

Studienarbeit

im Fach Praktikum Softwareentwicklung

*Task Management-System –  
mit integriertem Raumbuchungssystem*



**Hochschule  
Hof**

University of  
Applied Sciences

Verfasser: *Eugen Kudraschow*  
*00275520*  
*Gruppe: "Bottlenecks"*  
*Sommersemester 2022*  
*Hochschule Hof*

Dozent: *Stefan Müller*

Abgabetermin: *08.07.2022*

## Inhalt

|   |          |
|---|----------|
| <b>1. Vorwort</b>                                       | <b>4</b> |
| <b>2. Vorbereitungen</b>                                | <b>4</b> |
| 2.1 Gruppenbildung                                      | 4        |
| 2.2 Themenauswahl                                       | 4        |
| 2.3 Namensfindung und Styleguide                        | 4        |
| 2.4 Auswahl der Technologien und Frameworks             | 5        |
| 2.5 Aufgabenaufteilung der jeweiligen Gruppenmitglieder | 5        |
| <b>3. Organisatorisches</b>                             | <b>6</b> |
| 3.1 Protokollierung                                     | 6        |
| 3.2 Gruppen Zusammenführung                             | 6        |
| 3.3 Einweisung des neuen Mitglieds                      | 6        |
| <b>4. Design</b>  | <b>7</b> |
| 4.1 Gestaltung der Dokumente                            | 7        |
| 4.2 Gestaltung der Wireframes                           | 7        |
| <b>5. Frontend Struktur</b>                             | <b>8</b> |
| 5.1 Allgemein   | 8        |
| 5.2 User-Dashboard                                      | 8        |
| 5.3 Projekt-Dashboard                                   | 9        |
| 5.4 Weitere Elemente                                    | 9        |
| <b>6. Frontend Implementierung</b>                      | <b>9</b> |
| 6.1 Allgemein   | 9        |
| 6.2 Suchleiste  | 10       |
| 6.3 Fortschrittbalken                                   | 10       |
| 6.4 Navigationsleiste – User-Dashboard                  | 10       |
| 6.5 Navigationsleiste – Projekt-Dashboard               | 10       |
| 6.6 User-Dashboard – Projekt                            | 11       |
| 6.7 Projekt-Dashboard – Admin                           | 11       |
| 6.8 Projekt-Dashboard – Tasks                           | 12       |
| 6.9 Projekt-Dashboard – Deadline                        | 12       |
| 6.10 Projekt-Dashboard – Übersicht                      | 12       |

|          |   |           |
|----------|---|-----------|
| 6.11     | Projekt-Dashboard – Ankündigungen ..... | 12        |
| 6.12     | Projekt-Dashboard – Tags .....          | 13        |
| 6.13     | Registrierung .....                     | 13        |
| <b>7</b> | <b>Testen &amp; Errorhandling.....</b>  | <b>13</b> |
| 7.6      | Errorhandling.....                      | 13        |
| 7.7      | Testen.....                             | 13        |

## 1. Vorwort

Für den Kurs Praktikum Softwareentwicklung bestand die Aufgabe ein Software-Projekt im Rahmen von festgelegten Richtlinien innerhalb des Sommersemesters 2022 zu entwickeln. Dabei soll in einem Team eine arbeitsnahe Umgebung geschaffen werden. Darunter fällt die Zusammenarbeit mit „Arbeitskollegen“ sowie die Aufgabenaufteilung und auch organisatorische Aspekte. Dabei soll auf zukünftige Geschehnisse vorbereitet werden.

## 2. Vorbereitungen

### 2.1 Gruppenbildung

Zu Beginn bestand unsere Gruppe aus 3 Mitgliedern Johannes Matus, Daniel Vogel und Eugen Kudraschow. Dabei haben wir uns auf „Bottlenecks“ für den Gruppennamen geeinigt.

