

SEP Projektmappe

PROJEKTMAPPE DES PROJEKTES

SEP-Drive

Dokumentation des Projektes

Gruppe C:

Sara Akgüvercin

Aya Almalla

Rawan Almalla

Bashar Herzallah

Ibrahim Nagi Ahmed Mohamed Sayed

Rias Safi

Melike Ünver

Hinweis

An vielen Stellen findet Ihr im Dokument folgendes Kästchen:

Dies ist eine Hilfestellung.

Diese Kästen dienen dazu, Euch kurze Informationen über Ziele und Inhalte der jeweiligen Abschnitte zu geben. Auch die Beispiele und Templates dienen dazu, euch bei der Dokumentation eures Projektes zu unterstützen. **Sowohl die Kästchen als auch die Beispiele und Templates sind spätestens zur finalen Abgabe der Projektmappe vollständig zu entfernen.** Betrachtet dieses Dokument bitte nicht als Aufgabe, die man von oben nach unten abarbeiten soll; es soll vielmehr als durchgängige Dokumentation eurer Projektarbeit dienen und fortlaufend erweitert bzw. angepasst werden, sodass am Ende des SEPs der Entwicklungsprozess Eurer Software vollständig dokumentiert ist.

Das SEP-Team wünscht Euch
Viel Erfolg

Inhalt

Projektbeschreibung	4
Zyklus I	5
Spezifikationsplanung	5
User-Stories	6
Papierprototypen	7
Strukturdiagramme (Komponenten- und Klassendiagramme)	7
Verhaltensdiagramme (Kommunikationsdiagramme)	7
Funktionalitätsplanung	8
Systemtests	9
Zyklus II	11
Spezifikationsplanung	11
User-Stories	11
Papierprototypen	12
Strukturdiagramme (Komponenten- und Klassendiagramme)	12
Verhaltensdiagramme (Kommunikationsdiagramme)	12
Funktionalitätsplanung	13
Modultests	14
Systemtests	14
Zyklus III	15
Spezifikationsplanung	15
User-Stories	15
Papierprototypen	16
Strukturdiagramme (Komponenten- und Klassendiagramme)	16
Verhaltensdiagramme (Kommunikationsdiagramme)	16
Funktionalitätsplanung	17
Modultests	18
Systemtests	18
Nutzerhandbuch	19
Technische Anforderungen	19
Installationsanleitung	19
Bedienungsanleitung	19

Projektbeschreibung

In diesem Abschnitt soll die Projektbeschreibung abgedruckt werden, die ihr als Aufgabenbeschreibung von eurem Betreuer erhalten habt. Sie dient als initiales Anforderungsdokument für eure Spezifikationsaktivitäten.

Zyklus I

Spezifikationsplanung

Jedes Artefakt, das im Rahmen des SEP erstellt wird, muss mit dem Namen genau einer Verantwortlichen/eines Verantwortlichen versehen werden. Das bedeutet, dass jede User Story, jeder Papierprototyp, jedes Komponentendiagramm, jedes Klassendiagramm, jedes Kommunikationsdiagramm und sämtliche Tests mit dem Namen der/des Verantwortlichen versehen und hier in der untenstehenden Tabelle entsprechend eingetragen werden muss. Natürlich kann die Gruppe gemeinsam an einem Artefakt arbeiten, als Verantwortlicher sollte aber genau eine Person

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
1.	Registrierung der Nutzer			
1.1	Registrierung (Kunde/Fahrer)	User Story	Sara Akgüvercin	fertig
1.2	Registrierungs-Fenster	Papierprototyp	Sara Akgüvercin	fertig
2.	Login der Benutzer			
2.1	Login	User Story	Aya Almallal	fertig
2.2	Zwei-Faktor Authentifizierung	User Story	Aya Almallal	fertig
2.3	Login-Fenster	Papierprototyp	Aya Almallal	fertig
2.4	Zwei-Faktor Authentifizierung- Fenster	Papierprototyp	Aya Almallal	fertig
3.	Benutzerprofile			
3.1	Benutzerprofil (Kunde/Fahrer)	User Story	Rawan Almallal	fertig
3.2	Profilsuche	User Story	Ibrahim Sayed	fertig
3.3	Profilsuche-Fenster	Papierprototyp	Ibrahim Sayed	fertig
3.4	Benutzerprofil-Fenster (Kunde)	Papierprototyp	Rawan Almallal	fertig
3.5	Benutzerprofil-Fenster (Fahrer)	Papierprototyp	Rawan Almallal	fertig
4.	Kartenvisualisierung			

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
1.	Registrierung der Nutzer			
1.1	Registrierung (Kunde/Fahrer)	User Story	Sara Akgüvercin	fertig
1.2	Registrierungs-Fenster	Papierprototyp	Sara Akgüvercin	fertig
	von Routen			
4.1	Kartenanzeige & Interaktion	User Story	Sara Akgüvercin	fertig
4.2	Karten-Fenster	Papierprototyp	Sara Akgüvercin	fertig

5.	Fahranfragen			
5.1	Fahranfrage erstellen	User Story	Melike Ünver	fertig
5.2	Aktive Fahranfragen sehen und löschen	User Story	Melike Ünver	fertig
5.3	Aktive Fahranfragen Fenster	Papierprototyp	Melike Ünver	fertig
5.4	Fahranfrage-Fenster	Papierprototyp	Melike Ünver	fertig
6.	Systemarchitektur			
6.1	Komponentendiagramm Backend	Komponenten dia gramm	Bashar Herzallah	fertig
6.2	Klassendiagramm Backend + Frontend	Klassendiagramm	Rias Safi	fertig
6.3	Kommunikationsdiagramm Backend	Kommunikatio nsd iagramm	Ibrahim Sayed	fertig

User-Stories

User Story ID	1.1.1
User Story - Beschreibung	Als neuer Kunde möchte ich mich registrieren können, damit ich verfügbare Fahrzeuge suchen, buchen und nutzen kann
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	-

User Story ID	1.1.2
User Story - Beschreibung	Als Fahrer möchte ich mich registrieren können, damit ich Fahrten anbieten kann
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	-

User Story-ID	2.1
User Story-Beschreibung	Als Benutzer möchte ich mich mit meinem Benutzernamen und Passwort im System einloggen können, sodass ich Zugang zu meinen persönlichen Funktionen und Daten erhalte.
Geschätzter Realisierungsaufwand	1 Tag
Priorität	Hoch
Autor	Aya Almalla
Abhängigkeiten zu anderen User Stories	1.1

User Story-ID	2.2
User Story-Beschreibung	Als Benutzer möchte ich während des Logins zusätzlich einen per E-Mail gesendeten Sicherheitscode eingeben, sodass mein Konto durch eine Zwei-Faktor-Authentifizierung besser geschützt ist.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	Hoch
Autor	Aya Almalla
Abhängigkeiten zu anderen User Stories	2.1

User Story ID	3.1
User Story - Beschreibung	Als Nutzer möchte ich das Profil eines anderen Nutzers (Fahrer/Kunde) anzeigen können, sodass ich Informationen wie Name, Rolle, E-Mail-Adresse, Geburtsdatum, Fahrzeugklasse (nur für Fahrer), Rating und Anzahl der Fahrten sehen kann.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	Hoch
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	1.1 und 2.1

User Story ID	3.1
User Story - Beschreibung	Als Nutzer möchte ich im Profil die Anzahl der absolvierten Fahrten und die durchschnittliche Bewertung des jeweiligen Nutzers (Rating) sehen können, um Vertrauen in Erfahrung und Zuverlässigkeit zu gewinnen.
Geschätzter Realisierungsaufwand	1-2 Tage
Priorität	Mittel
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	5.1 und 5.2

User Story ID	3.2
User Story - Beschreibung	Als Benutzer möchte ich Fahrerprofile über eine Suchleiste nach Benutzername finden und anklicken können, damit ich gezielt den richtigen Fahrer auswählen kann
Geschätzter Realisierungsaufwand	1 Tag
Priorität	mittel
Autor	Ibrahim Sayed
Abhängigkeiten zu anderen User Stories	3.1

User Story ID	4.1
User Story - Beschreibung	Als Nutzer möchte ich meine geplante Route auf einer Karte sehen, damit ich nachvollziehen kann, wie ich von Start über Zwischenstopps zum Ziel gelange
Geschätzter Realisierungsaufwand	3-4 Tage
Priorität	Hoch
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	5.1 Karte braucht start und ziel, um etwas anzuzeigen

User Story ID	4.1
User Story - Beschreibung	Als Nutzer möchte ich die Karte verschieben und hinein- oder herauszoomen können, damit ich mir meine Route im Detail ansehen kann
Geschätzter Realisierungsaufwand	1 Tag
Priorität	hoch
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	4.1 Die Steuerung funktioniert nur, wenn Karte bereits angezeigt wird

User Story ID	5.1
User Story - Beschreibung	Als Nutzer möchte ich eine Fahranfrage erstellen, sodass ich den Startpunkt, das Ziel und die Fahrzeugklasse frei auswählen kann.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	hoch
Autor	Melike Ünver
Abhängigkeiten zu anderen User Stories	4.1, 4.2, 5.2

User Story ID	5.2
User Story - Beschreibung	Als Nutzer möchte ich aktive Fahranfragen auf einer speziellen Seite haben, sodass ich die Möglichkeit habe, diese zu löschen wenn nötig.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	hoch
Autor	Melike Ünver
Abhängigkeiten zu anderen User Stories	5.1

Template:

User Story-ID	<Eindeutiger Identifizierer>
User Story-Beschreibung	<Text der User Story mittels Satzschablone: Als <Rolle> möchte ich <Ziel> [, um/sodass <Nutzen>] (s. Foliensatz „Anforderungen“)>
Geschätzter Realisierungsaufwand	<Einschätzung der Zeit, die benötigt wird, um die Userstory zu implementieren>
Priorität	<Wichtigkeit der User Story hinsichtlich der Aufgabenstellung z.B. hoch, mittel niedrig>
Autor	<Hier bitte nur einen Zuständigen eintragen z.B. Max Mustermann>
Abhängigkeiten zu anderen User Stories	<Auflistung verwandter User Stories>

Schlechtes Beispiel:

User Story-ID	
User Story-Beschreibung	Ich möchte ich rechtzeitig informiert werden, wenn ein Patient einen Termin nicht wahrnimmt.
Geschätzter Realisierungsaufwand	1337
Priorität	-
Autor	Emmett Brown, Rick Sanchez, Amelia Pond
Abhängigkeiten zu anderen User Stories	

Gutes Beispiel:

User Story-ID	1.6
User Story-Beschreibung	Als Arzt möchte ich mindesten fünf Minuten vor dem Termin informiert werden, wenn ein Patient einen Termin nicht wahrnimmt, sodass ich andere Patienten vorziehen kann.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	Hoch
Autor	Emmett Brown
Abhängigkeiten zu anderen User Stories	1.3, 1.5

Papierprototypen

Das Erstellen eines Papierprototypen dient als Methode des Brainstormings, Designs, Herstellens, Testens und des Kommunizierens von Benutzer Interfaces.

1. Registrierung

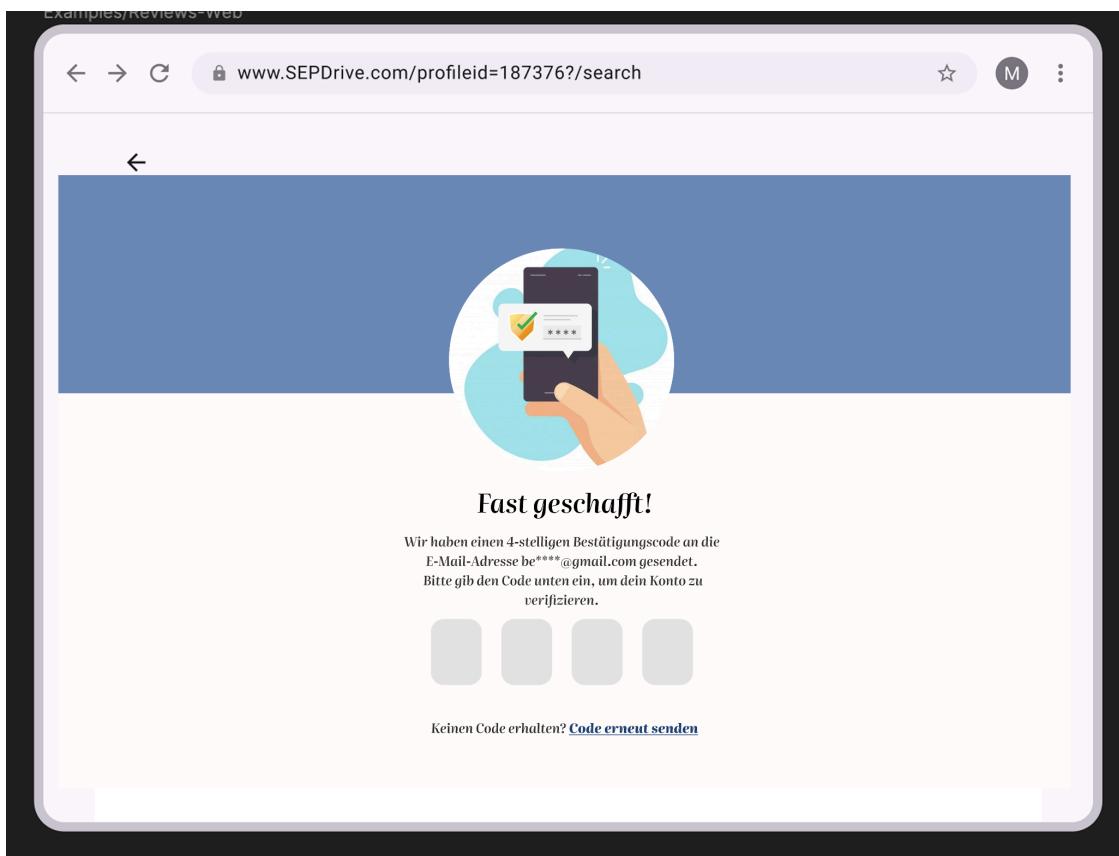
1.2 Registrierungs-Fenster

The screenshot shows a registration form for 'Carsharing'. At the top, there is a car icon and the word 'Carsharing'. Below it, the word 'Registrieren' is displayed. There are two tabs: 'Kunde' (selected) and 'Fahrer'. To the right is a circular profile icon with a plus sign. The form contains six input fields arranged in two rows: 'Vorname' and 'Nachname', 'Benutzername' and 'Geburtsdatum', and 'Passwort' and 'E-Mail-Adresse'. At the bottom are two buttons: 'Abschließen' and 'Abbrechen'.

