

## Protokoll Meeting Herr Renz 01.04.2022

Anwesend: Gesamtes Studententeam, Herr Renz, Herr Watzko

### Vorstellung Use Case Diagram:

Guppen verwalten hinzufügen beim Admin - Gruppenverwaltung Raumbezogen

### Vorstellung Mockup:

Auswahl von Gebäude, Stockwerk, Raum geschickter visualisieren, z.B. Gebäude,

Stockwerk usw. Trennen in verschiedene Auswahlen

Buchbaren Arbeitsplatz als einzelnen Tisch darstellen

Message View erstellen

Türen setzen könnte evtl. Kompliziert werden, Tür eher als Aussparung oder gestrichelt ..

Library/Framework für Design (Wände, Türen, etc.)

Lieber Räume anstatt Stockwerke zeichnen, Raum besitzt keine weiteren Raumteiler

Vorgefertigte Layouts für Räume sind optional

SinglePage ist optional, Multipage wäre auch in Ordnung

Layout Erstellung Priorität auf Desktop, Mobile wäre nice to have

Zeitplan für uns und It-Designers erstellen mit Aufgabenaufteilung

Möglichst viel Verwendung von bereits erstellten Komponenten

### Die Fragen:

- Was wären die Unterschiede bei der Implementierung zwischen **Angular** und React?  
>
- Für welche Geräte sollte die App funktionieren? (Mobile First?)  
**Web, Mobile First**  
> Mobile First ist richtig
- Auf welchen Browsern soll die App funktionieren?  
> Chrome, Chromium must have, Firefox optional
- Wie soll der Login funktionieren? (Admin, Geschäftsführer, Mitarbeiter)
  - “Bestimmter Personenkreis” definieren  
**“Key Cloak”**  
>
- Haben im System Administrator und Geschäftsführer die gleiche Rolle?  
> Rollen trennen, Könnte Personen mit beiden Rollen geben  
Geschäftsführer ändert Layout nicht, könnte evtl. Die Berechtigungsverteilung machen für z.B Projekte
- Kann jeder Arbeitsplatz von jedem Mitarbeiter belegt werden? (Unterschiedliches Equipment?)  
> Geschäftsführer beschränkt die Buchbarkeit je nach Personengruppen  
Raumbezogene Beschränkungen, nicht je Arbeitsplatz
- Hat jeder Tisch einen eigenen Monitor / PC?  
> Ein Tisch = Ein Arbeitsplatz, siehe Feedback zum Mock up
- Gibt es unterschiedliche Tische? (Größe, Equipment etc.)  
> nein, zu komplex
- Kann man mehrere Tische gleichzeitig belegen?  
> Ist möglich und erwünscht

- Designwünsche? (Mock-Ups zeigen)  
> Mockup Feedback
- Vorstellungen, wie ein Administrator einen Raum anpasst bzw. Wie ein Raum allgemein visualisiert wird?  
>Mockup Feedback
- Hat jeder Tisch automatisch einen Stuhl / PC dabei?  
> siehe Frage Monitor/PC je Tisch
- Gelten die Tische in Pausen als frei?  
> Während Pausen gebuchte Tische nicht frei, weil ansonsten der Tisch jede Pause geräumt werden müsste
- Soll ein Tisch direkt, ohne Puffer, nach Nutzung wieder von einer anderen Person gebucht werden können? (Angst vor Überschneidung wenn einer später geht und der Andere früher kommt)  
> Kein Puffer, direkte Anschlussbuchung möglich
- Können auch Einzelplätze in Konferenzräumen gebucht werden, oder nur der ganze Raum auf einmal?  
> Konferenzräume nicht im Buchungssystem von Sitzplätzen, nicht in unserem Tool

### **Protokoll Regelmeeting Betreuung 04.04.2022**

Anwesend: Herr Watzko, Gesamtes Studententeam

Projektplan:

1. Milestones mit übernehmen
2. Gantt Chart mit Zeitbalken erstellen für bessere Abschätzung von Verzögerungen
  - a. Draw.io hat Gantt Charts z.B.