Zum späteren Verlauf musste ein Mitglied der Gruppe „Bottlejobs“ krankheitsbedingt ausfallen, weshalb diese dann nur noch aus 2 Mitgliedern bestanden. Da die Gruppengröße bis max. 6 Leute zugelassen war, hat unsere Gruppe 2 neue Mitglieder erhalten, die da wären, Sebastian Bär und Dejan Fraas.

### 2.2 Themenauswahl

Nach der ersten Veranstaltung und Gruppenbildung, haben wir uns zum Brainstorming getroffen. Um geeignete Themen für unsere Studienarbeit zu bekommen. Darunter sind Themen gefallen wie ein E-Mail-Provider oder Video-Lernplattform für Studenten.

Am Ende fiel dann die Entscheidung, auf Task-Management-System, welche Features bietet, wie Projekte anlegen und Tasks innerhalb der Projekte erstellen.

### 2.3 Namensfindung und Styleguide

Um uns identifizieren zu können haben wir uns als Gruppe die „**Bottlenecks**“ genannt. Auch ist ein gleichmäßiges Erscheinungsbild erforderlich, deshalb wurde ein Styleguide erstellt. Zusätzlich benötigt unsere Web-Applikation ebenfalls diese Eigenschaften und so sind wir nach mehreren Fehlschlägen auf den Namen „**Stiva**“ gekommen, was keine weitere Bedeutung hat. Und haben dementsprechend auch ein Styleguide entwickelt.

## 2.4 Auswahl der Technologien und Frameworks

Da parallel zum Software-Praktikum eine weitere Studienarbeit erstellt werden sollte, fiel die Entscheidung einfach, uns auf eine weitere Web-Applikation zu einigen. So kann gelerntes aus beiden Kursen effektiver eingesetzt werden und man erhält mehr Erfahrung im Bereich der Web-Entwicklung, welches ein großes Themengebiet in der Informatik ist.

Bei der Auswahl der Frameworks entschieden wir uns für Laravel, welches im Hintergrund (Backend) läuft und die Datenbank sowie Struktur des Datenmodelles enthält.

Für die grafische Darstellung und Benutzereingaben haben wir uns im Frontend für das Framework React entschieden. Um uns die grafische Darstellung zu vereinfachen haben wir zusätzlich auf Tailwind gesetzt. Damit lassen sich verschiedene Eigenschaften wie Margin, Padding usw. leicht für die jeweilige Komponente anpassen. Als Schnittstelle wurde eine Rest-API von seitens Laravel entwickelt.

## 2.5 Aufgabenaufteilung der jeweiligen Gruppenmitglieder

Durch unsere Gruppengröße von anfangs 3 Leuten war es einfach die Aufgaben innerhalb des Projektes aufzuteilen.

Meine Aufgabe bestand daraus mich mit dem Framework React auseinander zu setzen. Somit war ich für das Frontend zuständig. Dies beinhaltet die grafische Darstellung der Web-Applikation und auch sämtliche Benutzereingaben zu managen. Außerdem ist ein weiterer Aufgabenbereich die dynamische Darstellung von Daten aus der Datenbank.

Johannes Matus hat sich mit dem Framework Laravel beschäftigt und war somit für das Datenmodell und die Datenbank zuständig. Diese werden bei jeder Web-Applikation benötigt.

Daniel Vogel agierte als Bindeglied zwischen uns und hat beide Aufgaben teilweise übernommen.

Wie unter Punkt 2.1 bereits beschrieben, wurden uns im späteren Verlauf 2 weitere Mitglieder zugewiesen. Dabei hat Dejan Fraas mich im Frontend und Sebastian Bär, Johannes Matus im Backend unterstützt.

## 3. Organisatorisches

### 3.1 Protokollierung

Eine organisatorische Aufgabe von mir bestand darin, die Protokollierung für unsere Treffen mit Herrn Müller durchzuführen. Darunter fällt auch das Layout für die Protokolle. Wir hatten uns geschlossen geeinigt, dass ich für alle Treffen die Protokolle anfertige. Dazu habe ich während des Treffens mir kurze Stichpunkte notiert und diese dann im Nachhinein aufgearbeitet und auf das Layout gebracht.