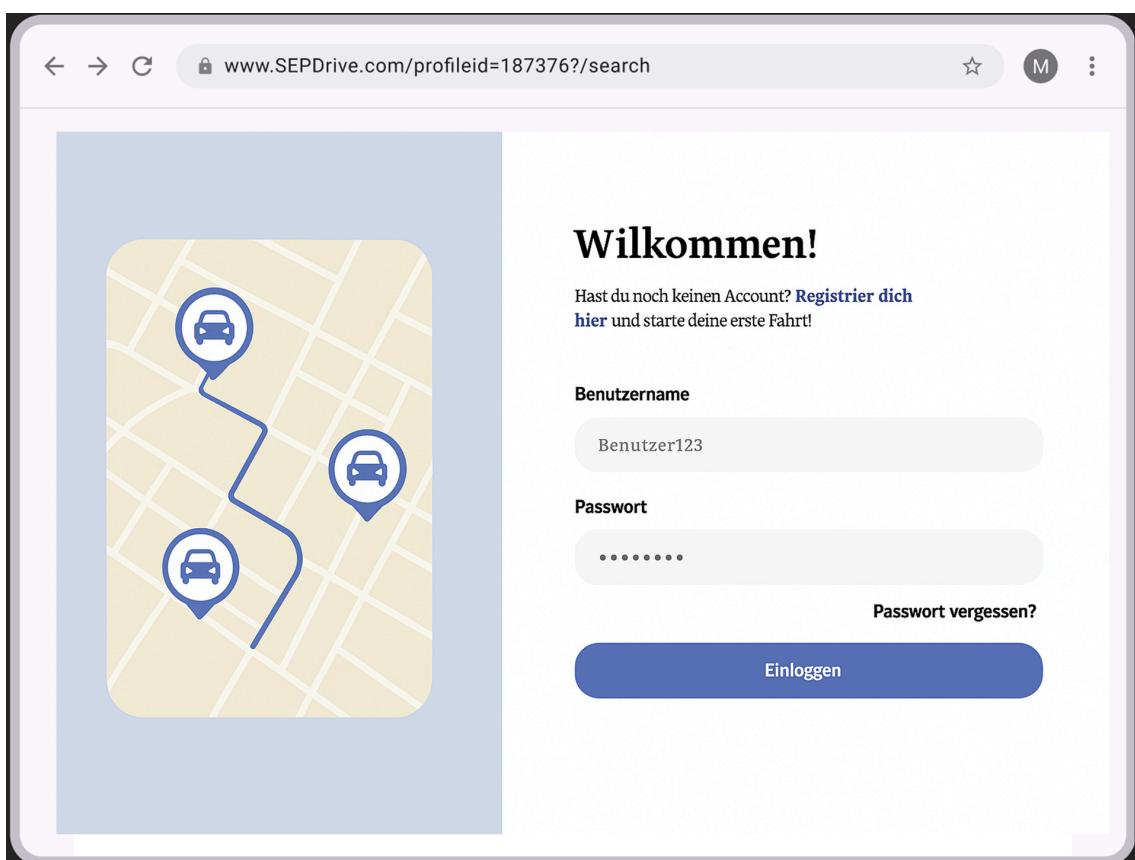
This screenshot shows the same registration form as above, but with an additional dropdown menu. The 'Auto Klassen' dropdown is open, displaying three options: 'Klein', 'Medium', and 'Deluxe'. The other fields and layout remain the same as in the first screenshot.

2. Login

Zwei-Faktor-Authentifizierung Fenster



Login Fenster



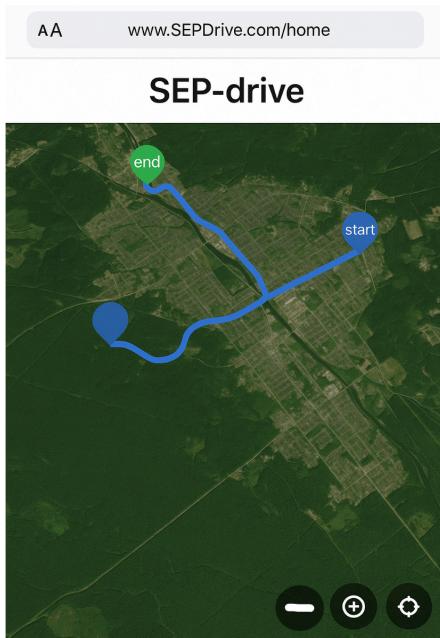
3. Benutzerprofile (Fahrer/Kunde)

The left screenshot shows a detailed profile for a driver named INagy. The profile includes a placeholder icon, the name 'INagy', the title 'Driver', and a heart icon. Below this, the driver's information is listed: Name: Ibrahim Sayed, Email: i.nagy80@yahoo.com, DOB: 12.04.1993, Car Class: Deluxe, and Rating: 4.9 ★★★★ 1245. At the bottom are two buttons: 'Message' and 'Hire Me'. The right screenshot shows a search results page for drivers. It has a header 'SEPDrive' and a search bar. Below are four driver profiles in a grid: Ron Greene (En-Class Standard, Rating 4.7 ★ 1152), Melanie Curtis (Car Class Economy, Rating 4.6 ★ 983), Jason Foster (Car Class Deluxe, Rating 4.9 ★ 1.287), and Linda Warren (En-Class Economy, Rating 4.8 ★ 1045). Each profile card includes a placeholder icon, name, title, email, DOB, car class, rating, and two buttons: 'Message' and 'Hire Me'. A 'Load More' button is at the bottom.

The left screenshot shows a customer profile for Amir. It features a placeholder icon, the name 'Profile', the title 'Amir', and the subtitle 'Customer'. Below this are the stats '400 Ride Token' and a yellow star rating '★ 4,9'. The driver's information is listed: Name: Amir Foster, Email: jan.cooper10@gmail.com, DOB: 10.07.1999. A blue speech bubble icon is on the right. The right screenshot shows a similar profile for Jano. It includes a placeholder icon, the name 'Profile', the title 'Jano', and the subtitle 'Driver'. Below this are the stats '400 Ride Token' and a yellow star rating '★ 4,9'. The driver's information is listed: Name: Jan Cooper, Email: jan.cooper10@gmail.com, DOB: 12.04.1996, and Car Class: Deluxe. A blue speech bubble icon is on the right.

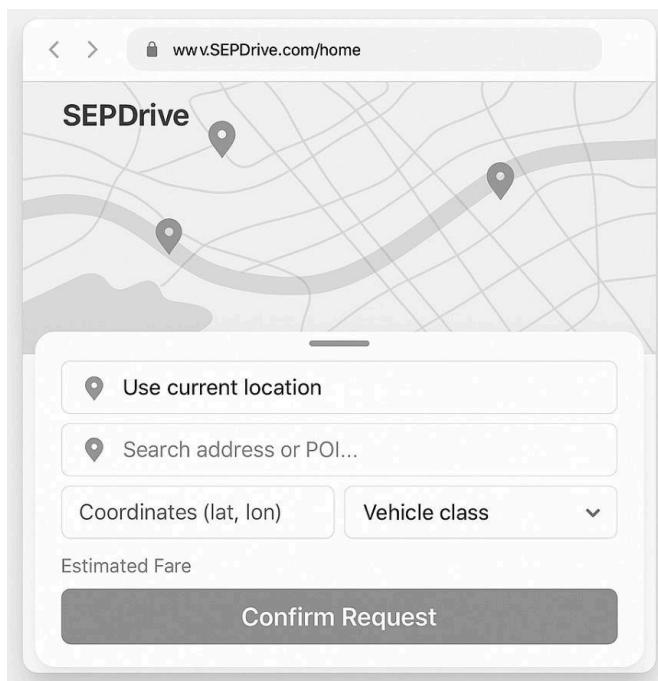
4. Kartenvisualisierung von Routen

4.4 Karten-Fenster



5. Fahranfragen

5.3



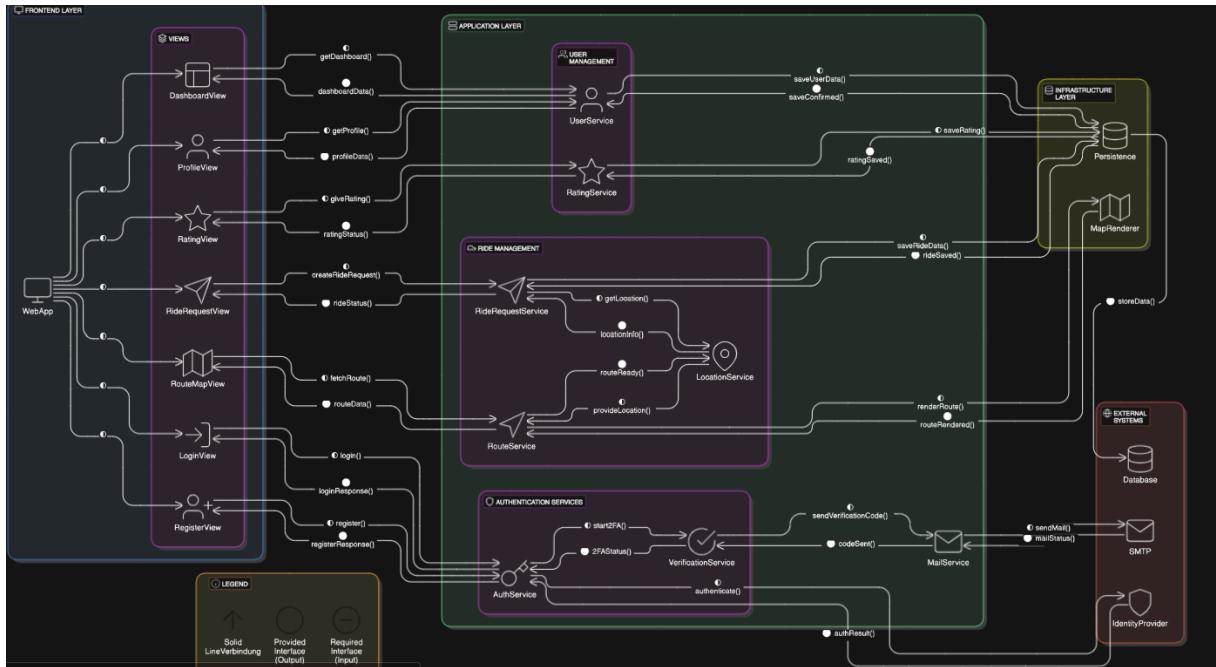
5.4

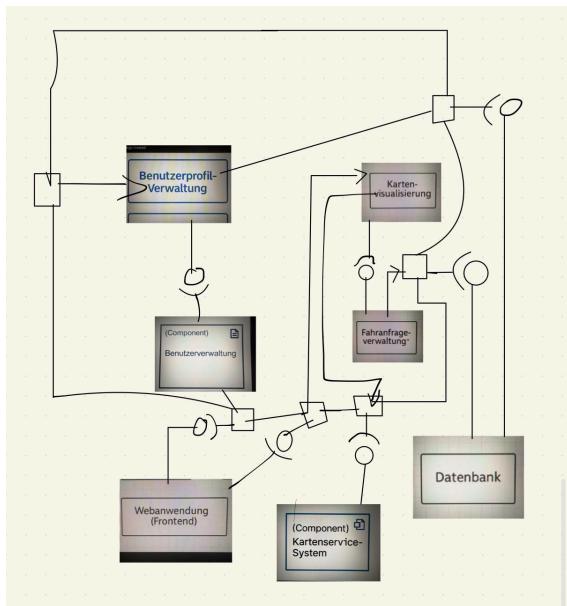
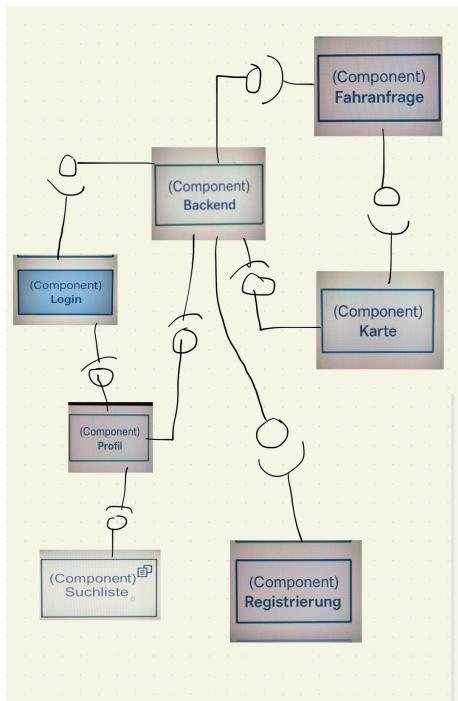
A screenshot of the SEPDrive mobile application showing an overview of travel requests. The title "Ihre Fahranfragen auf einem Blick" is at the top. Below it is a section titled "aktive Fahranfrage (max.1)" containing a single trip entry. The entry includes the start point ("Start: Essen Hbf") and end point ("Ziel: Duisburg Hbf"), followed by a checkbox and a delete icon, and the text "Anfrage löschen". Below this is a section titled "inaktive Fahranfragen" which lists two more trip entries: "Start: Herne Hbf, Ziel: Düsseldorf Hbf" and "Start: Essen Hbf, Ziel: Bochum Hbf".

Strukturdiagramme (Komponenten- und Klassendiagramme)

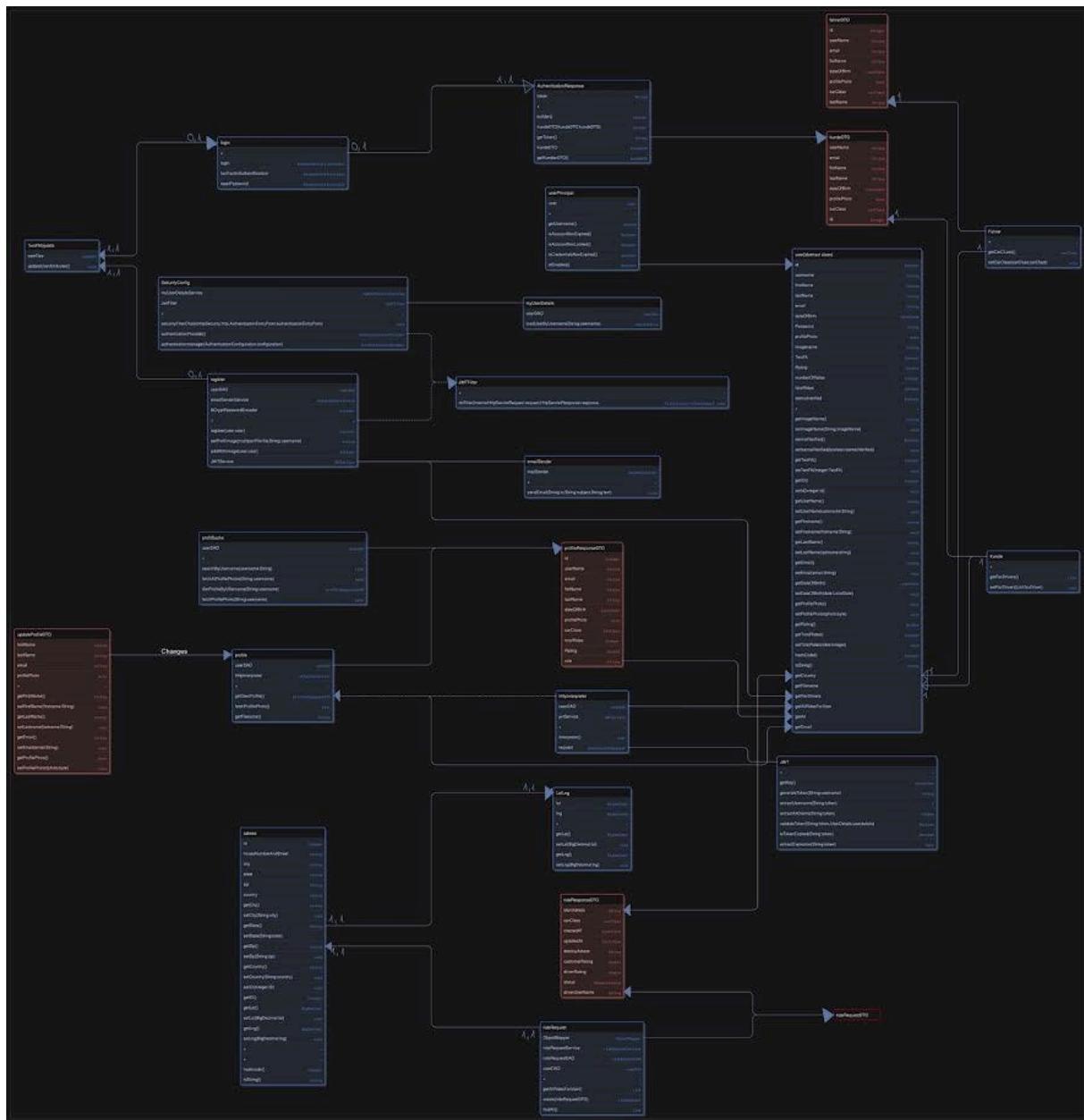
Im SEP soll die statische Struktur des Systems mittels Komponenten- und Klassendiagramme modelliert werden. Ein Komponenten- und Klassendiagramme dienen der grafischen Darstellung von Komponenten/Klassen, Schnittstellen und deren Beziehungen. Die Diagrammtypen helfen dabei, Quellcode und Implementierungsarbeiten zu strukturieren, bevor diese starten und ermöglicht eine Aufteilung der Programmieraufgaben.