Präsentation mit Fokus vorbereiten, auf diverse Fragen vorbereiten

Projektmanagement bzgl. Kanban, Scrum

Techniken ausarbeiten - Architektur Diagramm - wie kommunizieren die Komponenten, wo sind ihre Grenzen

Funktionsumfang - Was für Funktionen umsetzen, Requirements mit Übergang zu den Mockups

Aufgeführte Punkte auch in die Doku übernehmen

Issues in Github zukünftig evtl. Nutzen

In Präsentation anmerken, dass Dienstags Teamtreffen sind und Montags Betreuertreffen

Kanban Board/ GitHub Board auch interessant in Bezug auf Projektmanagement

Protokolle entweder in die Repo oder auch als Anhang in die Doku

Dokumentation summiert die Anforderungen der Meilensteine auf

Zwischenpräse 15 min Umfang  
EndPräsi 20min Umfang  
Doku minimum 30-40 Seiten, mit Bildern mehr  
Keine Präsentationsgestaltung vorgegeben

Grundstruktur eines Latex Dokuments erstellen besitzt auch eine Wichtigkeit  
Eigener Github Acc. mit eigenen Commits Nutzen  
Namen mit Github Acc. Namen verknüpfen  
Problematiken in der Gruppe kommunizieren und ansprechen  
Low Fidelity Mockups sind ausreichend aber es soll jede Funktion dargestellt werden

### **Protokoll Regelmeeting Betreuung 11.04.2022**

Anwesend: Michael Watzko, Gesamtes Studententeam

Präsentation durchsprechen:

Feedback dazu:

- Am Besten einen eignen MongoDB Server aufsetzen, jedoch ist Atlas in Ordnung, wenn man dies problemlos später wechseln könnte
- Dockercontainer seine gut bei MongoDB
- Typescript könnte interessant in der Nutzung sein
- Startdatum hinterlegen
- Foliennummern hinzufügen

Zeitplan:

- Optional den Zeitplan mit in die Präsentation einbauen
- Technischer Prototyp besteht schon aus Implementation -> Verbindung von den Technologien zum Laufen bringen, erste Funktionsweisen testen
- Implementation wesentlich vorziehen auf KW 17
- Userstories vorverschoben aufgrund von jetzigem Stand
- UI Entwürfe wurden ebenfalls schon größtenteils durchgeführt
- Technisches Konzept wurde ebenfalls vorverlegt und bereits überlegt
- "Dokumentation ausarbeiten" wurde als lange Task angelegt ab KW 15

Nach der Präsentation liegt der direkte Fokus auf der technischen Implentierung und der Dokumentation

Die Meilenstein Abgaben finden hauptsächlich über die Dokumentation statt

GitHub sollte mit weiterem Fortschritt bestückt werden

- Sehr viel Push und Pulls sind wünschenswert, auch Testprojekte sind ok
- Alles bisher Erstelltes kann/soll in die Repo gepusht werden
- Repo gilt auch als Fortschrittsindikator

15 Min vorher sollte ausreichen für Tests der Präsentation

Alle werden präsentieren

## **Protokoll Regelmeeting Betreuung 25.04.2022**

Anwesend: Michael Watzko, Gesamtes Studententeam

Latex Tipp: jedes Kapitel kann man in eigene Dateien auslagern, mit includes einbinden...

Trick mit Tex Datei: Jeden Satz in eigene Zeile schreiben vermeidet Merge Conflicts, weil GIT zeilenweise arbeitet

Über GitHub Probleme von Pinar gesprochen (Commit, Push, Pull, Merge)

Über Doku gesprochen: Erklärungen zu den jeweiligen Themen von den Studenten

Problemstellung evtl. Erweitern mit Mietersparnissen durch weniger Personal vor Ort

Und Nachvollziehbarkeit/Protokoll von Buchungen

Technisches Konzept in nächster Zeit ausarbeiten und angeben, warum die Entscheidung für jene Technologie ausgefallen ist (und warum nicht für anderen)

Doku soll schlüssig und detailliert sein

Vorteile und Nachteile immer vergleichen gegenüber was?

