LB 2: Projektumsetzung

Food Delivery

Lazar Petrovic

Inhaltsverzeichnis

Beschreibung der Applikation

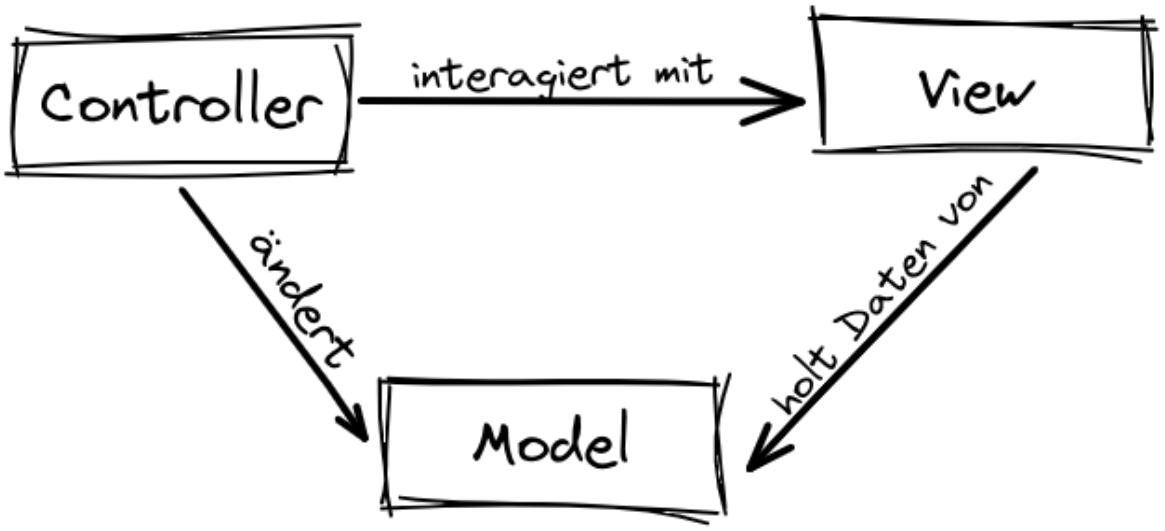
In der Food Delivery Web App ist es möglich, verschiedene Produkte aus der Datenbank, schön dargestellt, zu durchsuchen. Hat sich der User für etwas entschieden, kann er im UI Produkte in den Warenkorb setzen. Auf einer separaten Seite sieht er den Warenkorb und nachdem er sich seinem Entscheid sicher ist, wird er aufgefordert, benötigte Informationen, wie Adresse und Lieferort einzugeben.

Architektur

Die Architektur und das Design beschreiben den Aufbau und das Zusammenspiel von Komponenten aus den vorgesehenen Tiers.

Model **V**iew **C**ontroller

Das MVC Pattern ist eine gute Aufteilung der Komponenten für Entwickler einer Applikation, welche ein Front- und Backend hat. Dies vereinfacht uns Programmierern, uns in der App zurechtzufinden und an der Applikation zu arbeiten.



Quelle: Lazar Petrovic

Model

Alles, was mit Daten zu tun hat, wird vom Model gewartet. Daten aus dem Model können vom Controller geändert und durch die View gelesen werden.

View

Visuelle Komponente für die Seite, welches der Benutzer zu sehen bekommt (**U**ser **I**nterface). Ein Beispiel für einen UI-Komponent ist ein Button oder ein Input Field.

Controller

Im Controller ist die Logik für das Datenhandling, sowie für die Verhaltensweise des UI bei Benutzerinteraktionen.

3-Tier Architektur Konzept

Für mein Projekt wähle ich die 3-Tier-Architektur. Der Grund dafür ist die simple und einfach zu verstehende Aufteilung der Software Komponenten, welche mir ermöglicht, eine effiziente Applikation zu bauen. Dank einer grossen Auswahl an modernen Frameworks und Libraries existiert eine grosse Auswahl, durch die sich Entwickler anhand von eigenen Präferenzen entscheiden können.

Tiers and Frameworks/Libraries

Frontend

Der Presentation Layer meiner Webapplikation wird aus der Benutzer-ansicht und einem Service für Anfragen an das Backend bestehen. Das UI werde ich in **React** entwickeln. Für die Verknüpfung von Backend und Frontend werde ich Services mit HTTP Clients brauchen, um meine Endpoints aufzurufen.