This is Another Type of Component-Diagram (it is actually the same but the different is that the inputInterfaceIcon is Half-circle ,and the outputInterfaceIcon is circle) .

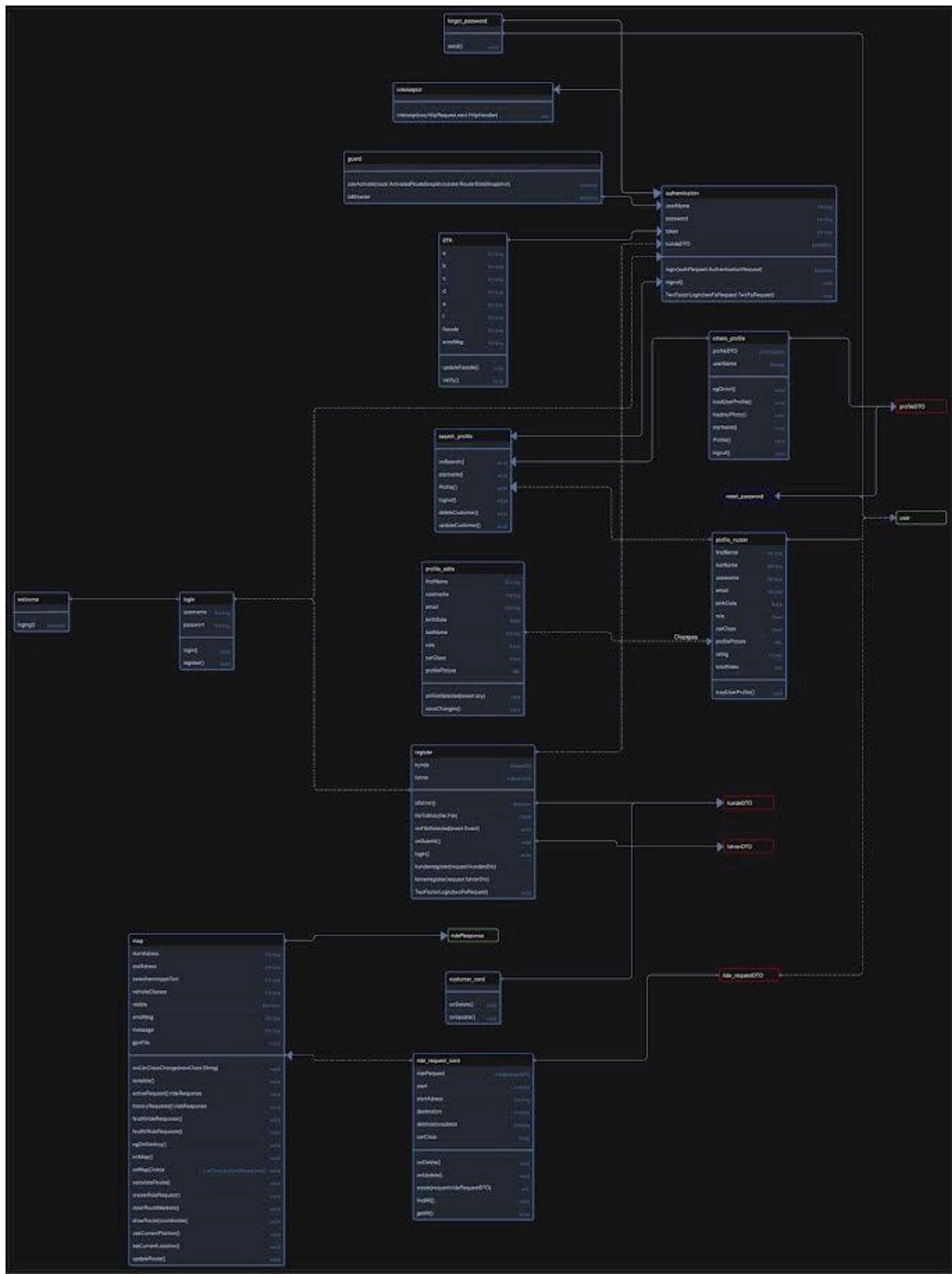




Backend Class-Diagram:



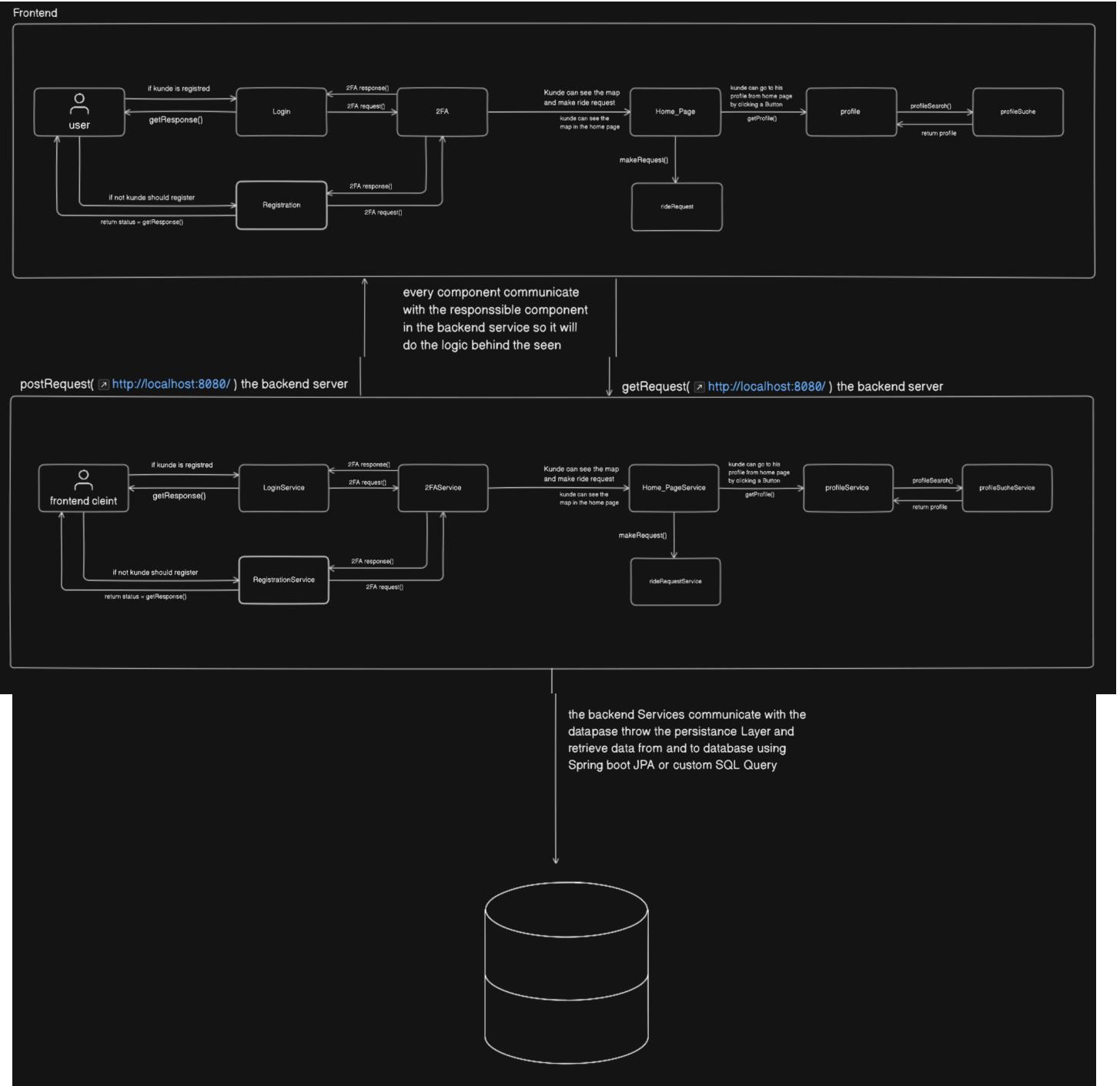
Frontend Class-Diagram:



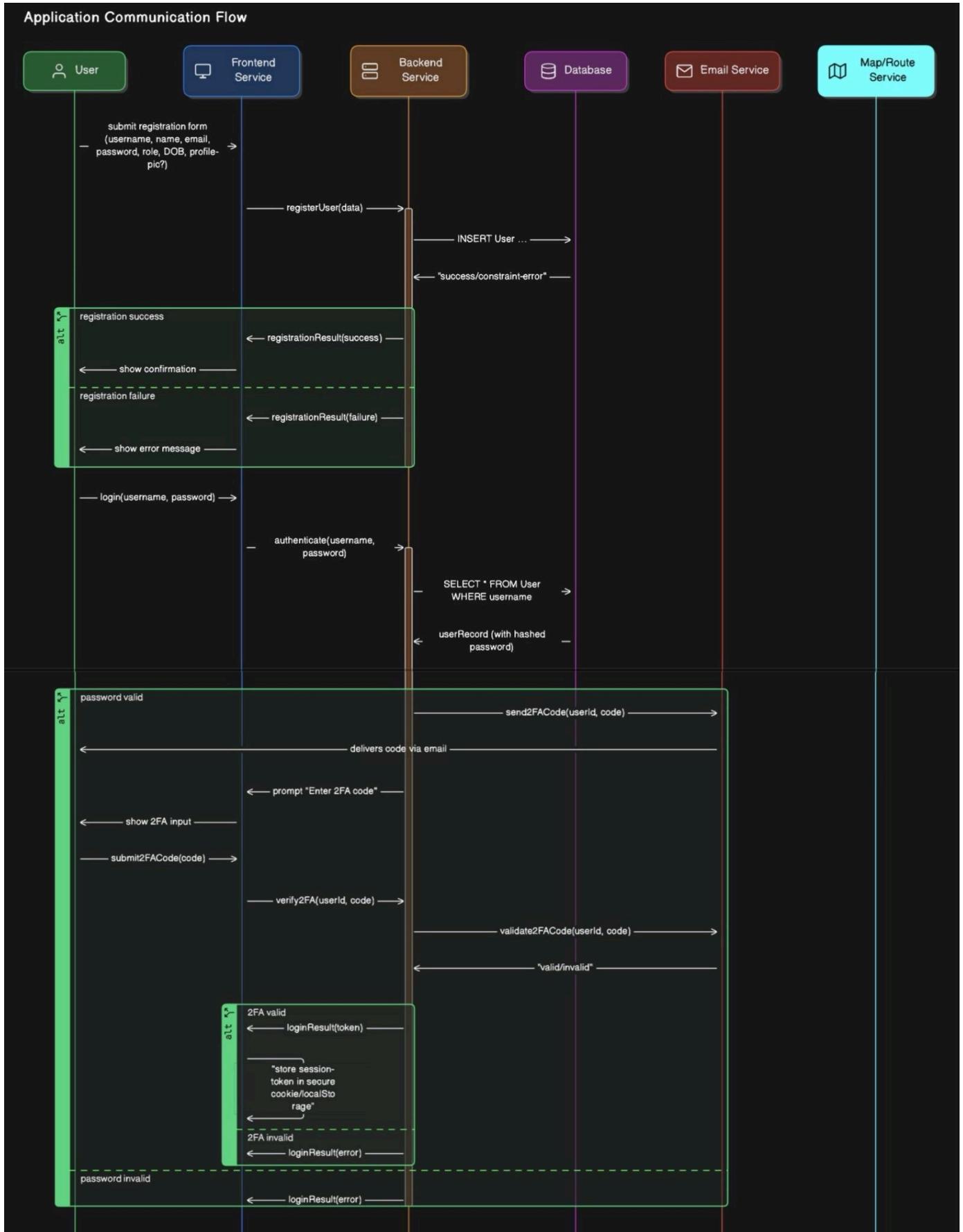
Verhaltensdiagramme (Kommunikationsdiagramme)

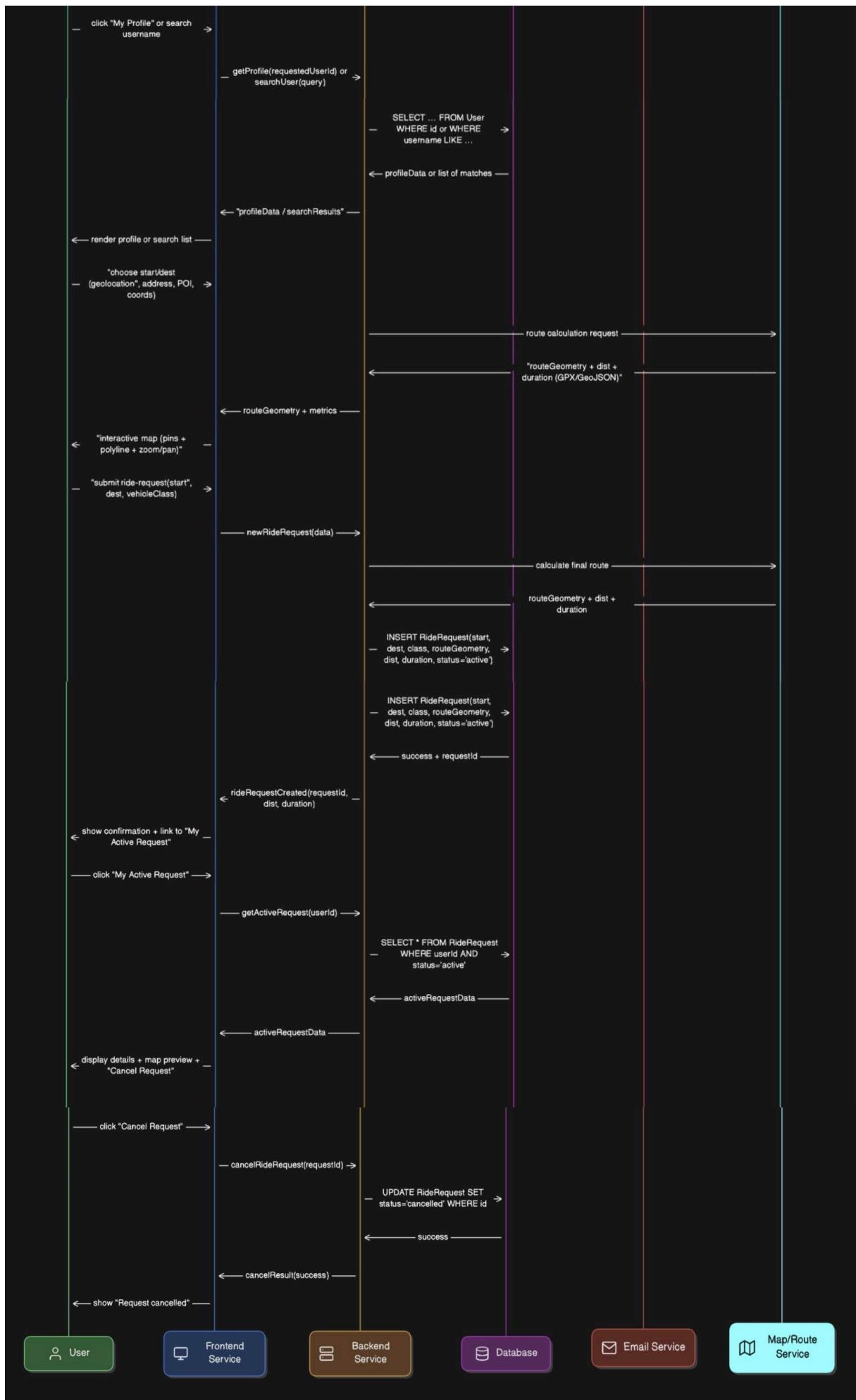
Im SEP soll das dynamische Verhalten des Systems mittels Kommunikationsdiagramme modelliert werden. Ein Kommunikationsdiagramme ermöglicht die grafische Darstellung des Nachrichtenaustausches zwischen Systemobjekten. Systemobjekte können Komponenten im Komponentendiagramm

und Klassen im Klassendiagramm sein. Kommunikationsdiagramme zielen darauf ab, die Zusammenarbeit der Systemobjekte darzustellen



