



üK 223 – Multiuser Application Dokumentation

Thema: Full-Stack Multiuser Application

Autoren: Filip Franek, Nikhil Ilango, Henri Koch

Inhaltsverzeichnis

Ausgangslage	3
Auftrag	3
Setup & Link zum GitHub.....	4
Domänenmodell	4
Entity Relationship Diagramm (ERD)	5
Sequence Diagramm	6
Use Case Definition	7
Create Group Use Case.....	7
Join Group Use Case.....	7
Use Case Diagramm	8
Testing.....	9
Was ist Cypress	9
Was ist Postman.....	9
Testing unserer Applikation.....	9
Mit Cypress	9

Ausgangslage

Unser Team hat den Auftrag bekommen, eine Social Media-Website zu erweitern. Auf der Website sollen Users Gruppen sehen und ein Teil einer freigewählten Gruppe sein. Die Page wird von Admins und Users verwaltet. Die Admins haben die Möglichkeit neue Gruppen zu erstellen und zusätzlich alle Gruppen und Ihre Informationen zu sehen. Users können sich anmelden, eine Gruppe auswählen und beitreten.

Auftrag

Mithilfe dieses Projekts lernt man eine Full-Stack Applikation mit verschiedenen Tools zu entwickeln. Um das gewünschte Produkt zu bekommen, werden passende Komponente mit React, SpringBoot und PostgreSQL gebraucht. Dabei wird die Sicherheit der Webseite getestet und auf eine Setup-Anleitung geachtet.

Tools & Frameworks

Folgende Tools wurden benutzt, um das Projekt zu realisieren.

Für Backend & Datenbank:

- IntelliJ IDEA, Community / Ultimate Edition
- Docker Desktop
- SpringBoot

Für Frontend & Diagramme

- Visual Studio Code
- React mit npm / yarn
- Draw.io

Für Testing, Version Control & Dokumentation

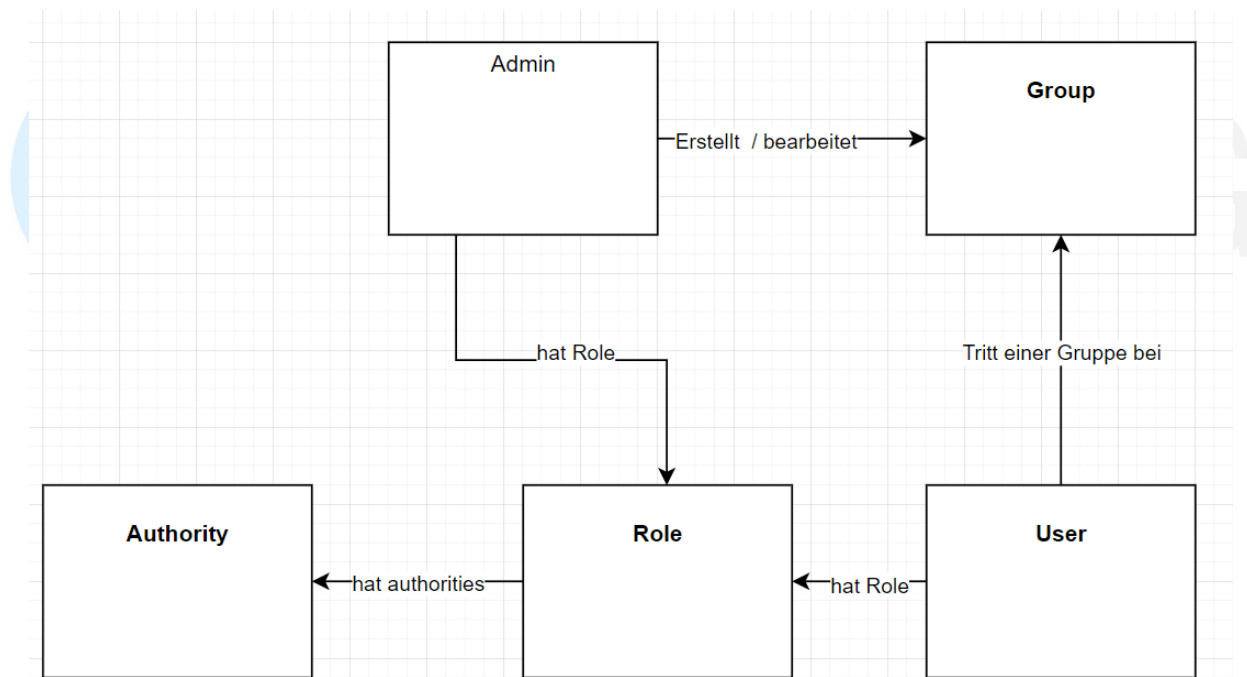
- Postman & Cypress
- GitHub (GitHub Desktop)
- Microsoft Word