Es wurde protokolliert, wer an dem Treffen teilgenommen hat, die Dauer der jeweiligen Tagesordnungspunkte, sowie die gesamte Besprechungszeit. Zusätzlich wurden die Ergebnisse von vorherigen Treffen mit Herrn Müller verglichen. Bei auftretenden Problemen wurde darüber ausführlich diskutiert, bis schlussendlich eine Lösung dafür gefunden wurde. Anschließend wurde der weitere Verlauf für das Projekt geklärt und auch das nächste Treffen vereinbart. Diese Protokolle habe ich dann auf unsere Plattform GitHub hinzugefügt. Damit sie für alle zugänglich sind.

### 3.2 Gruppen Zusammenführung

Wegen einem krankheitsbedingten Ausfall bei der Gruppe „Bottlejobs“, welche dann nur noch aus 2 Mitgliedern bestand und somit eine Realisierung des Softwareprojektes nicht mehr gegeben war. Entschieden wir uns die Gruppen zusammenzulegen und deren Fortschritt bis zu der Zeit so gut wie möglich in unser Projekt zu integrieren. Das führte dazu, dass die neuen Teilnehmer von uns in unser Projekt eingewiesen werden mussten. Dazu haben wir uns zusammengetroffen und auch die Integration in unser System angepasst, so dass es einen sinnvollen Einsatz hat.

### 3.3 Einweisung des neuen Mitglieds

Nachdem die neuen Mitglieder eingewiesen waren, mussten Ihnen auch neue Rollen zugeteilt werden. Dabei hat sich Herr Dejan Fraas dem Frontend zugeordnet.

Damit ein zusammenarbeiten ermöglicht wird, habe ich ihm den aktuellen Stand der Web-Applikation „Stiva“ gezeigt. Dazu wurde die Struktur genau erläutert und auch auf die Funktionalität der bereits vorhandenen Komponenten im Frontend-

Bereich erklärt. Auch wurde besprochen, wie wir die Aufgabenaufteilung nun zukünftig regeln. Da die Applikation in 2 Bereiche aufgeteilt war, einmal in die Projektansicht und einmal in die Useransicht, teilten wir uns auch dementsprechend auf. Zusätzlich unterstützten wir uns bei Fragen und Problemen.