Farbschema der Arbeitsplätze kann man selbst entscheiden, mit Begründung passen eigene Farbentscheidungen

Docker ist empfehlenswert

Abgabe in 2 Wochen, Zeitplan auch einfügen in die Abgabe

Zweite Hälfte des Projekt wird es Codereviews geben mit Her Nietzsche, jeder erklärt ein Stück Code von sich, mit dem Hintergrund, das jeder etwas programmiert hat

Apell zur gegenseitigen Kommunikation

Kanban Board Tasks in kleinere Unterpunkte aufbrechen für mehr Bewegung und kleinere, attraktive Aufgaben

Issues wären für den Punkt überhalb auch gut nutzbar

Issue-Tickets kann man dann in das Kanban Board hinzufügen

Issue Tickets kann man Leuten zuweisen

Labels(z.B Front End...) und Milestones(Deadlines) können ebenfalls hinzugefügt werden

Milestones zeigen an, wie viele Tickets offen oder abgeschlossen sind

Commits mit der Issuenummer werden dem Issue hinzugefügt, dort kann man auch diskutieren

Merge/Pull Request Strukturen evtl. Für die Zukunft aufbewahren

## **Protokoll Regelmeeting Betreuung 02.05.2022**

Anwesend: Michael Watzko, Gesamtes Studententeam

DokumentationsPDF durchgeschauen

Titelseite evtl. Mit Logos versehen

Titelseite mit Infos zum Studienprojekt versehen(SWTM)

Zeitplan um 90 Grad drehen, für eine größere Darstellung oder zerschneiden in zwei Bilder untereinander

Text zum Zeitplan hinzufügen

Überschrift Problem in Problemstellung ändern

Man, Sie, ihr.. Nicht in wissenschaftlichen Dokumentationen verwenden

Seitenränder/Vorlage in Latex abändern in Richtung schmalere Ränder

z.B. Dokumententyp

Bilder allgemein vergrößern

Bilder mal links mal rechts evtl. Anordnen für Abwechslung

Vspace in Latex für Leerzeilen benutzen

Plus/Minus Symbol ausschreiben

Table -> figure bei Tabellen

+/- evtl. in Farbe bei den Tabellen

Technische Begründungen sind bei der Technikwahl essenzieller

Besprechen der Diagramme:

Verteilungsdiagramm:

Verbindung User zu React zu NestJS

React und NestJS austauschen?

Komponentendiagramm:

Komponent Resteingang

Komponent Verarbeitung

Verbindungen symbolisieren, wie Komponenten andocken

Wer bietet Verbindungen an, wer nimmt sie an

Symbolbeschreibung auf <https://www.ionos.de/digitalguide/websites/web-entwicklung/komponentendiagramm/>

Welche Komponenten durchquert eine Anfrage

Entity-Relation Model

Kardinalitäten fehlen noch, 1,n

Uhrzeit Anfang und Ende in Kreisen verbunden mit Buchen

Oberer und unterer Pfad ist Fehl am Platz, Mitarbeiter Attribute mitgeben als

Geschäftsleitung und Administrator

[https://prnt.sc/bl7YEyq\\_7WZt](https://prnt.sc/bl7YEyq_7WZt)

Zu den Diagrammen auch beschreibenden Text

Neben der Doku auch die Implementierung beginnen

Für erste Implementations Abgabe evtl. Zwei Buchbare Arbeitsplätze, Keycloak Login

### **Protokoll Regelmeeting Betreuung 09.05.2022**

Anwesend: Michael Watzko, Gesamtes Studententeam

Doku soweit in Ordnung für die Abgabe am 10.05.2022

Durchschau von Datenbankanfragen

Milestone 2 tickets als erledigt markieren

Zukünftige Planung: Demo mit Login, Aufgabenteilung, Vorbereitung auf nächsten Milestone

Bibliotheken und Packages je nach License benutzen, komplette Komponenten auch License beachten

-> Rad nicht neu erfinden, möglichst das breite Funktionsspektrum abdecken

Milestone 3 mit Issues bestücken und assignen

Erkenntnisse/Reflexion in der Doku wäre ein Thema z.B. die nützliche Benutzung von Milestones und das Assignen von Issues

Datum auf dem Titelblatt genügt, keine Semesterangabe wegen verschiedenen Semestern

### **Protokoll Regelmeeting Betreuung 16.05.2022**

Anwesend: Michael Watzko, Gesamtes Studententeam

Feedback zur 1. Abgabe - Doku

Gegenseitig über den Text lesen und überarbeiten was auffällig ist.