I have made another variation for the communication that includes more detailed information but with another syntax :





Funktionalitätsplanung

Frontend Funktionalitätsplanung

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
1.	Registrierung von users					
1.1	Register-Komponente (Frontend)	Sara Akgüvercin	2.1, 2.2, 2.3	1.1.1, 1.1.2	Frontend/Angular/src/components/register	Fertig
1.2	Services : register.service.ts	Ibrahim Sayed	2.1, 2.2, 2.3	1.1.1, 1.1.2	Frontend/Angular/src/services/register	Fertig
1.3	Models : Kunde-dto	Sara Akgüvercin	2.1, 2.2, 2.3	1.1.1, 1.1.2	Frontend/Angular/src/models/Kunde-dto.ts	Fertig
1.4	Models: Fahrer-DTO	Sara Akgüvercin	2.1, 2.2, 2.3	1.1.1, 1.1.2	Frontend/Angular/src/models/Fahrer-DTO.ts	Fertig
2.	Login eines Benutzers					
2.1	Login Komponente (Frontend)	Aya Almallal	1.1	2.1	Frontend/Angular/src/components/login	fertig
2.2	Komponente für Passwort vergessen (Frontend)	Aya Almallal	-	-	Frontend/Angular/src/components/two-fa	fertig
2.3	Zwei-Faktor-Authentifizierung Komponente (Frontend)	Aya Almallal	2.1	2.2	Frontend/Angular/src/components/forgot-password	fertig
3.	Benutzerprofile					

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
3.1	Benutzerprofil View für Kunden und Fahrer im Frontend	Rawan Almallal	2.1	3.1	Frontend/Angular/src/app/components/profile-nutzer	Fertig
3.2	Profilsuche über Name view	Rawan Almallal	2.1, 3.1 und 3.1	3.2	Frontend/Angular/src/app/components/search-profile	Fertig
3.3	Profile bearbeiten view	Rawan Almallal	1.1		Frontend/Angular/src/app/components/profile-edit	Fertig
3.4	Profile Card View	Rawan Almallal	3.2		Frontend/Angular/src/app/components/custom-card	Fertig
4.	Kartenvisualisierung					
4.1	View für Karte	Melike Ünver	-	4.1	Frontend/Angular/src/app/components/map	Fertig
4.2	View für Fahranfrage Frontend	Melike Ünver	4.1	4.1, 5.1	Frontend/Angular/src/app/components/map	Fertig
4.3	View für aktive Fahranfrage mit Löschbutton Frontend	Melike Ünver	-	5.2		Fertig
5.	Startseite					
5.1	Willkommen Komponente (Frontend)	Aya Almallal	-	-	Frontend/Angular/src/components/welcome	fertig

Backend Funktionalitätplanung

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
1.	Registrierung von users					
1.1	Registrierung Service	Bashar Herzallah		1.1		fertig
2.	Login eines Benutzers					
2.1	LoginService	Bashar Herzallah		2.1		fertig
2.2	2FA code generator	Bashar Herzallah	2.1	2.2		fertig
2.3	EmailService	Bashar Herzallah	2.1			fertig
3.	profile					
3.1	profileService	Rias Safi	5	3.1	backend/src/main/java/SEPDrive/service (zusätzliche Klassen in backend/src/main/SEPDrive/controller)	fertig
3.2	profileSuche Service	Rias Safi	3.1	3.2	backend/src/main/java/SEPDrive/service (zusätzliche Klassen in backend/src/main/SEPDrive/controller)	fertig
4.	fahrAnfrage					
4.1	fahrAnfrage Service	Ibrahim Sayed		5.1		fertig
4.2	fahranfrage History service	Ibrahim Sayed	4.1	5.2		fertig
4.3	RatingService	Rias Safi	4.1		backend/src/main/java/SEPDrive/model	fertig
4.4	Mapp and location Service	Ibrahim Sayed	4.1	4.1		fertig
5.	Database					

5.1	Datenbank-modell des users	Ibrahim Sayed				fertig
6.	Docker					
6.1	Docker Images for Backend and Frontend	Ibrahim Sayed				fertig
6.2	Dockercontainers and docker-compose	Ibrahim Sayed	6.1			fertig

Systemtests

Systemtests sind Tests des Gesamtsystems gegen die Anforderungen nach erfolgreicher Integration. Eingaben und Sollverhalten werden dabei aus der Anforderungs-spezifikation abgeleitet.

Die Systemtests werden von Eurer Parallelgruppe spezifiziert und durchgeführt, daher ist dieser Bereich von den Mitgliedern der Parallelgruppe auszufüllen.

Datum	15.05.2025		
Tester	Bashar Herzallah		
SW-Version	V 0.2.0		
Vorbedin-gung(en)	Nutzer „Herzallah Test“ ist mit Passwort „passwort123“ registriert		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer gibt den Benutzernamen „Herzallah Test“ ein.	System zeigt „Herzallah Test“ auf dem Display.	✓
2	Der Benutzer gibt das Passwort „falsch123“ ein.	Passwort wird zensiert dargestellt.	✓
3	Der Benutzer klickt auf „Anmelden“.	System zeigt „Passwort falsch“ an.	✓
Nachbe-dingung(en)	Nutzer ist nicht angemeldet, kein Anmeldezeitpunkt gespeichert		
Testurteil	Test bestanden.		

Datum	15.05.2025		
Tester	Bashar Herzallah		
SW-Version	V 0.2.0		
Vorbedin-gung(en)	Anmeldemaske ist geöffnet.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer lässt das Feld für den Benutzernamen leer.	System zeigt Hinweis „Benutzername erforderlich“.	✓
2	Der Benutzer gibt Passwort „abc123“ ein.	Passwort wird zensiert dargestellt.	✓
3	Klick auf „Anmelden“.	Anmeldung wird verweigert, Hinweis erscheint.	✓
Nachbe-dingung(en)	Keine Anmeldung erfolgt. Kein Zugriff auf das System		
Testurteil	Test bestanden.		

Datum	19.05.2025		
Tester	Bashar Herzallah		
SW-Version	V 0.2.0		
Vorbedin-gung(en)	PostgreSQL-Datenbank aktiv (Docker läuft) Backend gestartet (Spring Boot auf Port 8080) Angular-Frontend läuft auf " http://localhost:4200 (npm start) " Browser geöffnet und auf " localhost:4200 "		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Ich habe die Website geöffnet, habe mich registriert mit Benutzername, Vorname , Nachname ,Passwort,Geburtsdatum, E-Mail registriert . Ich habe mich erfolgreich eingeloggt und zuerst den normalen Verifizierungscode(82421 3) aus der E-Mail eingegeben. Danach habe ich mich nochmal ausgeloggt und mit dem Super-Code (999999) eingeloggt , um zu prüfen, ob die Alternative greift..	Login-Maske wird angezeigt → Registrierung erfolgreich → Mail wird gesendet → Benutzer gibt Code ein → Login erfolgt → Startseite erscheint mit Benutzerinfo Nutzer wird gespeichert, E-Mail mit Code wird gesendet, Login erfolgreich, JWT-Token wird generiert.	✓
2	Ich habe auf der Profilseite ein Profilfoto hochgeladen und anschließend über die Suchleiste nach einem Fahrerprofil gesucht.	Das Profilfoto wurde korrekt angezeigt. Die Suchergebnisse wurden geladen und alle relevanten Informationen erschienen.	✓

3	<p>Ich habe über die Kartenansicht eine Fahrtanfrage erstellt, indem ich Start- und Zieladresse markiert sowie eine Fahrzeugklasse ausgewählt habe. Danach habe ich auf „Anfrage Bestätigen“ geklickt.</p>	<p>Fahrt wird erstellt, Adresse wird gespeichert, Anfrage als aktiv markiert, Die Route wurde auf der Karte angezeigt. Nach dem Absenden wurde die Anfrage erfolgreich gespeichert. rideRequestDTO und rideResponseDTO enthalten alle Informationen</p>	√
Nachbe-dingung(en)	<p>Der Benutzer war korrekt registriert, verifiziert und eingeloggt</p> <p>Das Profilfoto wurde erfolgreich im System gespeichert</p> <p>Der Nutzer hat ein verifiziertes Profil</p> <p>Eine gültige Fahrtanfrage ist im System unter dem eingeloggten Nutzer gespeichert und abrufbar</p> <p>JWT-Token wurde generiert und verwendet</p> <p>.</p>		√
Testurteil	Test bestanden		

Zyklus II

Spezifikationsplanung

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
7. Geldkonten				
7.1	Geldkonto-Funktion für Kunden	User Story	Melike Ünver	fertig
7.2	Geldkonto-Funktion für Fahrer	User Story	Melike Ünver	fertig
7.3	Geldkonto-Ansicht	Papierprototyp	Melike Ünver	fertig
8. Liste verfügbarer Fahranfragen				
8.1	Fahreranfrage-Liste (sortierbar)	User Story	Sara Akgüvercin	fertig
8.2	Positionsangabe zur Entfernungsmessung	User Story	Sara Akgüvercin	fertig
8.3	View der Fahranfragen	Papierprototyp	Sara Akgüvercin	fertig
9. Fahrtplanung				
9.1	Automatische Routenplanung für Anfragen	User Story	Aya Almallal	fertig
9.2	Berechnung: Distanz, Dauer, Preis	User Story	Aya Almallal	fertig
9.3	Kartenansicht mit geplanter Route	Papierprototyp	Aya Almallal	fertig
10. Fahrtangebote				
10.1	Fahrtangebot durch Fahrer	User Story	Melike Ünver	fertig
10.2	Annahme/Ablehnung durch Kunde	User Story	Melike Ünver	fertig
10.3	Angebotsansicht – Kunde	Papierprototyp	Sara Akgüvercin	fertig
11. Durchführung einer Fahrt (Simulation)				
11.1	Simulation starten, pausieren, fortsetzen	User Story	Rawan Almallal	fertig
11.2	Echtzeit-Marker-Bewegung und Synchronisation	User Story	Rawan Almallal	fertig
11.3	Simulations-Oberfläche mit Bedienelementen	Papierprototyp	Aya Almallal	fertig
12. Fahrthistorie				

12.1	Ansicht vergangener Fahrten	User Story	Rawan Almalla	offen
12.2	Tabelle mit Sortier- und Suchfunktion	Papierprototyp	Rawan Almalla	offen

13.	Systemarchitektur			
13.1	Komponentendiagramm	Komponentendiagramm	Bashar Herzallah	Fertig
13.2	Klassendiagramm	Klassendiagramm	Rias Safi	Fertig
13.3	Kommunikationsdiagramm	Kommunikationsdiagramm	Ibrahim Sayed	Fertig

User-Stories

Template:

User Story-ID	7.1
User Story-Beschreibung	Als Kunde möchte ich ein Geldkonto haben und darauf Geld laden können, um den Fahrer für die erfolgreich durchgeführten Fahrten bezahlen zu können.
Geschätzter Realisierungsaufwand	2
Priorität	hoch
Autor	Melike Ünver
Abhängigkeiten zu anderen User Stories	11, 7.2

User Story-ID	7.2
User Story-Beschreibung	Als Fahrer möchte ich ein Geldkonto haben, damit Kunden darüber meine Fahrten bezahlen können, um sicher für meine erbrachten Fahrdienstleistungen entlohnt zu werden.
Geschätzter Realisierungsaufwand	1
Priorität	hoch
Autor	Melike Ünver
Abhängigkeiten zu anderen User Stories	7.1

User Story-ID	8.1
User Story-Beschreibung	Als Fahrer möchte ich eine sortierbare Liste von Fahranfragen sehen, damit ich passende Fahrten schneller finden kann
Geschätzter Realisierungsaufwand	hoch
Priorität	mittel
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	8.2, 5.1, 5.2

User Story-ID	8.2
User Story-Beschreibung	Als Fahrer möchte ich meine Position angeben können, damit die Entfernung zum Startpunkt der Fahrten berechnet werden kann.
Geschätzter Realisierungsaufwand	Hoch
Priorität	hoch
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	8.1, 5.1, 5.2

User Story-ID	9.1
User Story-Beschreibung	Als <i>Kunde</i> möchte ich, dass beim Erstellen einer Fahranfrage automatisch eine Route vom Startpunkt zum Ziel berechnet wird, damit ich eine Vorschau der Strecke und die zugehörigen Informationen (z. B. Distanz, Dauer, Preis) erhalten
Geschätzter Realisierungsaufwand	4 Tage
Priorität	Hoch
Autor	Aya Almalla
Abhängigkeiten zu anderen User Stories	7.1, 8.1, 9.2

User Story-ID	9.2
User Story-Beschreibung	Als <i>Kunde</i> möchte ich, dass automatisch die Distanz, die geschätzte Dauer und der Preis der geplanten Route berechnet werden, damit ich vor der Bestätigung der Fahranfrage alle relevanten Informationen sehe.
Geschätzter Realisierungsaufwand	3
Priorität	Hoch
Autor	Aya Almalla
Abhängigkeiten zu anderen User Stories	7.1, 9.1

User Story-ID	10.1
User Story-Beschreibung	Als Fahrer möchte ich maximal eine beliebige Fahranfrage aus der Liste der verfügbaren Fahranfragen auswählen können und gegebenenfalls zurückziehen.
Geschätzter Realisierungsaufwand	3
Priorität	hoch
Autor	Melike Ünver
Abhängigkeiten zu anderen User Stories	8.1

User Story-ID	10.2
User Story-Beschreibung	Als Kunde möchte ich die Möglichkeit haben, Angebote seitens der Fahrer tabellarisch sortiert zu bekommen und diese anzunehmen oder abzulehnen.
Geschätzter Realisierungsaufwand	3
Priorität	hoch
Autor	Melike Ünver
Abhängigkeiten zu anderen User Stories	10.1

User Story-ID	11.2
User Story-Beschreibung	Als Kunde möchte ich während einer laufenden Fahrt die aktuelle Position des Fahrzeugs in Echtzeit auf der Karte verfolgen können, sodass ich genau weiß, wo sich mein Fahrer befindet und wann ich mein Ziel erreiche
Geschätzter Realisierungsaufwand	2-3 Tage
Priorität	Hoch
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	10.1, 10.2 und 11.