Setup & Link zum GitHub

Im folgenden Link befindet sich das GitHub. In diesem [Link](#) kann man ein README.md finden, welches erklärt wie man die Applikation startet. Alle Files, Frontend und Backend, kann man auch auf dem GitHub finden.

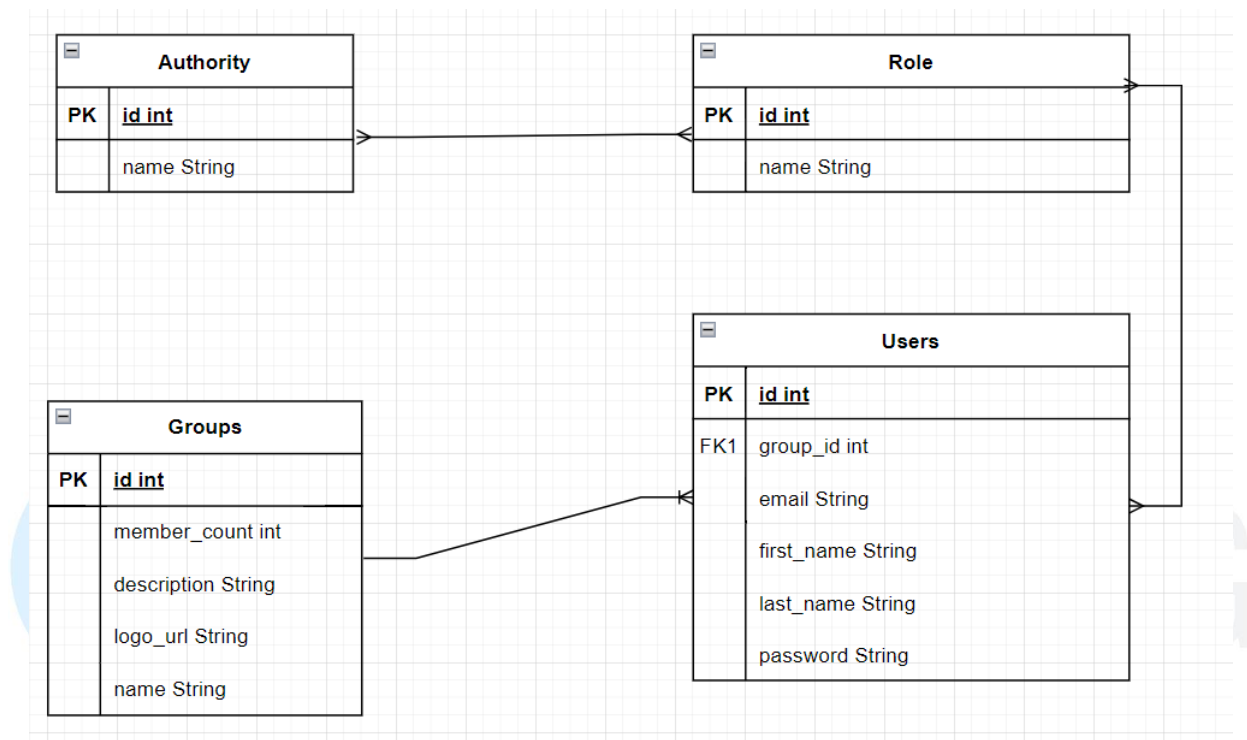
Domänenmodell

Dies ist das Domänenmodell für unsere Applikation, bei der man die Beziehungen der verschiedenen Entitäten nachvollziehen kann.