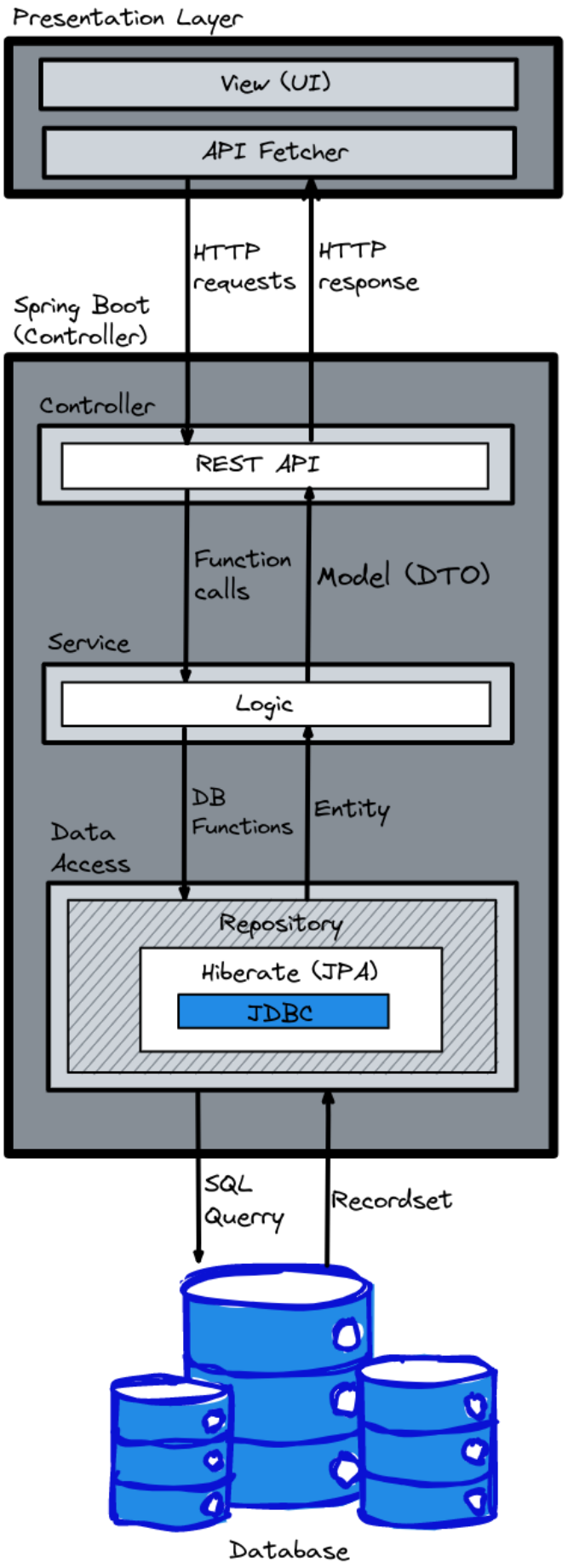
Backend

Spring Boot ist meine Wahl für die Backend- entwicklung. Ein Grund für meine Entscheidung ist die Sicherheit meiner Web App. JSON Web Tokens sind ein Beispiel, welches in Spring Boot sehr einfach implementierbar ist. Ausserdem ist das Integrieren von Tools wie Hibernate ohne grossen Aufwand möglich. Hibernate ermöglicht uns, sichere und effiziente SQL Queries zu generieren.

Datenbank

Als relationale Datenbank werde ich eine **Postgresql** Instanz verwenden. Postgres ist eine performante open source Datenbank, die Ähnlichkeiten zu MySQL aufweist.