## 4. Design

### 4.1 Gestaltung der Dokumente

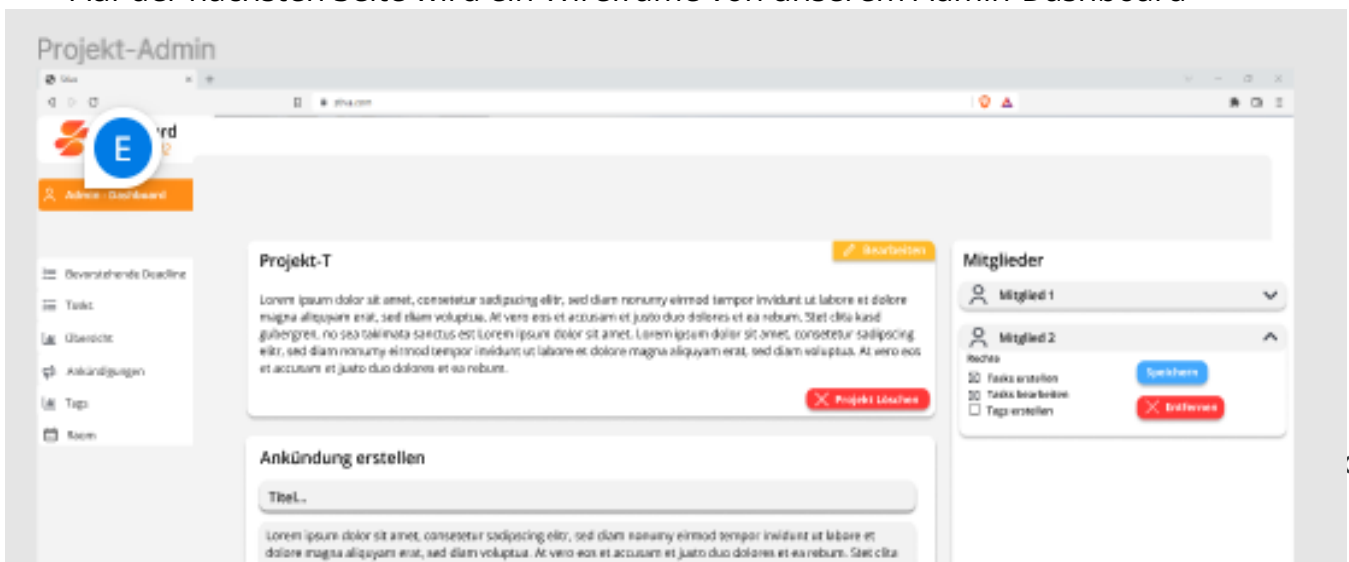
Wie unter Punkt 2.3 bereits erwähnt, wird ein gleichmäßiges Erscheinungsbild benötigt. Dabei habe ich verschiedene Layouts für unsere Dokumente erstellt. Darunter fallen die Protokolle oder auch allgemeine Unterlagen. Orientiert habe ich mich an den von uns entwickelten Styleguide. Mir war es wichtig die Dokumente schlicht zu halten, um nicht zu sehr abzulenken. Als Farbakzente setzte ich die Farbe unseres Gruppenlogos, welches nicht ablenkend wirkt. Bei der Typographie war ich auf unser Styleguide beschränkt, dabei ist die Schriftart klar lesbar.

### 4.2 Gestaltung der Wireframes

Um sich ein grobes Bild zu verschaffen, wie endgültig unsere Web-Applikation aussehen soll haben wir uns dazu geeinigt das Tool von [www.figma.com](http://www.figma.com) einzusetzen. Mit welches man einfach Wireframes für Webseiten erstellen kann.

Dabei werden verschiedene Elemente wie z.B. Rechtecke gruppiert und es lassen sich damit Fensteransichten anfertigen. Somit kann man verschiedene Wireframe miteinander verknüpfen und mit Ihnen Interagieren. Damit kann eine beispielhafte Webseite für z.B. Kunden dargestellt werden, ohne auch wirklich eine Applikation programmiert zu haben.

Auf der nächsten Seite wird ein Wireframe von unserem Admin-Dashboard



[Abb. 1: Wireframe des Admin-Dashboards]

## 5. Frontend Struktur

### 5.1 Allgemein

Grundsätzlich wird zwischen dem User-Dashboard und dem Projekt-Dashboard unterschieden. Da diese verschiedene Ansichtsmöglichkeiten und Funktionen haben. Zusätzlich gibt es für die Anmeldung und Registrierung einen eigenen Ordner. Bei der Aufgabenverteilung wollten wir das auch demensprechend so einsetzen, dass jeder für einen Bereich zuständig ist. Im späteren Verlauf mussten wir aber feststellen, dass es so nicht möglich ist und wir deshalb in allen Bereichen gearbeitet haben.

### 5.2 User-Dashboard

Bei der Struktur im User-Dashboard wurde zwischen den einzelnen Reitern in der Navigationsleiste unterschieden. Diese sind:



|                             |   |
|-----------------------------|---|
| <b>Projekte</b>             | Zeigt die aktuellen Projekte an, in denen man sich befindet |
| <b>Ankündigungen</b>        | Zeigt Ankündigungen von den jeweiligen Projekten an         |
| <b>Abgeschlossene Tasks</b> | Zeigt abgeschlossenen Tasks innerhalb eines Projekts an     |
| <b>Offene Tasks</b>         | Zeigt offenen Tasks innerhalb eines Projektes an            |
| <b>Account verwalten</b>    | Damit lassen sich benutzerdefinierte Einstellungen ändern   |
| <b>Raumbuchungen</b>        | Zeigt die Raumbuchungen an                                  |
| <b>Info</b>                 | Informationen über die Entwickler                           |

