

Technical Documentation

Fachbereich Informatik

Telekooperation

Prof. Dr. Max Mühlhäuser

***Telekooperation Internet Praktikum – Technische Dokumentation***

Viet Duc Dinh, Johannes Klumpe, Thomas Maciejewski

Betreuer: Christian Meurisch

Darmstadt, 08.09.2016

Inhalt

[1. Introduction 3](#_Toc461055137)

[1.1 Stack 3](#_Toc461055138)

[1.2 Development Process 3](#_Toc461055139)

[2. General technical setting 4](#_Toc461055140)

[2.1 Entity Relationship Model 4](#_Toc461055141)

[2.2 Client/app 5](#_Toc461055142)

[2.2.1 Account 5](#_Toc461055143)

[2.2.2 Admin 5](#_Toc461055144)

[2.2.3 Charts 5](#_Toc461055145)

[2.2.4 Main 6](#_Toc461055146)

[2.2.5 Reviews 6](#_Toc461055147)

[2.2.6 Submissions 6](#_Toc461055148)

[3. User Guide - Conference-Platform 7](#_Toc461055149)

[3.1 General functionalities – Get started 7](#_Toc461055150)

[3.1.1 Registration 7](#_Toc461055151)

[3.1.2 Log-in 8](#_Toc461055152)

[3.2 Author 8](#_Toc461055153)

[3.2.1 Create submission 8](#_Toc461055154)

[3.2.2 Submissions - options 10](#_Toc461055155)

[3.3 Reviewer 10](#_Toc461055156)

[3.3.1 Access assigned submissions 11](#_Toc461055157)

[3.3.2 Make a review 11](#_Toc461055158)

[3.4 Chair 12](#_Toc461055159)

[3.4.1 Access all submissions 12](#_Toc461055160)

[3.4.2 Information retrieval: Authors, submissions, reviewers, reviews 12](#_Toc461055161)

[3.4.3 Submission to reviewer assignment 13](#_Toc461055162)

[3.4.4 Analytics 14](#_Toc461055163)

[3.5 Implemented Features 14](#_Toc461055164)

# Introduction

The „Team Uniform“-project is an implementation of a conference platform. In order to plan the contributions of a conference,”Team Uniform” provides several functionalities to submit, review and assess research papers.

It is based on yeoman, a scaffolding tool that utilizes modern technical web-technologies like angularJS or nodeJS. Requirements and functionalities were implemented according to the specifications of the TK3 department from the Technical University Darmstadt.

## Stack

Our team used the yeoman generator to initially set up an environment and having a starting point for development. At the first step the team worked on the backend using Node.js as primary technology. As database we used the latest version of postgresql, an open source object-relational database system that is very popular and works well among different operating systems (quelle: postgresql.org).

The frontend is served by Angular.js which allows dynamic views on our platform.

Finally, we used bootstrap as CSS framework.

## Development Process

In an initial meeting the team decided on following five process steps for the development and implementation of “Team Uniform”

1. Setting up the technical stack with Yeoman, developing a relational data-scheme and determination of API-specifications.
2. Setting up a rights-management-logic based on binary variables and creating CRUD-functionalities for review- and submission-objects,
3. Creating views to enable CRUD-functionalities on frontend for reviews and submissions
4. Developing features on frontend- and backend-side for role chair.
5. Finalization of the project and realization of several bonus-features.



# General technical setting

The general folder-structure of “Team Uniform” looks like this:

├─── Team Uniform

│ └─── client

│ └─── e2e

│ └─── server

│ └─── node\_modules

│ └─ app.sh

│ └─ package.json

│ └─ README.md

│ └─ ...

## Entity Relationship Model

## Client/app

Located in the client-folder is the app-directory, which contains frontend functionalities mainly utilizing Angular.js.

├─── client/

│ └─ app/

│ └─── account/

│ └─── admin/

│ └─── charts/

│ └─── main/

│ └─── reviews/

│ └─── submissions/

We have ordered and named the subdirectories after their respective functionality necessary to provide a functional platform in order to manage conference papers and their reviews.

### Account

├─── account/

│ └─ login/

│ └─ settings/

│ └─ signup/

The directory account controls the process of registration and authentication. There is also an implementation to allow users to change their password. We created three controllers and views for these processes. The login-process consists of the controller login.controller.js and the view login.html. Signing up is realized by signup.controller.js and settings.html. Furthermore /settings provides functions to allow users to change their settings, e.g. change the password. This is achieved with settings.controller.js and settings.html. Finally, account.js provides the routing logic.

### Admin

├─── admin/

│ └─ admin.controller.js

│ └─ admin.html

│ └─ admin.scss

│ └─ admin.router.js

│ └─ admin.module.js

In the admin directory we have implemented the functionalities of the chair-role. He is allowed to see all registered users on the platform. Also deletion of users in that list is possible.

Functionalities are provided by admin.controller.js. Please be aware that some parts of this view are changed by admin.scss, e.g. the trash-icon or the font-style.

### Charts

├─── charts/

│ └─ charts.controller.js

│ └─ charts.index.html

│ └─ charts.module.js

│ └─ charts.router.js

We used zingchart, a javascript library, to provide the possibility of displaying general statistics of the platform. The controller is utilized via charts.controller.js whereas the view is displayed by charts.index.html.

### Main

├─── main/

│ └─ main.controller.js

│ └─ main.html

│ └─ main.js

│ └─ main.scss

The main directory contains the controller and view when opening the index-page. Controllers are utilized by main.controller.js, the view is managed by main.html. Also some parts of the design are modified by main.scss.