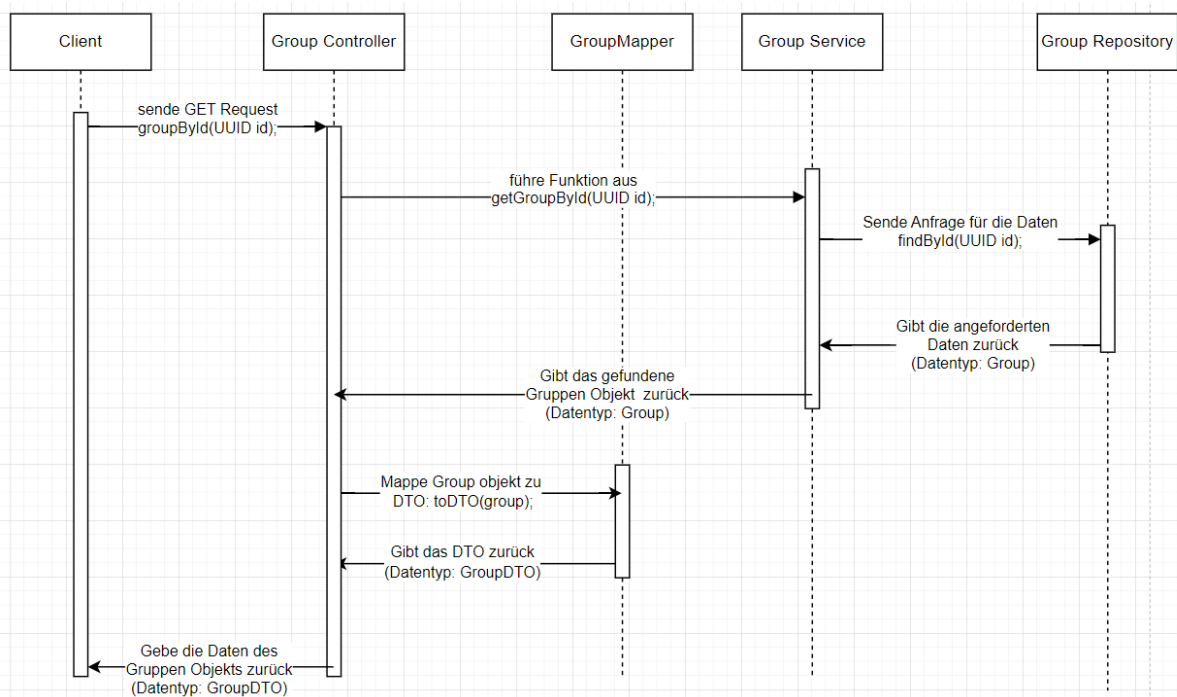
Entity Relationship Diagramm (ERD)

Dies ist unser Entity Relationship Diagramm, welches die Beziehungen zwischen den Entities darstellt.



Sequence Diagramm

Dies ist unser Sequence Diagramm bei dem wir den Prozess des Endpoints <http://localhost:8080/group/{id}> beschreiben.