### 5.3 Projekt-Dashboard

Bei der Struktur im Projekt-Dashboard wurde zwischen den einzelnen Reitern in der Navigationsleiste unterschieden. Diese sind:

|                               |   |
|-------------------------------|---|
| <b>Tasks</b>                  | Zeigt alle Tasks innerhalb eines Projektes an   |
| <b>Bevorstehende Deadline</b> | Zeigt Tasks an, die innerhalb einer Woche bzw. innerhalb eines Monats zu erledigen sind |
| <b>Übersicht</b>              | Gibt eine Übersicht des aktuellen Projektes an  |
| <b>Ankündigungen</b>          | Zeigt Ankündigungen innerhalb eines Projektes   |
| <b>Tags</b>                   | Zeigt Tags an, die zur Verfügung stehen   |
| <b>Räume</b>                  | Gibt die Räume wieder die für das Projekt zur Verfügung stehen                          |
| <b>Info</b>                   | Informationen über die Entwickler   |

### 5.4 Weitere Elemente

Zusätzlich werden noch weitere Elemente benötigt, die dafür sorgen, dass die gesamte Seite gerendert wird. Diese beinhalten den Main-Bereich inkl. der Suchleiste und auch der Navigationsleiste.

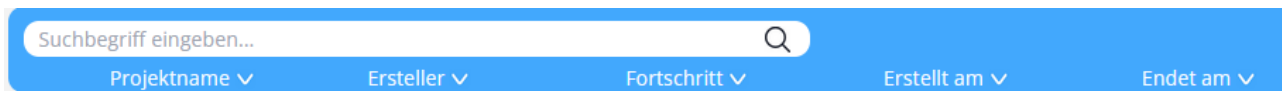
## 6. Frontend Implementierung

### 6.1 Allgemein

Wie ursprünglich angedacht, wollten wir eine klare Grenze ziehen, wer welche Komponenten für das Projekt erstellt. Dabei ist uns aber aufgefallen, dass es so unmöglich ist und wir deshalb an alle vorhanden Komponenten gearbeitet haben. Das hat dazu geführt, dass wir einen besseren Überblick über das gesamte Projekt hatten. Nachfolgend werden aber trotzdem die Komponenten erwähnt, an denen ich hauptsächlich beteiligt war.

## 6.2 Suchleiste

Meine Aufgabe war es die Suchleiste zu strukturieren, mit Tailwind zu gestalten und auch die richtige Platzierung zu definieren. Die Implementierung der Funktionen wie z.B. suchen eines Projektes hat der Kollege Dejan Fraas übernommen. Bei der Strukturierung habe ich diese in verschiedene Bereiche aufgeteilt zum einen den Bereich der Suche und zum anderen in den Bereich der Sortierelemente unterhalb der Suchleiste. Nachfolgend die Suchleiste als Bild.



[Abb. 2: Suchleiste]

## 6.3 Fortschrittbalken

Beim Fortschrittbalken wird der Fortschritt eines Projektes in einem Prozentualen wert dargestellt (siehe Beispiel)



