbory v rozličných verziách. Takto sa jednoducho

ukladajú len zmeny a aj tak máme prístup k celej verzii súboru. Táto funkcionálnosť je tiež nejakou pridanou hodnotou oproti niektorým existujúcim riešeniam.

### **5.1.5 Kolaborácia používateľov**

Veľmi dôležitým prvkom našej aplikácie, ktorým sa asi najviac odlišujeme od podobných riešení, je kolaborácia používateľov na projektoch. Ide o mechanizmus, kedy používatelia môžu spolupracovať. Používateľ môže jednoducho na svojej pracovnej ploche v aplikácii takzvanom Dashboarde vidieť aj projekty, ku ktorým bol pridaný ako spolupracovník. Tieto projekty môže plnohodnotne upravovať a editovať.

Pri kolaborácii je dôležité to, že k svojmu projektu alebo projektu na ktorom spolupracuje môže používateľ jednoducho pridať ďalších spolupracovníkov. Pridávanie spolupracovníkov, prebieha pomocou výpisu emailových adries za sebou oddelených čiarkou. Takýmto jednoduchým spôsobom môžeme pridať neobmedzený počet spolupracovníkov iba jednoduchým zoznamom emailových adries. Aplikácia je sofistikovaná a kontroluje, či sú používatelia registrovaní, a teda či nie je zadaná chybná emailová adresa. Aplikácia na rôzne chyby upozorní prehľadnou správou. Pri úspechu je tiež užívateľ upovedomený.

Aplikácia je tak šikovná, že jednotlivým používateľom, ktorí boli pridaní na kolaboráciu pošle informačný email s názvom projektu, v ktorom boli pridaní medzi spolupracovníkov.

Spolupracovníci sú prehľadne zobrazení v tabuľke ku každému projektu. Ktoréhokoľvek spolupracovníka okrem vlastníka projektu možno kedykoľvek jednoducho zmazať.

### **5.1.6 Autorizácia**

Vytvárame našu aplikáciu tak, aby bola použiteľná pre veľké množstvo ľudí. Pri takýchto projektoch je dôležité dbať na bezpečnosť. Musíme preto zabezpečovať prístup k jednotlivým projektom. Prístup k projektu a operácie nad ním by mali byť umožnené iba vlastníkovi projektu alebo spolupracovníkom. A presne takto je to v našej aplikácii zrealizované. Zabezpečujeme preto dôležité informácie pred očami nepovolaných používateľov. Ak by sa iný používateľ pokúšal pristupovať k projektom, na ktoré nemá oprávnenie, napríklad by mohol skúšať zmenu URL adresy, naša aplikácia mu to zamietne a neumožní mu to, presmeruje ho na chybovú stránku.

### **5.1.7 Editor projektu**

Základom dobrej aplikácie kde treba meniť obsah súboru je kvalitný editor. Takýto editor je o to dôležitejší, keď ide o úpravu kódu nejakého druhu počítačového jazyka. Takýmto kódom nepochybne LaTeX je. Preto sme chceli aby editor čo najviac uľahčoval prácu používateľovi pri písaní kódu. V tomto prototypy sme preto zaviedli do editora číslovanie riadkov a zvýrazňovanie syntaxe, ktoré uľahčujú prácu. Ide o celkom rozsiahlu funkcionality editora, ktorá je používateľovi umožnená. Prispôbujeme si totiž editor otvoreného projektu CODEMIROR. Editor je implementovaný pomocou Javascriptu, je dobre použiteľný, a dá sa rozširovať. Preto by mal byť kvalitným základom pre náš projekt do budúcnosti, keď budeme zavádzať mnohé vylepšenia.

### **5.1.8 Stromová štruktúra**

Asi netreba príliš vysvetľovať čo to znamená. Súbor by mali byť zobrazené prehľadne, tak aby bolo jasné, ktoré sú nadradené zložky. Práve toto je vhodné realizovať vizualizáciou pomocou nejakého stromu. Túto vizualizáciu rieši náš prototyp. Neskôr by sme chceli implementovať aj úpravu časti stromu.

### **5.1.9 Preklad LaTeX súborov**

Tento preklad je asi najväčšou podstatou aplikácie, keďže ide o online editor LaTeXu. Náš prototyp aplikácie zatiaľ umožňuje kompiláciu ľubovoľného súboru v projekte do formátu PDF. Po kompilácii sa výsledný súbor uloží a používateľ si ho môže jednoducho stihnúť.

### **5.1.10 Grafický dizajn a rozhranie**

Ide o dôležitý element, ktorý v prvom rade osloví používateľa. Vymysleli sme prehľadný profesionálny a intuitívny dizajn k celej doteraz implementovanej funkcionalite. Dokonca sme pracovali aj na rôznych verziách loga a vytvárali sme ho pomocou vektorovej grafiky, ako pri profesionálnych produktoch vysokého štandardu. Celkové rozhranie aplikácie pôsobí veľmi príjemne, farebné ladenie nie je rušivé a zachováva si veľkú prehľadnosť. Zvolili sme tmavšie farby ktoré sú šetrnejšie k očiam v kombinácii s jemne výraznými farbami, ktoré upozorňujú na dôležité elementy a upozornenia. Dizajn je realizovaný pomocou moderných technológií ako SCSS,

compass, jQuery, Javascript. Samotné technológie hovoria za pokrokovosť aplikácie o ktorú usilujeme.

## **5.2 Zhodnotenie a vízia do budúca**

### **5.2.1 Zhodnotenie prototypu**

Usudzujeme, že prototyp našej aplikácie zvláda naozaj veľa. Priniesli sme mnoho nápadov a nimi môžeme pri kvalitnej implementácii náš produkt postaviť medzi svetové jednotky na trhu. V súčasnom stave už preskakujeme začnú časť konkurencie. Základ našej aplikácie je položený. Naša aplikácia zvláda to čo by používateľ očakával. Podstata toho čo sme chceli zrealizovať je implementovaná. Hrubé múry sú postavené, a mali by byť pevným základom pre našu aplikáciu do budúca.

### **5.2.2 Vízia do budúca**

Je dobré, že tá najdôležitejšia kostra je pripravená už v tomto prototypu. Iste, vždy je čo vylepšovať, a preto by sme do budúca chceli vychytať aj najmenšie detaily. Ďalším ťažiskom našej práce by malo byť riešenie problémov ako preklad Latexu v reálnom čase za účelom okamžitého náhľadu. Významnou by mala byť možnosť pridávať poznámky k jednotlivým častiam preloženého dokumentu. Ďalším cieľom bude vylepšiť editor tak, aby poskytoval používateľovi všetko, čo by mohol potrebovať, a čo by uľahčovalo jeho prácu, napríklad automatické dopĺňanie príkazov LaTeXu. Ak by sa nám tieto prvky podarilo úspešne zrealizovať, trufame si povedať, že by naša aplikácia jednoznačne preskočila exitujúcu konkurenciu a stala by sa lídrom na trhu.