Use Case Definition

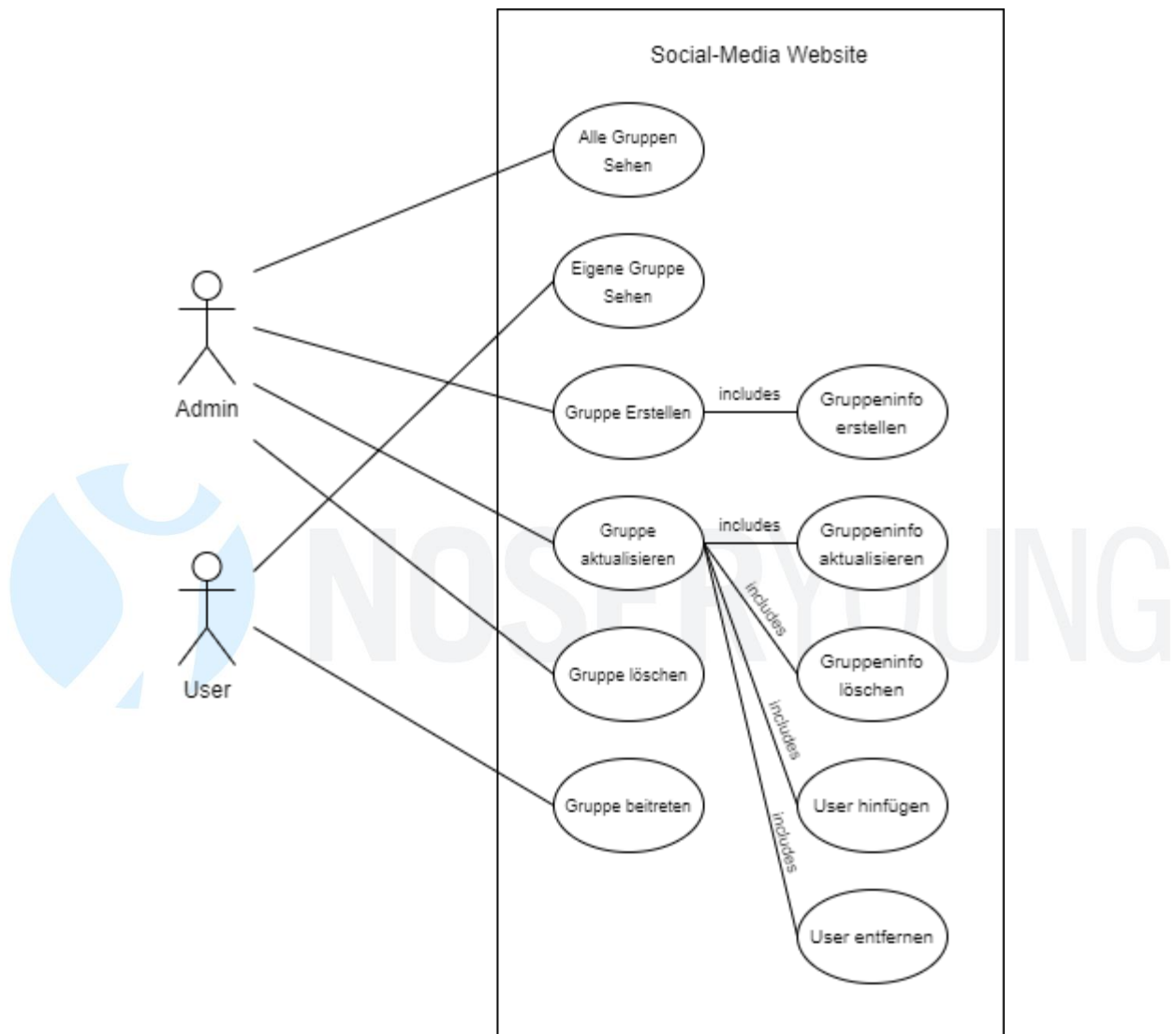
Create Group Use Case

Actor:	System Admin
Beschreibung:	In dieser Use Case Definition geht es darum, dass ein System Admin eine neue Gruppe erstellt. Dabei kann er alle Informationen, die für das erstellen einer Gruppe notwendig sind, selbst bestimmen.
Voraussetzungen:	<ul style="list-style-type: none"> - Group Create Authority - Gültigen Admin Account (anmelde Daten) - Information für eine neue Gruppe (Name, Beschreibung usw...)
Nachbedingungen:	<ul style="list-style-type: none"> - Der Admin hat die Gruppe erstellt - Man kann die Gruppe unter allen Gruppen finden - User können der Gruppe beitreten
Ablauf:	<ul style="list-style-type: none"> - Einloggen mit Admin Account - Admin klickt auf "Group Create" Knopf - Fülle All Daten ein für die neue Gruppe und erstelle sie dann - Admin wird auf Homepage geleitet - Der Admin sucht die Gruppe bei der Gruppensuche und findet sie
Alternativer weg:	-
Exceptions:	-