### Reviews

├─── reviews/

│ └─ reviews.controller.js

│ └─ reviews.index.html

│ └─ reviews.form.html

│ └─ reviews.show.html

│ └─ reviews.router.js

│ └─ reviews.module.js

Reviews are important objects of the conference-platform and directly linked to submissions. Functionalities included are creating, deleting and updating a review as well as displaying submissions that were assigned to users to be reviewed by them.

### Submissions

├─── submissions/

│ └─ submissions.controller.js

│ └─ submissions.index.html

│ └─ submissions.create.html

│ └─ submissions.show.html

│ └─ submissions.update.html

│ └─ submissions.file.html

│ └─ submissions.assign.html

│ └─ submissions.router.js

│ └─ submissions.module.js

Submissions are besides reviews the most important objects on the conference-platform. Users create a submission by giving it a title, adding keywords and providing an abstract. Then the user is asked to upload it in an appropriate file-format such as pdf. After completing this steps a new submission is created. Until a submission is reviewed the user is able to change the title, keywords and abstract of his submission. He is also allowed to change the file he has uploaded.



# User Guide - Conference-Platform

## General functionalities – Get started

In order to achieve a fully functional platform to submit, read, update and review journal or paper submissions it is necessary to create an user on the platform.

To initially set up the Uniform-environment follow these steps:

1. Prerequisites

|  |  |
| --- | --- |
| Git |  |
| Node.js & npm | Node: 4.2.3, npm: 2.14.7 |
| Bower |  |
| Ruby | gem install sass |
| Grunt | npm install –global grunt-cli |
| Postgres |  |

1. Run ‘npm install’ to install server dependencies.
2. Run ‘bower install’ to install frontend dependencies
3. Copy ‘app.sh.sample’ and name it ‘app.sh’
4. Modify the ‘SEQUELIZE\_URI’ in ‘app.sh’ to match your local Postgres database configuration
5. Allow ‘app.sh’ to be executed by executing ‘chmod +x app.sh’
6. Run ‘./app.sh serve’ to start the development server, it should automatically open the client in your browser when ready.
7. Testing: Running `./app.sh test` will run the unit tests with karma.
8. Deployment: Run `grunt build` and use the `grunt buildcontrol:heroku` command to deploy the assembled dist folder to the production system. You can adjust the production system repository in the `Gruntfile.js`.

### Registration

To register an account on the conference-platform, click the ‘Sign up’ sign in the upper right corner.

The user can then provide his credentials and choose a safe password for his user. If the user already has an account he can click on the “Login”-button and then sign in with his email-address and password.

### Log-in

To login select the Login option in the upper right corner and sign in.

## Author

When an user registers on the platform he automatically becomes an author where he is able to create submissions.

### Create submission

To create a new submission, click on the button “+ New Submission” in the middle-left part of the view.

In the second step provide further information about the paper you want to submit.

At the last step the user can upload a file that contains the paper proposal for his submission.



In order to upload a file for submission the user has to click the button “Datei auswählen”.

He is then directed to his computer folder directory where he can select one or several files for uploading.

### Submissions - options

As the user is able to create multiple submissions the platform provides a view where he can access all his submissions. To open this view click on “Submissions” in the top-navigation.



* + When clicking this button you can upload further files.
  + By clicking this button, the title, keywords and abstract of a submission can be changed.
  + To delete your submission, click this button, you will then be asked if you are sure to delete is. By affirming it the submission will be deleted.

## Reviewer

Concerning the quality of papers so called ‘reviewers’ are able to check and evaluate the quality of a submission by providing different information about it. To become a reviewer an user has to ask for permission to that role by selecting “Reviews” in the homescreen and then click the button “Become a Reviewer”.

### Access assigned submissions

After applying for the reviewer-position the chair user can grant access for reviewing specific submissions as mentioned in 3.4.3.

The user then has in the view “Reviews” his assigned and finished reviews listed as shown in this screenshot:

### Make a review

To create a review the reviewer the has to click the “+”-button for one of his assigned submissions to come to a form for rating the submission:

## Chair

The chair is similar to an admin on this platform. He mainly administrates authors and reviewers.

### Access all submissions

On the view after login the chair is able to see all submissions. He can also check the status of each submission and their creation date



### Information retrieval: Authors, submissions, reviewers, reviews

To get information about all users displayed, click in the top-navigation on the tab “Users”.

The application will then display all registered users with further information. The chair has also the option to delete specific users.

To get information about submissions click the tab “Submissions” in the top-navigation. You will then see following:

### Submission to reviewer assignment



To assign a submission to a reviewer the chair has to click this button.

You will then be directed to a view where it is possible to add users to the list of reviewers for a specific submission:

By clicking the “+”-button the chair can assign user as reviewers for the chosen submission.

### Analytics

As an administrator it is useful to have access to basic analytic views instead of gathering information in a lengthy manual process.

Following you can see a screenshot displaying useful information about submissions and reviews:



## Implemented Features

1. **User:** Register to system with name, email and password. Login with email and password.
2. Unique emails.
3. Set new password in profile.
4. **Author:** Create a submission and access current submission as overview.
5. Updating/editing submissions
6. Withdraw a submission
7. **Reviewer:** Access assigned submissions
8. Make a review and updating/editing it.
9. **Chair:** List all submissions/reviews/users.
10. Assign submissions to reviewers.
11. Schedule management: Closing/opening submissions and reviews.
12. Reporting overview/analytics.