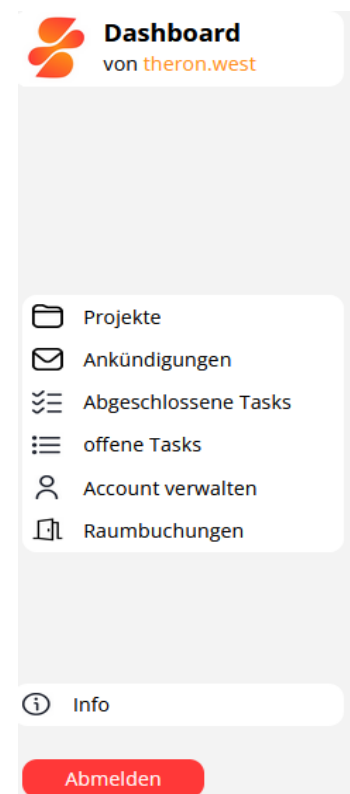
Dabei wurde der von der Datenbank geladene Fortschritt (ein Prozent-Wert berechnet aus den noch bestehenden Tasks) als Weite eines Div.-Elementes definiert. Zusätzlich wird noch der Prozentuale Wert angezeigt zu einer besseren Sicht.

## 6.4 Navigationsleiste – User-Dashboard

Die Navigationsleiste im User-Dashboard besteht aus unserem App-Logo, Benutzernamen und den verschiedenen Reitern wie in Punkt 5.2 beschrieben.

Der Zweck dieser Navigationsleiste ist, dass man durch die verschiedenen Ansichten wechseln kann, wenn man auf einen der Reiter klickt.

Es kommt noch ein Abmelde-Knopf hinzu, welches dafür sorgt, dass sich der Benutzer abmelden kann.



## 6.5 Navigationsleiste – Projekt-Dashboard

Die Navigationsleiste im Projekt-Dashboard unterscheidet sich lediglich aus den anderen Reitern wie unter Punkt 5.3 beschrieben.

[Abb. 3: Navigationsleiste des User-Dashboards]

## 6.6 User-Dashboard – Projekt

Die Projektansicht des User-Dashboards besteht aus einer Liste, aller vorhanden Projekten, in denen man sich befindet. Dabei wird aus der Datenbank ein Datensatz aus Projekten geladen.

Es enthält Informationen über den **Projektnamen**, den **Ersteller**, den **Fortschritt**, **wann** das **Projekt erstellt** wurde, bis **wann** das **Projekt beendet** werden soll und einer **Beschreibung**. Außerdem wird im Listeneintrag noch ein **Knopf** angezeigt, welches auf das jeweilige **Projekt weiterleitet**.

Da das zu viele Informationen auf einmal wären haben wir uns dazu entschieden, dass man einen Listeneintrag aufklappen kann. Dabei wird durch drücken des Pfeilsymbols der Eintrag aufgeklappt und es wird die Beschreibung dargestellt, was übersichtlicher wirkt.

Zusätzlich haben wir im unteren Bereich der Seite ein Knopf hinzugefügt, um Projekte erstellen zu können. Dabei wird ein Projekt mit den entsprechenden Attributen in die Datenbank geschrieben.

## 6.7 Projekt-Dashboard – Admin

Die Adminansicht im Projekt-Dashboard wird nur dem Ersteller des Projektes angezeigt. Diese Funktion dient dazu, dass auch nur der Ersteller des Projektes sensible Daten, wie z.B. die Mitglieder und Ihre jeweiligen Rechte innerhalb eines Projektes ändern kann. Dabei wird die Adminansicht in 3 Bereiche unterteilt:

### 1. Projektdetails

Bei den Projektdetails kann der Projektname, die Beschreibung und auch das Enddatum geändert werden.

### 2. Ankündigungen

In diesem Bereich können Ankündigungen für alle Mitglieder innerhalb des Projektes erstellt werden.

### 3. Mitglieder

Hierbei handelt es sich um eine Liste von allen Mitgliedern des Projektes. Wenn man den Listeneintrag aufklappt (selbes Vorgehen wie bei der Projektansicht im User-Dashboard) können die Rechte des jeweiligen Mitglieds geändert werden. Folgende Rechte kann ein Mitglied besitzen: Tasks erstellen, Tasks bearbeiten und Tags erstellen. Auch kann in der erweiterten Ansicht das Mitglied vom Projekt entfernt werden. Zusätzlich können in diesem Bereich auch neue Mitglieder hinzugefügt werden.