Join Group Use Case

Actor:	User
Beschreibung:	In dieser Use Case Definition tritt ein User einer bestimmten Gruppe bei. Damit er der Gruppe beitreten kann sucht er sie zuerst in der Gruppensuche.
Voraussetzungen:	<ul style="list-style-type: none"> - Der User braucht die NO_GROUP_USER Rolle - Gültiger User Account - Name der Gruppe der er beitreten möchte
Nachbedingungen:	<ul style="list-style-type: none"> - User kann nun keine anderen Gruppen mehr sehen - User kann genaue Informationen seiner Gruppe einsehen - User hat nun die Rolle GROUP_USER
Ablauf:	<ul style="list-style-type: none"> - User loggt sich ein - User sucht sich eine Gruppe aus - User tritt der Gruppe bei - User sieht die Informationen seiner Gruppe
Alternativer weg:	Ein User kann auch von einem Admin in eine Gruppe hinzugefügt werden.
Exceptions:	-

Use Case Diagramm



Testing

Für das Testing haben wir zwei verschiedene Tools benutzt. Cypress wurde für das End-to-End Testing des Frontends benutzt und Postman für das Komponent-Testing des Backends.

Was ist Cypress

Cypress-Tests laufen innerhalb von einem Browser. Es werden automatische Snapshots gemacht, welche bei der Fehleranalyse zur Hilfe gezogen werden können. Aufgetretene Fehler können leicht durch Developer Tools debuggt werden. Ein noch nicht implementierter Backend-Service kann simuliert werden.

Was ist Postman

Postman ist eines der beliebtesten Tools zum Testen von APIs. In unserem Fall testet es eine REST API auf HTTP-Basis. Postman konzentriert sich auf das Verarbeiten und Validieren von Requests und Responses. Man braucht nicht viel Programmierkenntnisse, um mit Postman arbeiten zu können. Es speichert bereits durchgeführte Tests ab und macht sie wiederholbar.

Testing unserer Applikation

Mit Cypress

Mit Cypress wollten wir unser ganzes Programm vom Anfang zum Ende testen. Cypress hat eine User-Interaktion simuliert und uns Snapshots geliefert. Diese haben wir analysiert und jegliche Fehler oder Probleme wurden behoben.

Die Use Cases die wir testen werden:

- [Create Group Use Case](#)
- [Join Group Use Case](#)

Mit Postman

Im Postman wurden unsere Endpoints getestet und CRUD-Operationen getestet. Wir haben sichergestellt, dass alles funktioniert und unsere User bekommen, was sie benötigen. Postman ist sehr einfach und schnell zu benutzen und hat uns gross geholfen, beim Verbessern unserer Applikation.

Endpoints die wir testen:

Method:	HTTP Urls:	Beschreibung:
GET	http://localhost:8080/group	Gibt alle Gruppen zurück
GET	http://localhost:8080/group/{id}	Gibt die gruppe mit der gewünschten id zurück.
POST	http://localhost:8080/group/create	Erstellt eine neue Gruppe
PUT	http://localhost:8080/group/update/{id}	Updatet die gruppe mit der gewünschten id.
PUT	http://localhost:8080/group/update/{group_id}/{user id}	Entferne oder füge neue User zu einer Gruppe hinzu.
DELETE	http://localhost:8080/group/delete/{id}	Lösche die gruppe mit der gewünschten id.