User Story-ID	11.1
User Story-Beschreibung	Als Kunde/Fahrer möchte ich die Simulation einer Fahrt starten, pausieren und fortsetzen können, sodass ich den Fortschritt der Fahrt kontrollieren und bei Bedarf unterbrechen kann
Geschätzter Realisierungsaufwand	3 Tage
Priorität	Hoch
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	9.1 , 9.2 und 10.2

User Story-ID	12.1
User Story-Beschreibung	Als Fahrer möchte ich eine Liste meiner abgeschlossenen Fahrten sehen können, sodass ich meine Einnahmen, gefahrenen Distanzen und erhaltenen Bewertungen nachverfolgen kann.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	Hoch
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	10.1, 10.2 und 11.1

User Story-ID	12.1
User Story-Beschreibung	Als Kunde möchte ich eine Liste meiner vergangenen Fahrten einsehen können, sodass ich nachvollziehen kann, wann ich wohin gefahren bin und wie viel ich bezahlt habe
Geschätzter Realisierungsaufwand	2 Tage
Priorität	Hoch
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	10.1, 10.2 und 11.1

Papierprototypen

7.3 View der Geldkonten



Auf der Mapseite soll im Header oben rechts ein Button stehen, wo der Betrag zu sehen ist und mit einem Plus zeichen Geld aufladbar ist. Die Bezahlung des Fahrers soll am Ende des erfolgreich abgeschlossenen Auftrags automatisch durchgeführt werden.

8.3 View der Fahranfragen

->Nach dem man Button Fahranfrage klickt erscheint unten die Tabelle der Verfügbaren Fahranfragen

SEP Drive

Startseite Fahranfragen Profil Abmelden

> Aktuellen Standort verwenden

Startadresse: Köln Hbf

Zieladresse: Gerichtsplatz Mülheim an der Ruhr

Geschätzte Distanz: 36.2 km
Geschätzte Fahrtzeit: 44 min

Route anzeigen Anfrage bestätigen

Dowic Wattenup Wang
Rochum Willet
Reeittack Ettken Unger Henfacke
Ratkemani Lethlack Lurgaar
Zeim Blasen

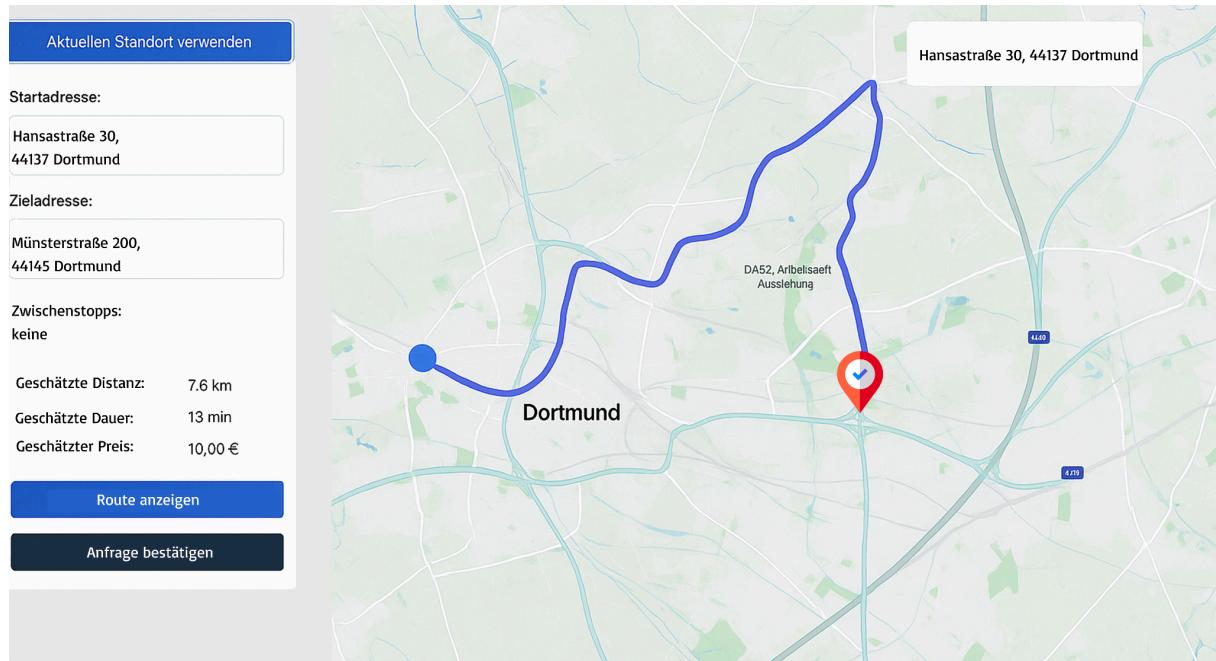
Verfügbare Fahranfragen

Suche... (ab-/aufsteigend)

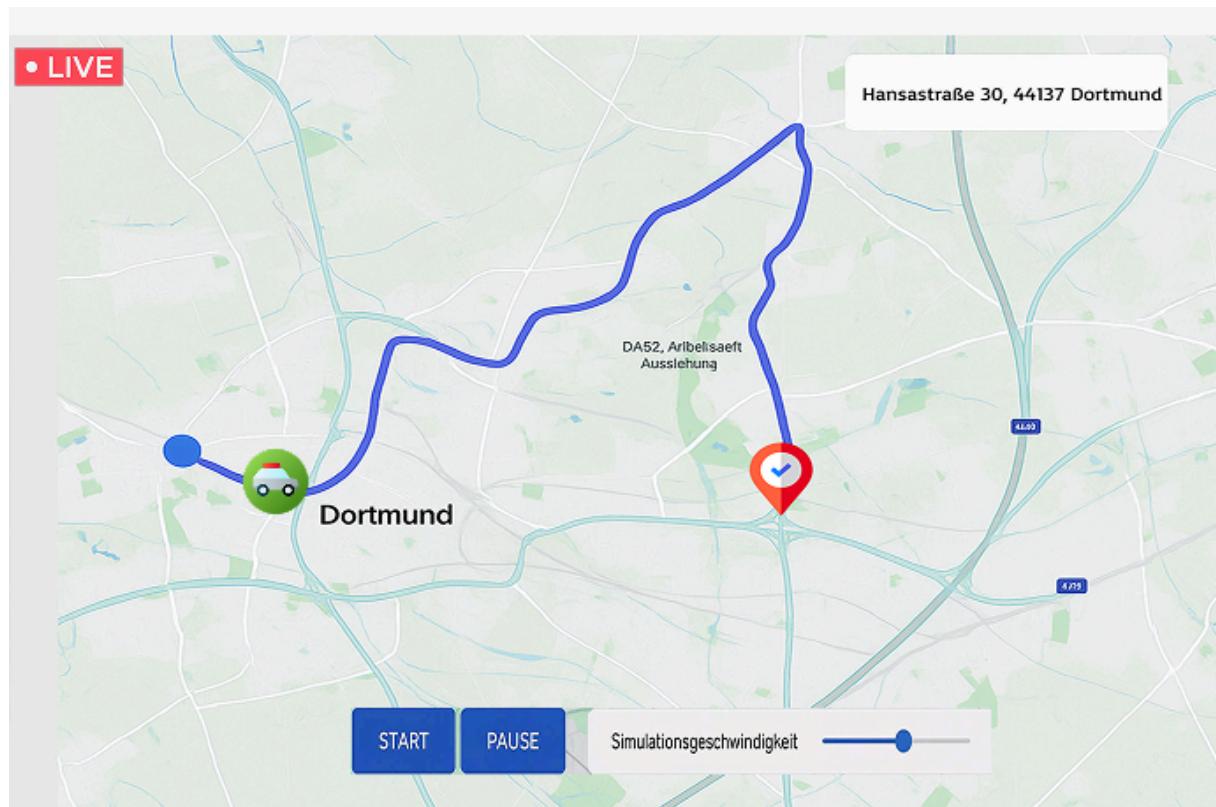
Fahrt-ID	Erstellt am	Entfernung (km)	Kundenname	Bewertung	Fahrzeugklasse
F123	2024-04-22 10:30	1.2	Max Mustermann	★★★★★	medium
F124	2024-04-22 00:15	3.4	Erika Müller	★★★★★	klein
F125	2024-04-23 14:30	0.8	Peter Schmitz	★★★★★	klein
F126	2024-04-23 05:45	2.3	Julla Wagner	★★★★★	medium

41

9.3 Kartenansicht mit geplanter Route



11.3 Simulations-Oberfläche mit Bedienelementen



10.3 Fahrangebote - Kunde

Nach dem man ein Button Fahrangebote anklickt erscheint die Seite

Fahrtangebot

Fahrt-ID	ingesamt durchgeführte Fahrten	Distanz	Kundenname	Kundenbewertung
8724	312	11.850 km	Maria Klein	★★★★★ 4,2
8742	287	9.420 km	Jens Müller	★★★★★ 4,9
8750	10	4.530 km	Sophie Lange	★★★★☆ 3,7
8761	100	6.366 km	Leo Frank	★★★★★ 5,0

Angebot gesendet

Angebot zurückziehen

OK

Benachrichtigung: Ihre Anfrage wurde angenommen

12.2 Fahrthistorie

SEP Drive

Startseite Fahrtragen Profile Abmelden

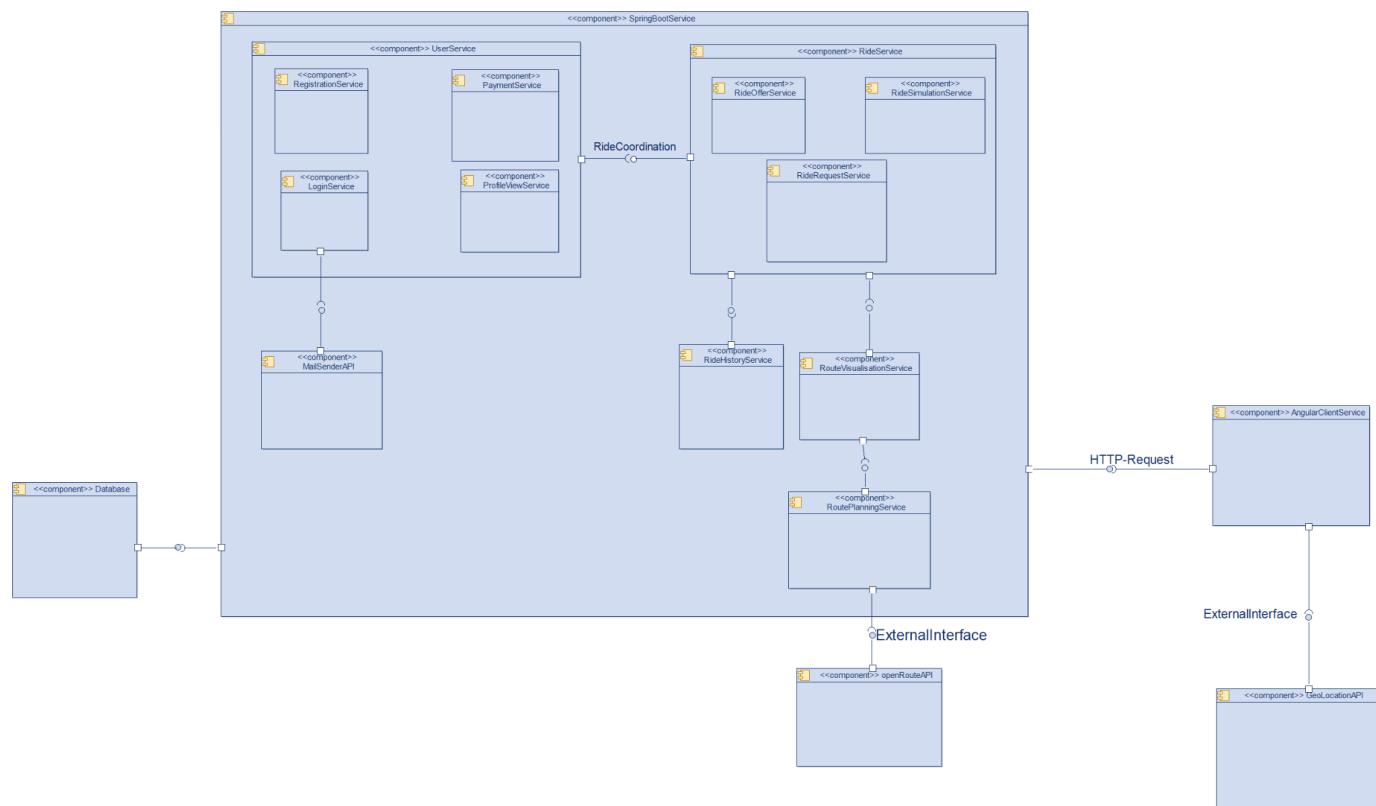
Fahrthistorie

Suche.....

Fahrt-ID	Abschlusszeit	Entfernung (km)	Dauer	Betrag (€)	Bewertung	Name	Benutzername
F1	20.05.2024 10:50	1,2	20 min	20€	★★★★★	Rawan Almalia	Rawan0
F2	13.04.2023 11:50	3,4	30 min	30€	★★★★★	Ahmad Ali	Ahmadali12
F3	11.01.2012 12:00	12,4	1 h 30 min	80€	★★★★★	Max Mustermann	Max00Muster
F4	23.02.2012 16:00	2,4	40 min	40€	★★★★★	Lisa Bergmann	Lisa12345
F5	27.01.2003 11:00	1,5	15 min	15€	★★★★★	Rahaf Rij	01rahaf12