Quelle: Lazar Petrovic



Frameworks

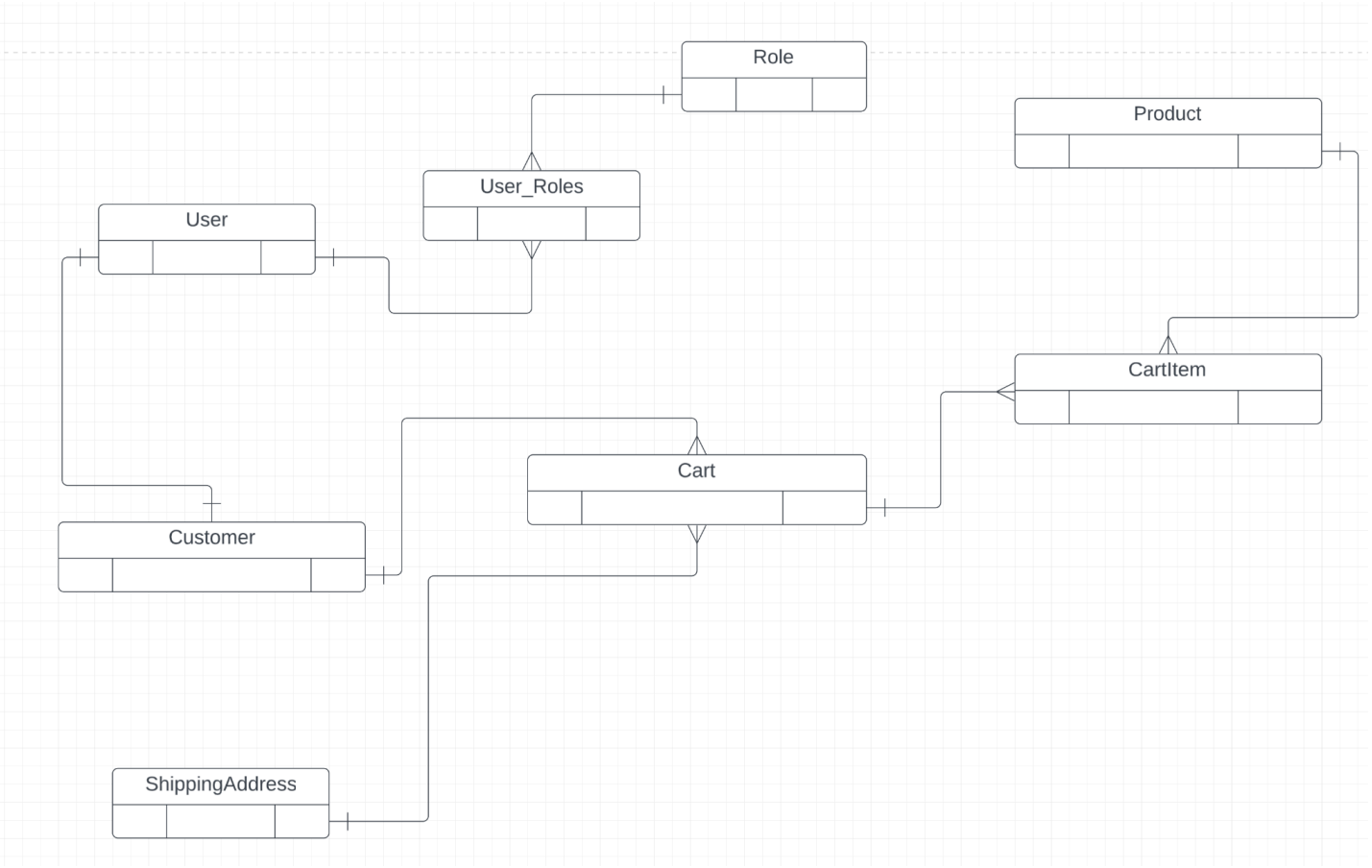
Frontend

React mit TypeScript.