## 6.8 Projekt-Dashboard – Tasks

Die Tasks Ansicht des Projekt-Dashboards, besteht aus einer Liste, aller vorhandenen Tasks innerhalb eines Projektes. Es enthält Informationen über den **Titel**, den **Status**, die **Priorität** und der **Deadline**. Die Tasks lassen sich ebenfalls ausklappen, um erweiterte Informationen zu erhalten. Diese wären, Von **wem** der **Task bearbeitet** wurde, von **wem** der **Task erstellt** wurde, wann der Task **zuletzt aktualisiert** wurde und welche **Tags** erwähnt, wurden. Zusätzlich sind noch 2 Knöpfe enthalten, welche den Task abschließen bzw. abbrechen lassen können.

Auch hier befindet sich, ähnlich zum User-Dashboard – Projekt ein Knopf, der dafür sorgt, ein neuen Task zu erstellen.

## 6.9 Projekt-Dashboard – Deadline

Diese Ansicht ähnelt der Taskansicht sehr, da diese Tasks anzeigen lässt. Der Unterschied besteht darin, dass die Tasks gefiltert ausgegeben werden. Dabei wird gefiltert, ob der Task innerhalb der aktuellen Woche bzw. innerhalb des aktuellen Monats abgeschlossen werden muss.

Das hilft einen Überblick zu erhalten, welche Tasks bevorzugt behandelt werden sollen.

## 6.10 Projekt-Dashboard – Übersicht

In der Übersicht lässt sich der gesamte Fortschritt des Projektes anzeigen. Dies wird durch ein Tortendiagramm und einem Fortschrittsbalken verstärkt dargestellt, basierend auf die verfügbaren Tasks. Zusätzlich bietet die Übersicht Informationen über



[Abb. 4: Tortendiagramm]

## 6.11 Projekt-Dashboard – Ankündigungen

Die Ankündigungsansicht beinhaltet Ankündigungen vom Projektersteller in Form einer Listenansicht. Diese dienen zum Hinweis auf bestimmte Ereignisse.

## 6.12 Projekt-Dashboard – Tags

Bei der Tagsansicht können Tags erstellt werden. Tags dienen zur Gruppierung von bestimmten Tasks. Es erleichtert Tasks innerhalb eines Projektes zu unterscheiden und dient auch der Übersichtlichkeit. Außerdem wird angezeigt welche Tags bereits vorhanden sind. Wenn der Benutzer die benötigten Rechte besitzt, kann er unwichtige bzw. nicht mehr benötigte Tags wieder entfernen.

## 6.13 Registrierung

Die Registrierung dient dazu, um sich ein Benutzerkonto für Stiva zu erstellen. Benötigte Datensätze für einen Benutzer wären Vorname, Nachname, E-Mail, Nutzernamen und Passwort. Bei korrekter Eingabe der Daten, wird in der Datenbank ein Benutzer angelegt, der den vollen Funktionsumfang der Web-Applikation nutzen kann.

# 7 Testen & Errorhandling

## 7.6 Errorhandling

Beim Errorhandling sind wir agil vorgegangen. Bedeutet, wenn ein Fehler beim Programmieren festgestellt wurde, wurde er sofort behoben. Bei Fehlermeldungen die später festgestellt wurden, wurde der Fehler zuerst analysiert an welcher Komponente es liegen könnte, danach behoben. Typische Fehler waren z.B. falsch aufgerufene Routen von der API. Weitere typische Fehler waren Listen von Objekten wurden falsch geladen. Diese sind sogar gravierend aufgefallen, da es zum Systemabsturz der gesamten Seite gekommen ist.

## 7.7 Testen

Beim Testen wurden die verschiedenen Funktionen der Web-Applikation mit Benutzer Interaktionen getestet. Diese wurden parallel zur Programmierung durchgeführt. Zum Ende der Programmierphase wurde gemeinsam nochmal jede Funktion getestet, ob diese fehleranfällig ist.