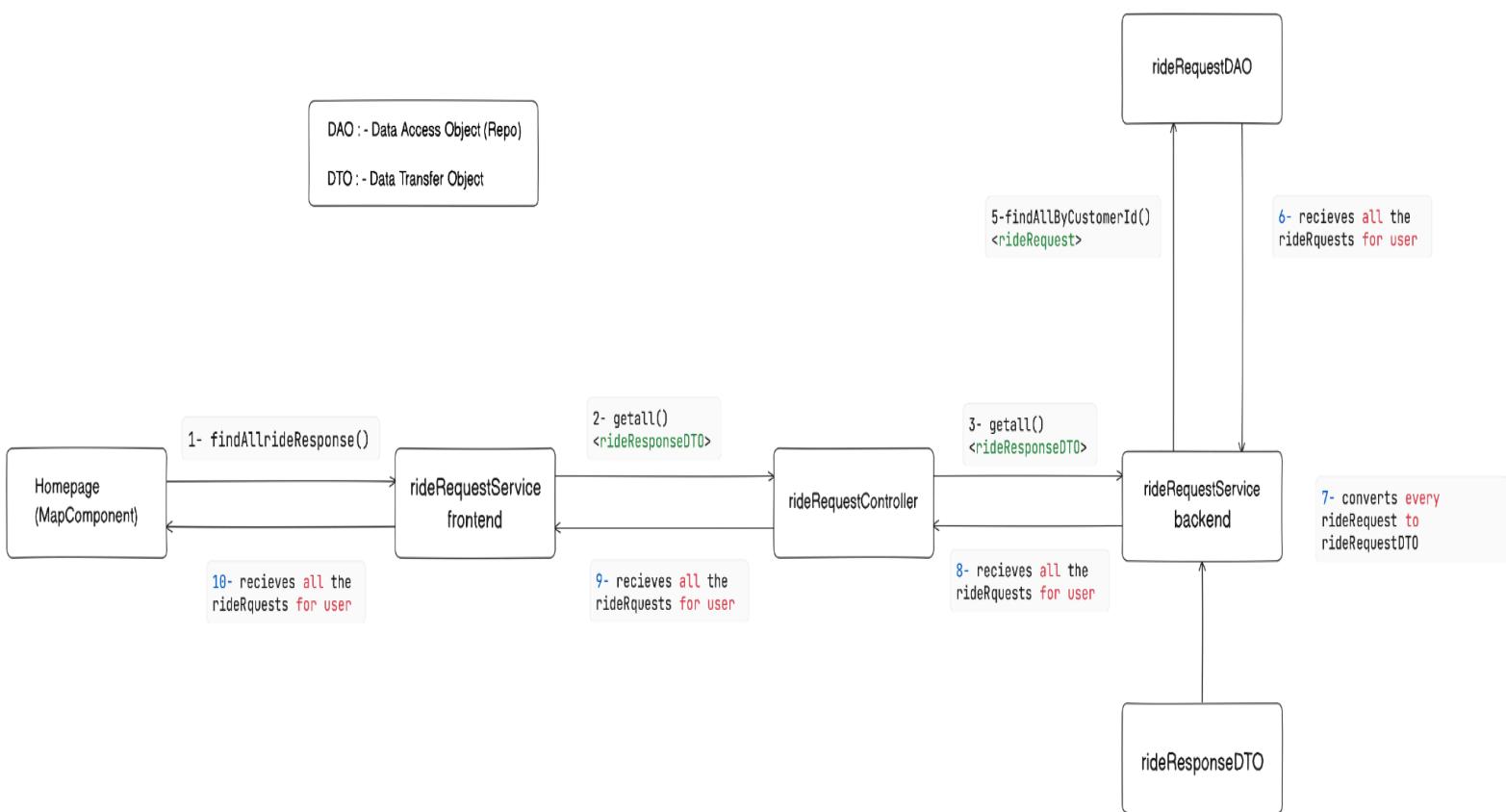
Strukturdiagramme (Komponenten- und Klassendiagramme)

Komponentendiagramm:

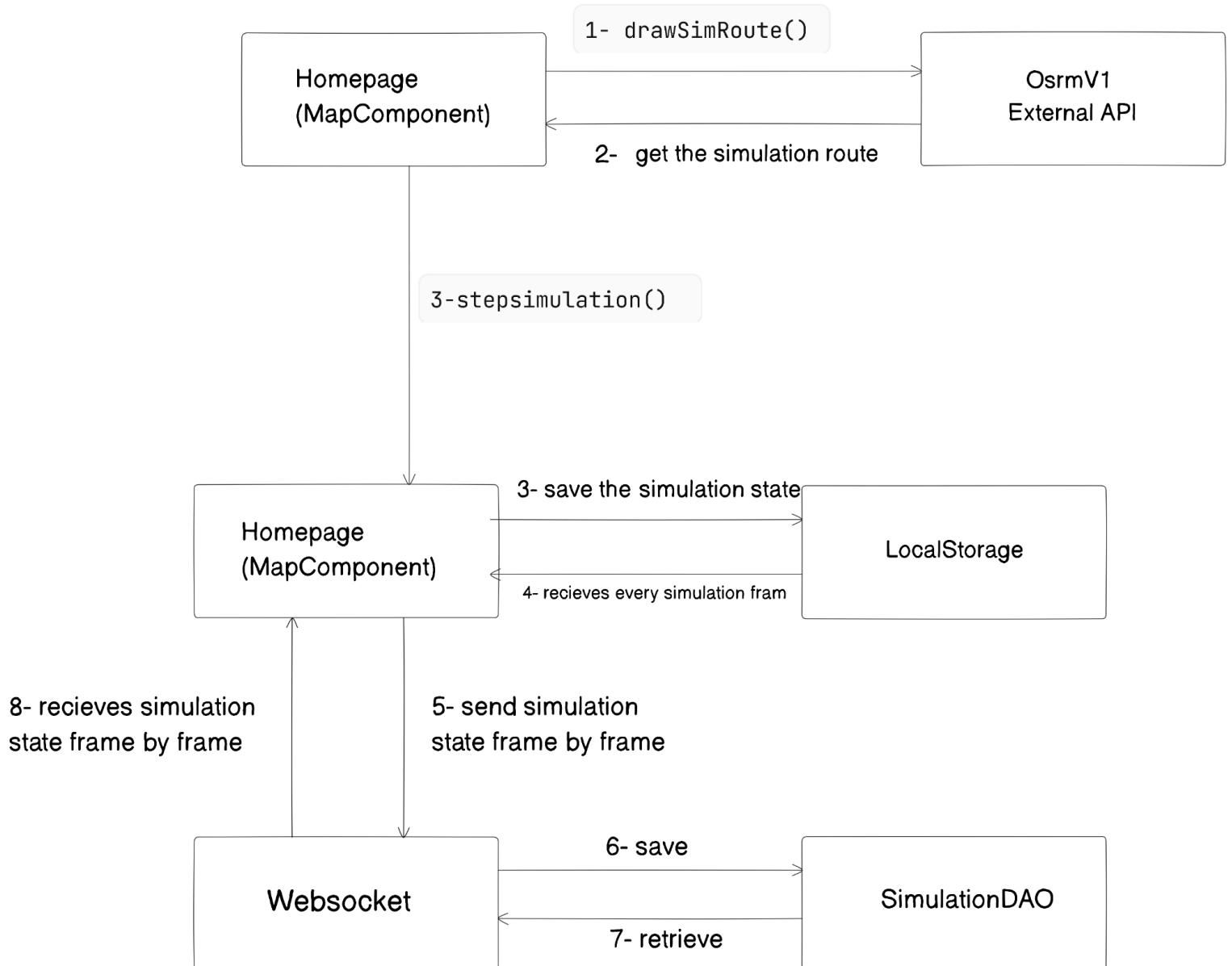


Verhaltensdiagramme (Kommunikationsdiagramme)

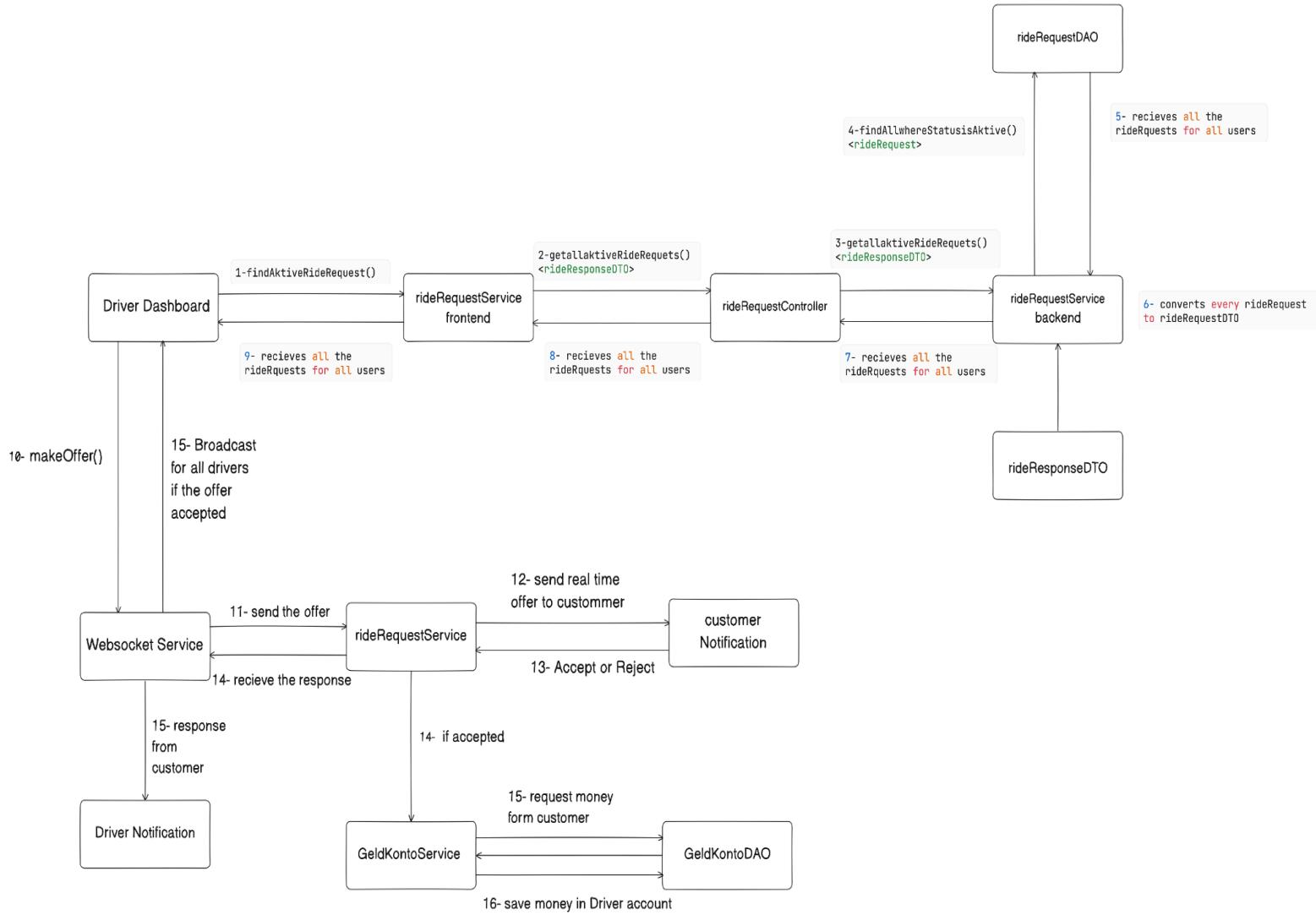
Fahrt History:



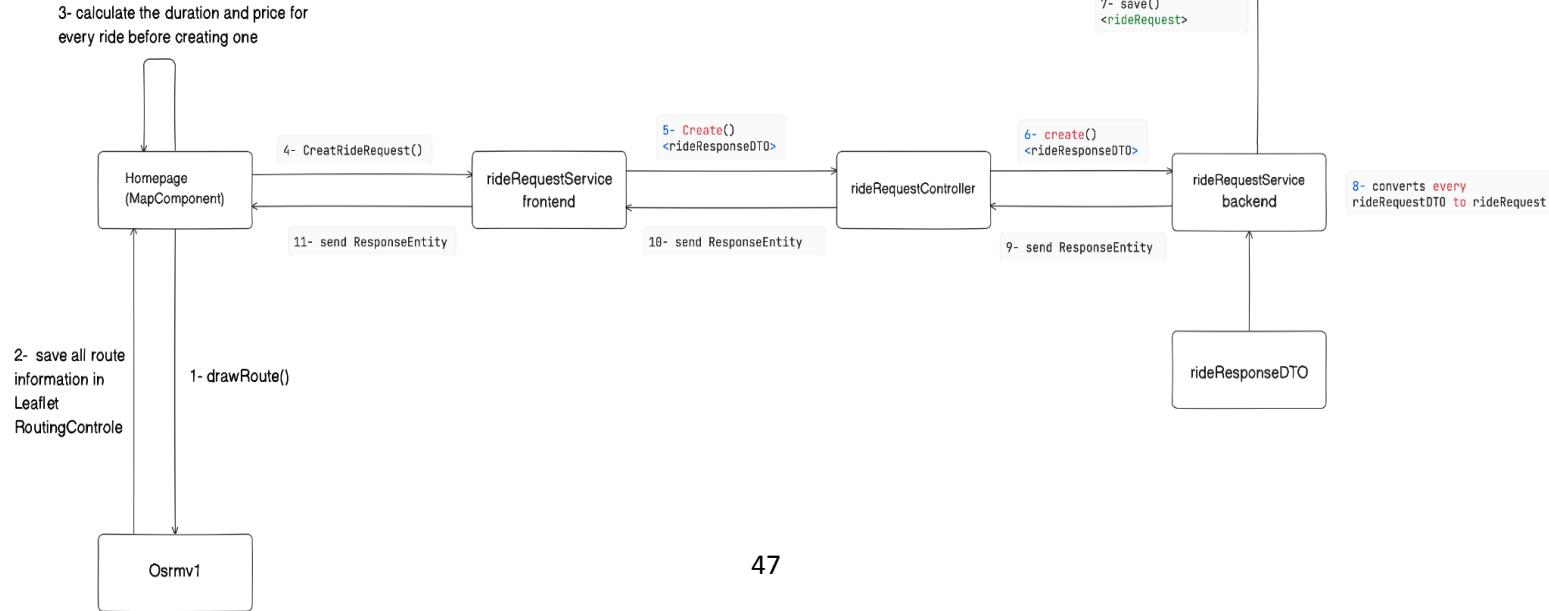
Simulation:



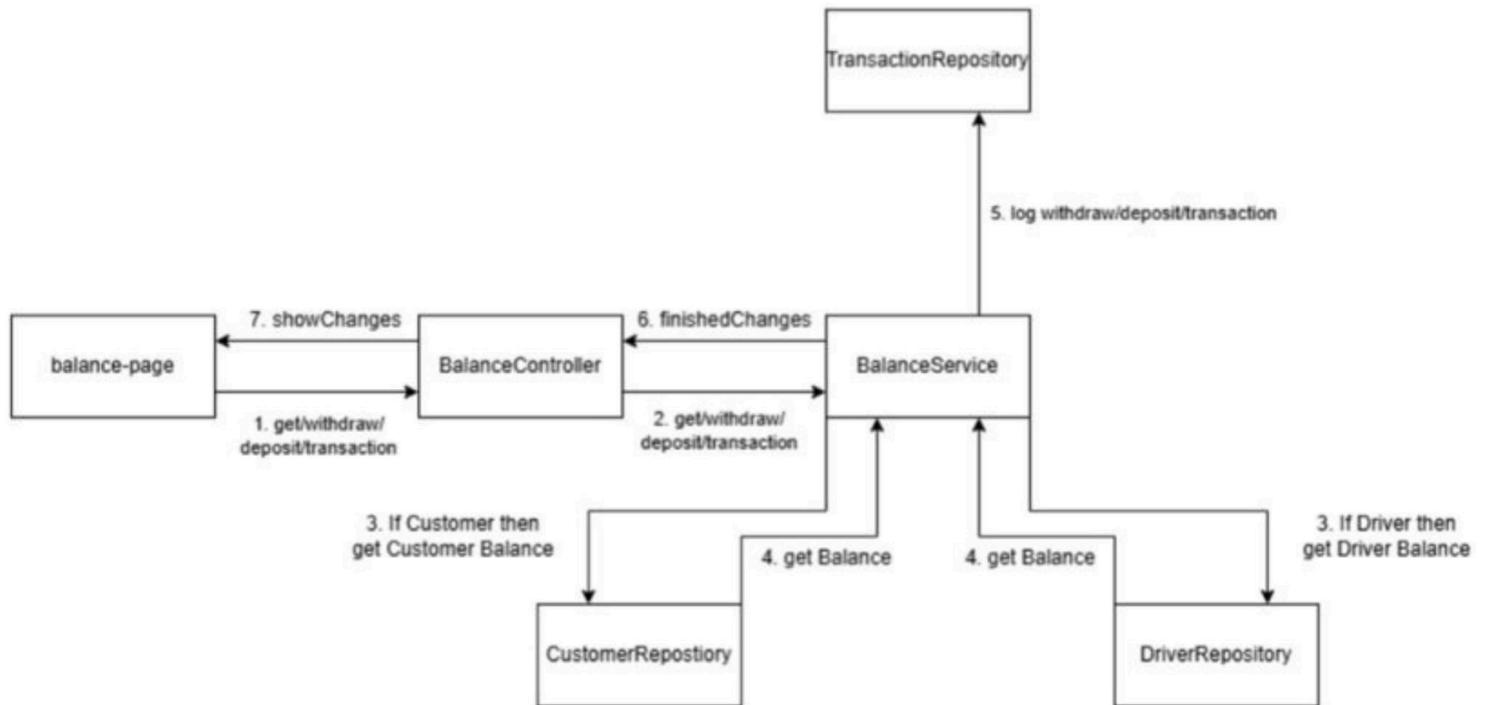
Fahrtangebot:



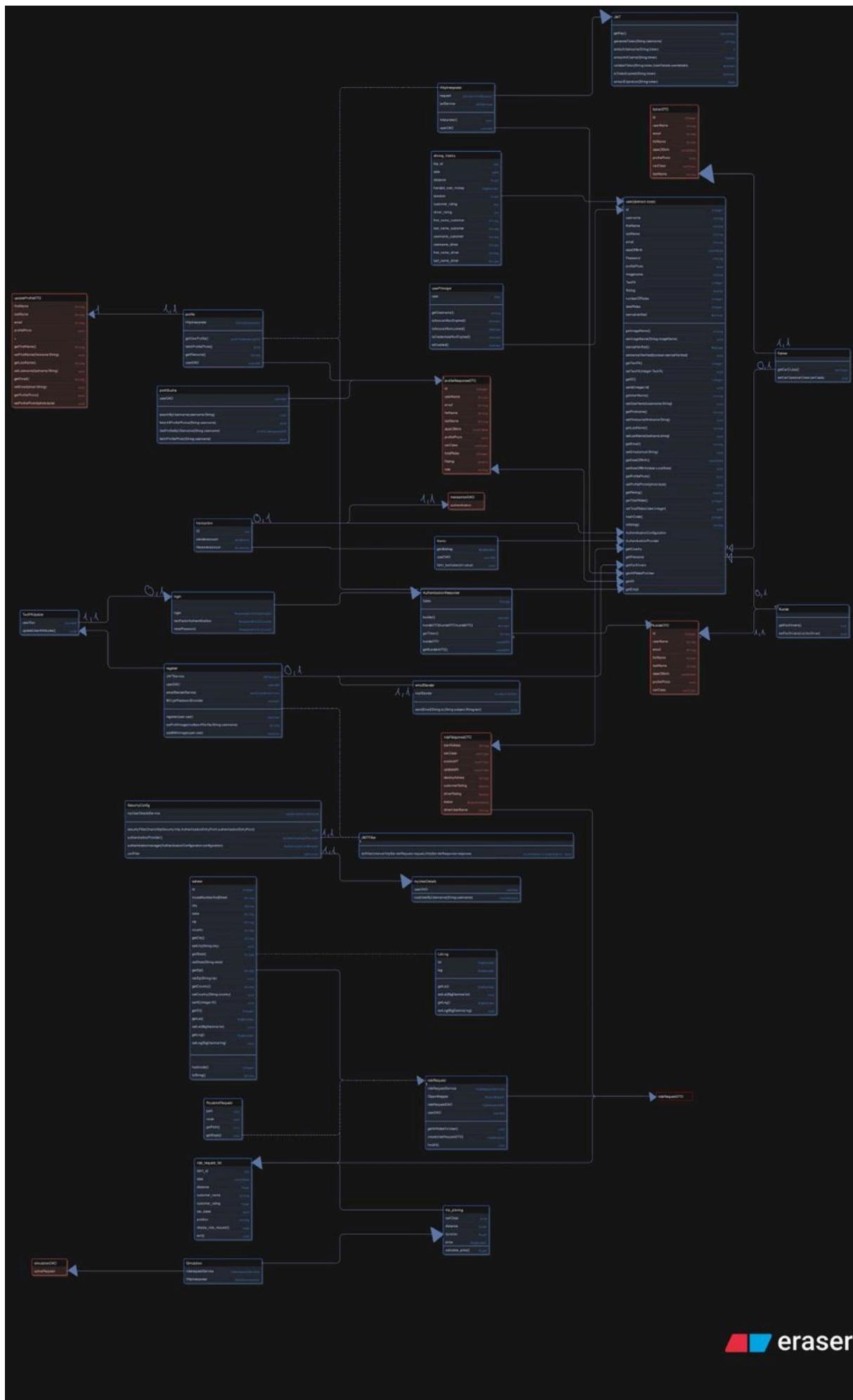
FahrtPlannung:



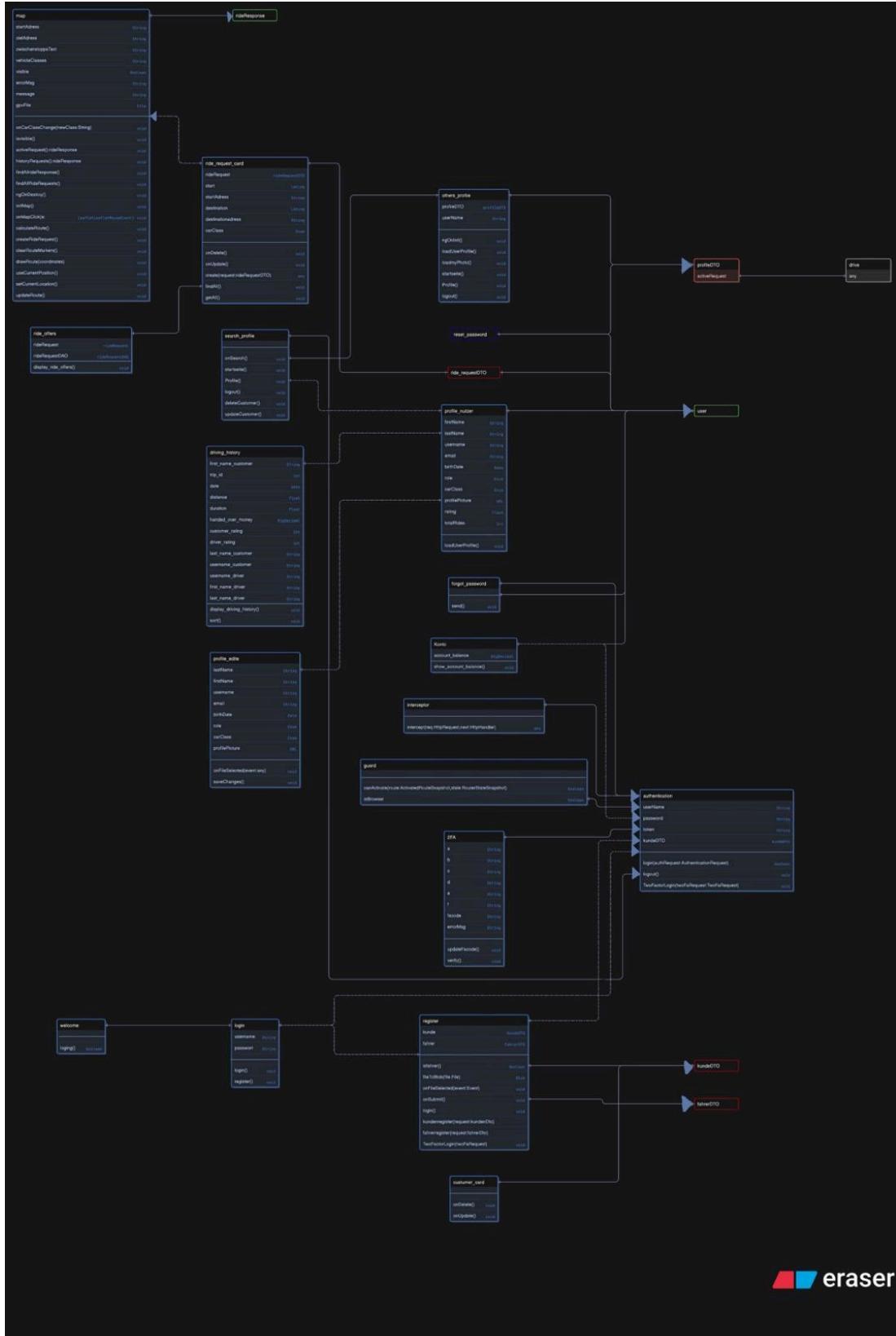
GeldKonto :



Class-Diagram Backend:



Class-Diagram Frontned:



Funktionalitätsplanung

Frontend :

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
6.	Geldkonto					
6.1	Geldbetrag anzeigen	Melike Ünver	4.1,4.2,8.6	7.1, 7.2,9.2,12.1	gruppeC/frontend/Angular/src/app/components/bank-account	fertig
6.2	Geldkonto aufladen	Melike Ünver	4.1,4.2,8.6	7.1,9.2,12.1	gruppeC/frontend/Angular/src/app/components/bank-account	fertig
7.	Liste verfügbarerer Fahranfragen					
7.1	Tabelle der Fahranfragen mit den angegebenen Spalten	Sara Akgüvercin	4.2., 4.3	8.2, 5.1, 5.2	gruppeC/frontend/Angular/src/app/components/driverdashboard (Zeile 1-792)	fertig
7.2	Funktion : Sortierung nach allen Spalten	Sara Akgüvercin	4.2, 4.3	8.1, 5.1, 5.2	gruppeC/frontend/Angular/src/app/components/driverdashboard (Zeile 1-792)	fertig
7.3	Kartenanzeige	Sara Akgüvercin	4.2, 4.3	8.1, 5.1, 5.2	gruppeC/frontend/Angular/src/app/components/driverdashboard (Zeile 1-792)	fertig
7.4	Filter/Suchleisten Funktion	Sara Akgüvercin	4.2, 4.3	8.1, 5.1, 5.2	gruppeC/frontend/Angular/src/app/components/driverdashboard (Zeile 1-792)	fertig
7.5	Funktion : Entfernungsberechnung	Ibrahim Sayed	4.2, 4.3	8.2, 5.1, 5.2	gruppeC/frontend/Angular/src/app/components/driverdashboard (Zeile 793-1056)	fertig

7.6	Standort angeben	Ibrahim Sayed	4.2, 4.3	8.2, 5.1, 5.2	gruppeC/frontend/Angular/src/app/components/driverdashboard (Zeile 793-1056)	fertig
8.	Fahrtplanung					
8.1	Gesamtdistanz automatisch berechnen	Aya Almallal	8.6	8.2, 9.2	gruppeC/frontend/Angular/src/app/components/map 881-904	fertig
8.2	Gesamtdauer automatisch berechnen	Aya Almallal	8.6	9.1, 9.2	gruppeC/frontend/Angular/src/app/components/map 881-904	fertig
8.3	Fahrtpreis berechnen (Formel: Distanz × Fahrzeugklassen-Faktor)	Aya Almallal	-	9.2, 7.1, 7.2	gruppeC/frontend/Angular/src/app/components/map 881-904	fertig
9	Fahrthistorie					
9.1	Fahrthistorie anzeigen (view)	Rawan Almallal	8.4, 8.5, 8.6	12.1, 12.2	gruppeC/frontend/Angular/src/app/components/map (map.component.ts Zeile 92 - 97 und Zeile ... (Methoden : get historyRequests() und map.component.HTML Zeile 221 -379)	fertig
9.2	Tabelle mit Sortier-/Suchoptionen	Rawan Almallal	9.1		gruppeC/frontend/Angular/src/app/components/map (map.component.ts Zeil 1078 - 1217 (Methoden onAscending itemChange()); ondescending itemChange() und onSearchCha	fertig

					<code>ngFor()</code> und map.component.html Zeile 221 - 288)	
9.3	ride-request.service.ts	Ibrahim Sayed			gruppeC/frontend\Angular\src\app\services\rideRequest	fertig
10.	Durchführung einer Fahrt (Simulation)	Ibrahim Sayed		8.1, 8.2, 8.3, 8.4, 8.5, 8.6	gruppeC/frontend/Angular/src/app/components/MapComponent gruppeC/frontend/Angular/src/app/model/notification - notificationperson - simulationstatmodel gruppeC/frontend/Angular/src/app/service/rereshService - WebsocketService	fertig
11.	Fahrtangebot					
11.1	Angebot auf Fahranfrage abgeben	Ibrahim Sayed	7.1, 7.2, 8.6	8.1	gruppeC/frontend/Angular/src/app/services/websocket	fertig
11.2	Angebot zurückziehen	Ibrahim Sayed	11.1	10.2	gruppeC/frontend/Angular/src/app/services/websocket	fertig
11.3	Fahrangebote filtern	Melike Ünver	7.4	10.2	gruppeC/frontend/Angular/src/app/components/fahrt-anfrage 250-382	fertig
11.4	Fahrangebote laden	Aya Almallal	11.3, 10.2	10.2	gruppeC/frontend/Angular/src/app/components/fahrt-anfrage	fertig

					178-248	
11.5	Sortierfunktion für Angebote	Aya Almallal	11.3, 10.2	10.2	gruppeC/frontend/Angular/src/app/components/fahrt-anfrage 418-441	fertig
11.6	Angebot ablehnen/annehmen	Melike Ünver	11.3	10.2	gruppeC/frontend/Angular/src/app/components/fahrt-anfrage 448-480	fertig
11.7	Konkurrierende Angebote nach Annahme automatisch ablehnen	Ibrahim Sayed	11.4	10.1	gruppeC/frontend/Angular/src/app/services/websocket	fertig

Backend :

ID	Funktionalitäten	Verantwortlicher	Abhängigkeiten	Verknüpfte User-Stroys	Quellcodereferenz	Status
11.	Geldkonten(Wallets)	Bashar Herzallah	6.	7.1, 7.2,9.2,12.1	gruppeC/backend/src/main/java/com.example.SEPDrive/controller/geldKontoController. gruppeC/backend/src/main/java/com.example.SEPDrive/Service/geldKontoService. gruppeC/backend/src/main/java/com.example.SEPDrive/model/geldKonto.	Fertig

ID	Funktionalitäten	Verantwortlich er	Abhängige iten	Verknüpfte User-Stroys	QuellcodeReferenz	Status
11.1	Transactions	Bashar Herzallah	6.1 10. 11.	7.1, 7.2,9.2,12.1	gruppeC/ba ckend/src/ main/java/c om.example.SEPDrive/ model/Transactions. gruppeC/ba ckend/src/ main/java/c om.example.SEPDrive/ repository/ transaction DAO.	Fertig
12.	Fahrtsimulation	Ibrahim Sayed	10.	8.1, 8.2, 8.3, 8.4, 8.5, 8.6	gruppeC/ba ckend/src/ main/java/c om.example.SEPDrive/ service/sim ulationStat e gruppeC/ba ckend/src/ main/java/c om.example.SEPDrive/ services/si mulationDA O. gruppeC/ba ckend/src/ main/java/c om.example.SEPDrive/ controller/s imulationC ontroller. gruppeC/ba ckend/src/ main/java/c om.example.SEPDrive/ controller/s imulationU	Fertig