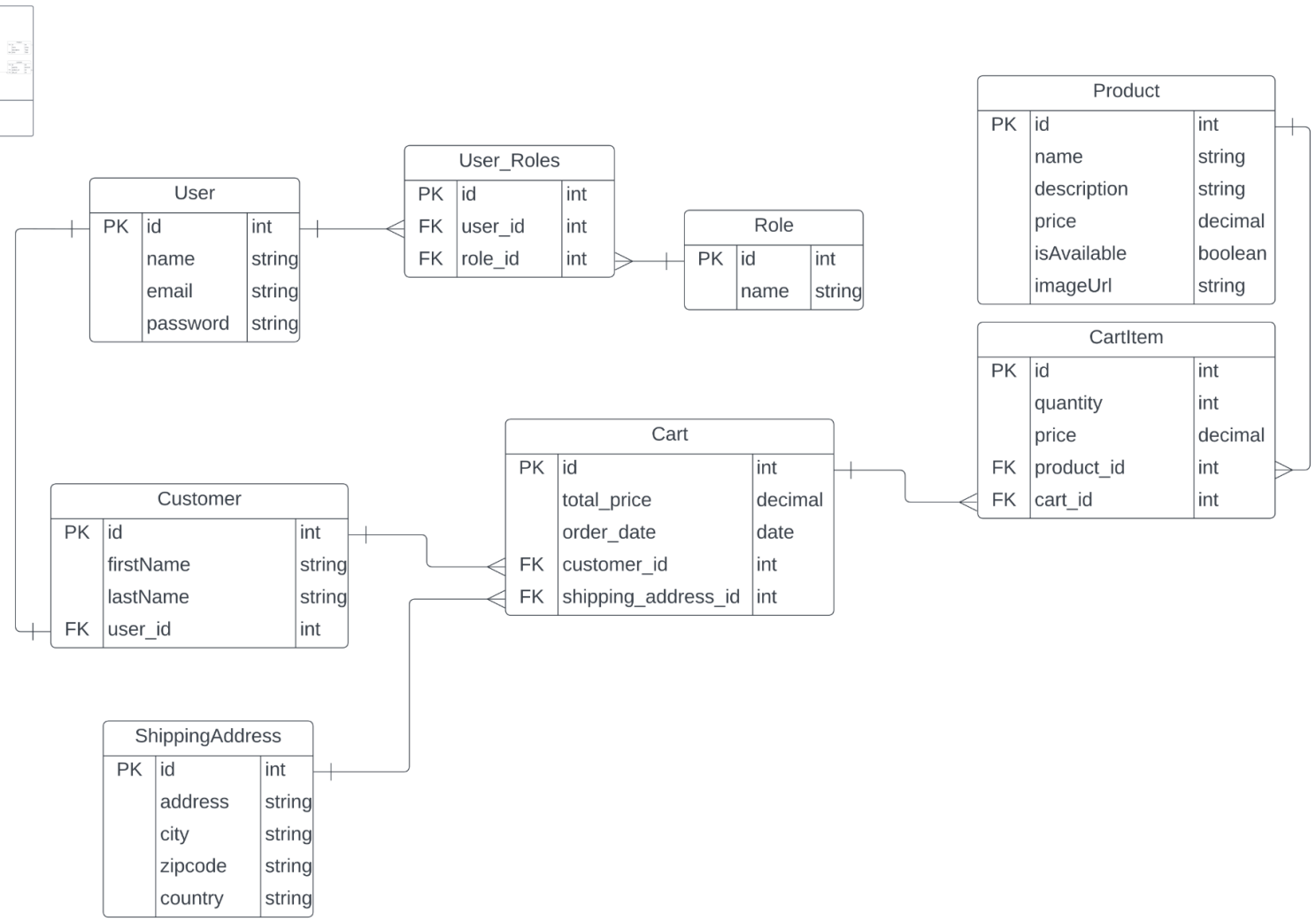
Backend

Spring Boot: 2.7.4 (Spring Version 5.3.15)

ERM


Role
Product
User_Roles
User
CartItem
Cart
Customer
ShippingAddress

ERD


Product
PK id int
name string
description string
price decimal
isAvailable boolean
imageUrl string
User_Roles
PK id int
FK user_id int
FK role_id int
User
PK id int
name string
email string
password string
Role
PK id int
name string
CartItem
PK id int
quantity int
price decimal
FK product_id int
FK cart_id int
Cart
PK id int
total_price decimal
order_date date
FK customer_id int
FK shipping_address_id int
Customer
PK id int
firstName string
lastName string
FK user_id int
ShippingAddress
PK id int
address string
city string
zipcode string
country string

Testkonzept

Unit testing

Um die Funktionalität einzelner “Units” zu testen in meinem Spring Boot Backend, würde ich auf JUNIT greifen, womit ich schnell die Applikation “mocken” kann, um so die einzelnen Stücke der Applikation zu testen.