Feedbackdokument sollte folgen auf Moodle

Rest Abfragen schützen, JWT Token im Backend verarbeiten, Verarbeitung evtl. über Plugin siehe Beispiel vorherige Gruppe (wird nachgereicht)

Fehlerhafte Weiterleitung beim Login gefixt

An der App an sich zu arbeiten ist wichtiger als am Login (Backend)

Termin mit Herr Renz wäre in dieser Woche zu vereinbaren, damit es vor der finalen Präsentation noch einen Termin mit Herr Renz ergibt (bis ca. 2 Wochen nach Meilenstein 3)

Angebote zum Praxissemester: <https://karriere.it-designers.de#stellenangebote>

Besprechung React Prototyp:

Rico wird Design Spezialist

Andere sollten Funktionen von Buttons implementieren

In zwei Wochen macht Herr Nietzsche Code Reviews im normalen Meeting

### **Protokoll Regelmeeting Betreuung 23.05.2022**

Anwesend: Michael Watzko, Gesamtes Studententeam

An Herr Renz erneut eine Mail schicken

An einem Fix für die direkte Colorierung der Tische gearbeitet

An einem Fix für die kurze Einblendung der Loginseite nach dem eigentlichen Login gearbeitet

Änderungen einfach direkt in die Dokukapitel einfließen lassen

Den Prototyp, der auch abgegeben wird, kann man taggen oder in einem Branch speichern

### **Protokoll Regelmeeting Betreuung 23.05.2022**

Anwesend: Michael Watzko, Jörg Nitzsche, Gesamtes Studententeam

Codereviews in Reihenfolge: Kyle, Jay, Rico, Dario, Pinar

7.06 neues Regelmeeting 17:30 Uhr

13.06 und 20.06 keine Meetings

23.06 neues Regelmeeting 17:30 Uhr

27.06 normales Meeting 17:30 Uhr

28.06 Abgabe

Websockets evtl. Für Nachrichten, Subscriber/Published

Json diekt Übertragung

Geht über Url, Websocket Objekte

Interessante Funktionen: OnErrors, OnOpen, OnClose, onMessage

SendFunktion -> Text verschicken und in JsonStruktur verpacken

Mit Öffnen der Seite oder Nachrichten wird ein Websocket je Nutzer erstellt

Unsaubere Version nur bei Aktualisierung der Ansicht werden neue Nachrichten aus der Db gezogen

-> Aber auch als Anfang der Nachrichtenchats nutzbar

### **Protokoll Meeting Herr Renz 02.06.2022**

Anwesend: Herr Renz, Herr Watzko, Gesamtes Studententeam

Herr Renz wird nicht an der Schlusspräsentation dabei sein

Feedback zur Anwendung:

Backend Routen Anbinden

Authentifizierung mitgeben - Kyle, Jay

Layout einfügen/editieren - Dario, Rico

Für Stufenlosen zoom ist .svg nützlich, fabricjs wäre eine Möglichkeit dafür

MessageSender kommt aus dem Token, später also überflüssig

Chat funktioniert so, dass man mit der Person schreibt, die im gewählten Zeitraum einen Konflikt mit eigener Buchung auslöst

Chat könnte automatisch gelöscht werden, wenn der Buchungszeitraum schon abgelaufen ist

Wenn User Nachrichten lädt und der Chat schon abgelaufen ist, löscht das Backend die Nachrichten die abgelaufen sind

Dokumentation vervollständigen bis zur Abgabe

UI sieht zufriedenstellend aus  
Dauerbuchung beachten  
Zeitbunchung und gebuchte Zeiträume könnte man trennen

### **Protokoll Regelmeeting Betreuung 07.06.2022**

Anwesend: Michael Watzko, Gesamtes Studententeam

Zu Message-Seite  
Falls Seitenwechsel bei Nachrichten nicht funktioniert, dann Hintergründe färben  
Flexbox für Input z.B.

Zum Backend/Auth Token:  
Maileinladung zum Repro einer vorherigen Gruppe könnte helfen  
<https://medium.com/devops-dudes/secure-nestjs-rest-api-with-keycloak-745ef32a2370>  
addroles

Zum LayoutDesigner:  
Layout Fenster entweder mit Canvas oder Bibliotheksvorschlag von Herr Renz

Präsentation vorüberlegen:  
Siehe Moodle:  
Thema, Problemstellung, Live-Demo  
Wie hat es sich entwickelt  
15 min Präsentation

Doku weiter arbeiten nach: Installationshandbuch (detailliert), Reflexion, Aufgabenteilung