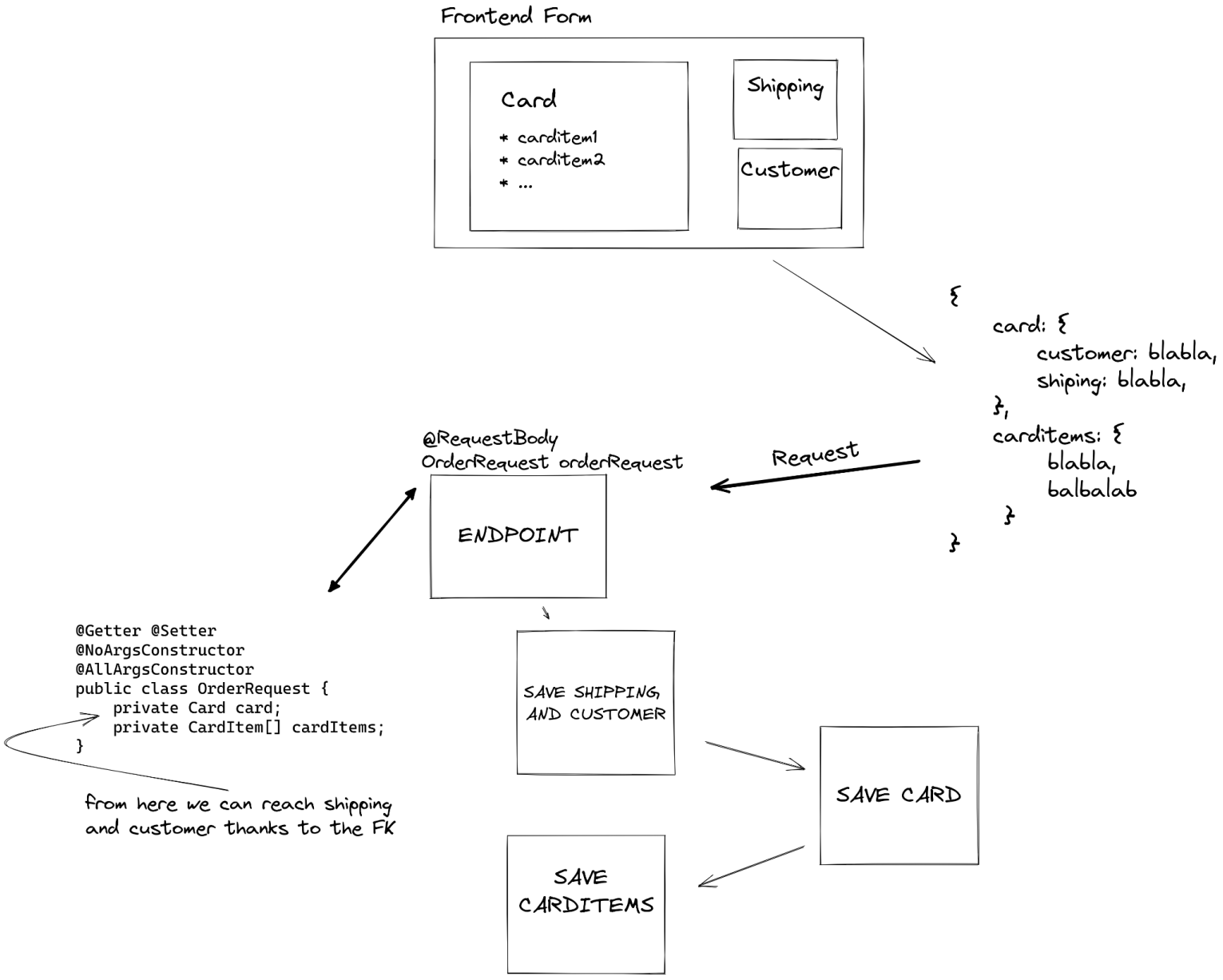
Postman testing
Endpoint Testing würde ich mit Postman machen, wo ich auch sonst meine Requests vorher geschrieben habe, um nicht manuell im Frontend arbeiten zu müssen.

Fazit

Ich bin sehr zufrieden mit der Applikation und am meisten stolz bin ich auf die JWT Implementation, welche einen Access und Refresh Token verwaltet. Gemeinsam mit Kollegen haben wir uns unterstützt und viel Neues gelernt.

Tuned sequence diagram :D

Hier ist ein Diagramm, welches ich während der Entwicklung an der App erstellt habe, um mir einen klaren Plan zu schaffen:



Schlusswort