ID	Funktionalitäten	Verantwortlich er	Abhängige iten	Verknüpfte User-Stroys	QuellcodeReferenz	Status
					pdatePaylo ad.	
13.	Fahrtangebote					
13.1	POST, GET, PULL von Fahrangeboten den Rollen entsprechend	Rias Safi	11,8,7	8.3, 10.3	gruppeC/ba ckend/src/ main/java/s com.example.SEPDrive /rideReque stController	Fertig
13.2	Rollenspezifische Rechte (ablehnen/löschen von Fahrangeboten)	Rias Safi	11,8,7	8.3, 10.3	gruppeC/ba ckend/src/ main/java/s com.example.SEPDrive /RideReque stService Zeile486-82 0 zeile 294-380 zeile 56-172	Fertig
14.	Fahrtabschluss- und Bewertung					
14.2	Bewertung	Rias Safi	10,12		gruppeC/ba ckend/src/ main/java/s com.example.SEPDrive /service/rid eRequestSe rvice Zeile 75-79	Fertig
15.	Fahrthistorie	Ibrahim Sayed	9	12.1, 12.2	gruppeC/ba ckend/src/ main/java/c om.example.SEPDrive/ service/ride RequestSer vice	Fertig

ID	Funktionalitäten	Verantwortlich er	Abhängige iten	Verknüpfte User-Strogs	QuellcodeReferenz	Status
16.	Websocket	Ibrahim Sayed	6,7,10 11, 12	8.1, 8.2, 8.3, 8.4, 8.5, 8.6, 5.1,5.2	gruppeC/ba ckend/src/C onfig/webS ocketConfigur ation. /webMvcC onfig. gruppeC/ba ckend/src/ main/java/c om.exampl e.SEPDrive/ Service/rid eRequestSe rvice.	Fertig
17.	NotificationSyste m	Ibrahim Sayed	6,7,10 11, 12	8.1, 8.2, 8.3, 8.4, 8.5, 8.6, 5.1,5.2	gruppeC/ba ckend/src/ main/java/c om.exampl e.SEPDrive/ Service. gruppeC/ba ckend/src/ main/java/c om.exampl e.SEPDrive/ Controller/ notification Controller- Notification DTO- Notification Message gruppeC/ba ckend/src/ main/java/c om.exampl e.SEPDrive/ model/noti cationStat us. gruppeC/ba ckend/src/ main/java/c om.exampl e.SEPDrive/ notification	Fertig

ID	Funktionalitäten	Verantwortlich er	Abhängige iten	Verknüpfte User-Stroys	QuellcodeReferenz	Status
					. gruppeC/ba ckend/src/ main/java/c om.exampl e.SEPDrive/ nofitication DAO	

Modultests

Modultests sind Komponententests. Diese werden in der Softwareentwicklung angewendet, um die funktionalen Einzelteile (Units) von Computerprogrammen zu testen.

ID	Getestete Funktionalität	Quellcode Referenz	Status
1	Das ganze GeldKontoService.Class mit dem benötigte AuszahlungMethod.(SendMony).	gruppeC/backend/src/test/java/com.example.SEPDrive/service/geldKontoServiceTest	Fertig

Systemtests

Datum	14.06.2025		
Tester	Aya Almalla		
SW-Version	V 0.2.0		
Vorbedin-gung(en)	Kunde ist eingeloggt und hat ein aktives SEP-Drive-Konto. Fahrer ist ebenfalls eingeloggt.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde lädt 50 € auf das Konto	Konto zeigt 50 € mehr Guthaben	✓
2	Kunde zahlt eine Fahrt (z. B. 10 €)	Betrag wird vom Kundenkonto abgezogen	✓
3	Fahrt wird abgeschlossen	Betrag wird dem Fahrerkonto gutgeschrieben	✓
Nachbe-dingung(en)	Kundenkonto -10 € (Rest: 40 €), Fahrerkonto +10 €		
Testurteil	Bestanden		

Datum	16.06.2025		
Tester	Sara Akgüvercin		
SW-Version	V 0.2.0		
Vorbedin-gung(en)	Fahrer ist im System eingeloggt und es existieren mindestens 5 offene Fahranfragen mit unterschiedlichen Attributen.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Fahrer ruft die Ansicht „Dashboard“ auf.	Tabelle mit Fahranfragen wird angezeigt, inkl. Spalten.	✓
2	Fahrer gibt eine Adresse manuell über die Suchleiste ein.	System berechnet Entfernung zum Startpunkt und zeigt sie an.	✓
3	Fahrer klickt auf andere Spaltenköpfe (z. B. Kundenbewertung, Fahrzeugklasse) zur Sortierung.	Tabelle wird entsprechend der gewählten Spalte sortiert.	✓
Nachbe-dingung(en)	Keine Systemfehler oder falsche Entfernungs berechnungen.		
Testurteil	bestanden		

Datum	16.06.2025		
Tester	Aya Almallal		
SW-Version	V 0.2.0		
Vorbedingung(en)	Kunde ist eingeloggt und befindet sich auf der Map Seite. Geodaten-Dienste (openrouteservice) sind aktiv		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde gibt Start- und Zieladresse ein	Adresse wird erkannt und Kartenausschnitt angepasst	✓
2	Kunde wählt Fahrzeugklasse „Medium“	Faktor 2,0 €/km wird für Preisberechnung verwendet	✓
3	Kunde überprüft geplante Route auf Karte	Route wird zwischen Start und Ziel dargestellt	✓
4	Kunde prüft berechnete Gesamtdistanz und Dauer	Werte werden korrekt (basierend auf Routingdaten) angezeigt	✓
5	Kunde prüft angezeigten Preis (z. B. 12,4 km → 24,80 €)	Preis ist korrekt nach Formel: Distanz × Fahrzeugklassen-Fakto	✓
6	Kunde speichert die Fahranfrage ab	Anfrage wird im System gespeichert und erscheint in der Fahranfrageliste für Fahrer	✓
Nachbedingung(en)	Neue Fahranfrage inkl. Route, Distanz, Dauer und Preis ist im System gespeichert und sichtbar für Fahrer.		
Testurteil	Bestanden		

Zyklus III Spezifikationsplanung

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
14	Chatfunktion zwischen Fahrer und Kunde			
14.1	Echtzeit-Übertragung von Nachrichten	User Story	Sara Akgüvercin	fertig
14.2	Nachricht kann gelöscht/bearbeitet werden, solange sie ungelesen ist	User Story	Sara Akgüvercin	fertig
14.3	Chat-Oberfläche Design	Prototyp	Sara Akgüvercin	fertig
15	Fahrten mit mehreren Zwischenstopps			
15.1	Kunde kann zusätzliche Wegpunkte festlegen und Reihenfolge der Wegpunkte wird berücksichtigt	User Story	Aya Almalla	fertig
15.2	UI für Zwischenstopp Eingabe	Prototyp	Aya Almalla	fertig
16	Grafische Statistiken für Fahrer			
16.1	Anzeige von Einnahmen, Distanz, Zeit und Bewertung	User Story	Rawan Almalla	fertig
16.2	Auswahl zwischen täglicher und monatlicher Ansicht	User Story	Rawan Almalla	fertig
16.3	Prototyp für Statistik-Dashboard	Prototyp	Rawan Almalla	fertig
17	Live-Änderungen während der Fahrt			
17.1	Kunde kann Ziel/Wegpunkte während Fahrt ändern und System berechnet Route bei Änderung neu	User Story	Rawan Almalla	fertig
17.2	UI zur Bearbeitung der Route während der Simulation	Prototyp	Sara Akgüvercin	fertig
18	Fahrer-Leaderboard			
18.1	Anzeige aller Fahrer mit Leistungsdaten	User Story	Aya Almalla	fertig
18.2	Sortier- und Filterfunktion der Tabelle	User Story	Aya Almalla	fertig
18.3	Leaderboard Ansicht im Frontend	Prototyp	Aya Almalla	fertig

19	Systemarchitektur		
----	-------------------	--	--

19.1	Komponentendiagramm	Komponentendiagramm	Bashar Herzallah	Fertig
19.2	Klassendiagramm	Klassendiagramm	Rias Safi	Fertig
19.3	Kommunikationsdiagramm	Kommunikationsdiagramm	Ibrahim Sayed	Fertig

User-Stories

Template:

User Story-ID	14.1
User Story-Beschreibung	Als Kunde oder Fahrer möchte ich in der App Nachrichten in Echtzeit senden und empfangen können, damit eine direkte und schnelle Kommunikation zwischen beiden Parteien möglich ist.
Geschätzter Realisierungsaufwand	3
Priorität	hoch
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	5.1, 5.2, , 10.2, 11.1, 11.2

User Story-ID	14.2
User Story-Beschreibung	Als Nutzer möchte ich Nachrichten bearbeiten oder löschen können, solange sie vom Empfänger noch nicht gelesen wurden, damit ich Tippfehler korrigieren oder ungewollte Inhalte zurücknehmen kann.
Geschätzter Realisierungsaufwand	3
Priorität	mittel
Autor	Sara Akgüvercin
Abhängigkeiten zu anderen User Stories	14.1

User Story-ID	15.1
User Story-Beschreibung	Als Kunde möchte ich während der Fahranfrage zusätzliche Wegpunkte zwischen Start und Ziel festlegen können, damit meine Route individuell angepasst wird und die Reihenfolge der Wegpunkte berücksichtigt wird.
Geschätzter Realisierungsaufwand	4 Tage
Priorität	Hoch
Autor	Aya Almalla
Abhängigkeiten zu anderen User Stories	-

User Story-ID	16.1
User Story-Beschreibung	Als Fahrer möchte ich meine Einnahmen, gefahrene Distanz, Fahrzeit und durchschnittliche Bewertung in einem Dashboard visualisiert sehen, damit ich meine Leistung nachvollziehen und analysieren kann.
Geschätzter Realisierungsaufwand	5 Tage
Priorität	Hoch
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	12.1

User Story-ID	16.2
User Story-Beschreibung	Als Fahrer möchte ich zwischen einer täglichen und einer monatlichen Ansicht meiner Leistungsdaten wechseln können, damit ich sowohl detaillierte Tagesdaten als auch Trends auf Monatsebene erkennen kann.
Geschätzter Realisierungsaufwand	3 Tagen
Priorität	Mittel
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	16.1

User Story-ID	17.1
User Story-Beschreibung	Als Kunde möchte ich während einer laufenden Fahrt mein Ziel oder Zwischenziele ändern können, sodass das System die Route automatisch neu berechnet und mich weiterhin korrekt navigiert.
Geschätzter Realisierungsaufwand	4 Tage
Priorität	Hoch
Autor	Rawan Almallal
Abhängigkeiten zu anderen User Stories	9.1, 11.1

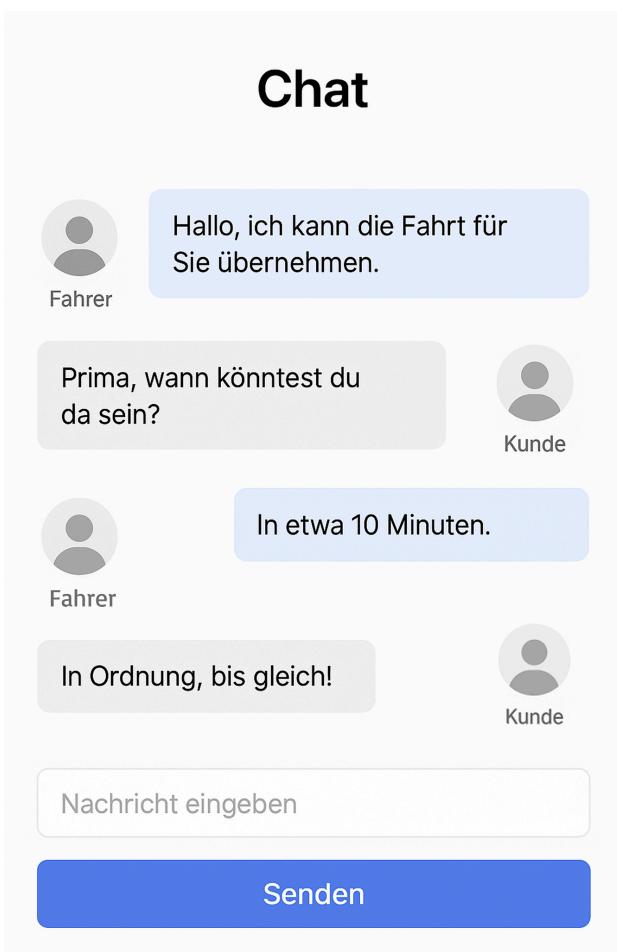
User Story-ID	18.1
User Story-Beschreibung	Als Benutzer möchte ich eine Tabelle sehen, in der alle Fahrer mit ihren Leistungsdaten (Benutzername, Name, Distanz, Bewertung, Fahrzeit, Fahrtenanzahl, Einnahmen) angezeigt werden, damit ich die Fahrer vergleichen kann.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	Mittel
Autor	Aya Almallal
Abhängigkeiten zu anderen User Stories	18.2

User Story-ID	18.2
----------------------	------

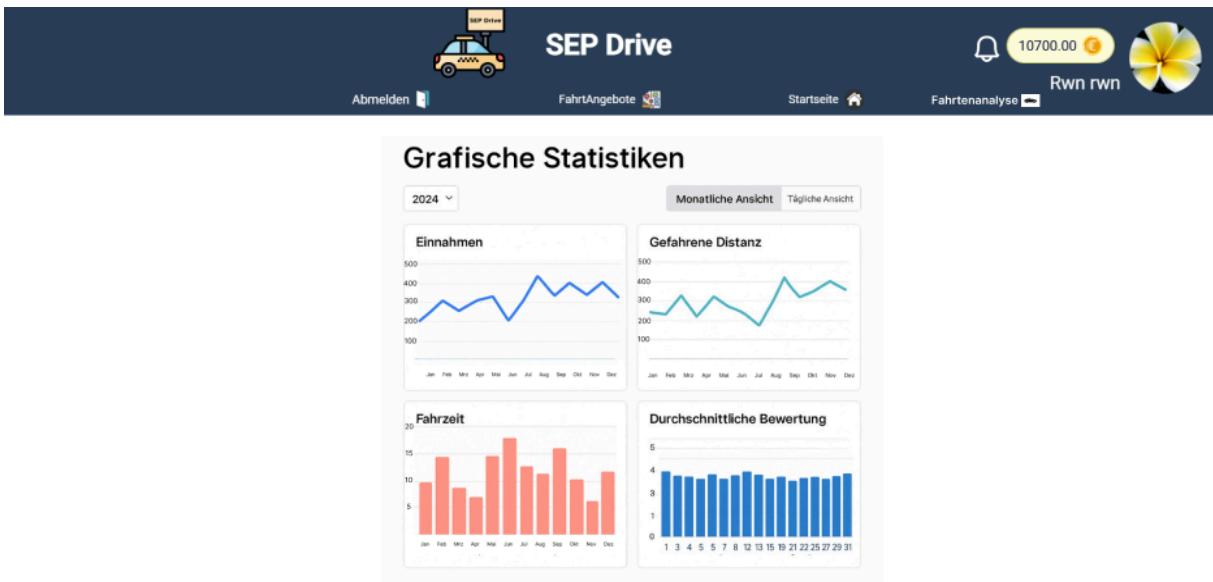
User Story-Beschreibung	Als Benutzer möchte ich die Fahrertabelle nach einzelnen Spalten sortieren und gezielt nach Fahrernamen filtern können, um bestimmte Informationen schneller zu finden und besser vergleichen zu können.
Geschätzter Realisierungsaufwand	1 Tag
Priorität	Mittel
Autor	Aya Almallal
Abhängigkeiten zu anderen User Stories	18.1

Papierprototypen

Chat-Funktion



Grafische Statistiken



UI zur Bearbeitung der Route während der Simulation

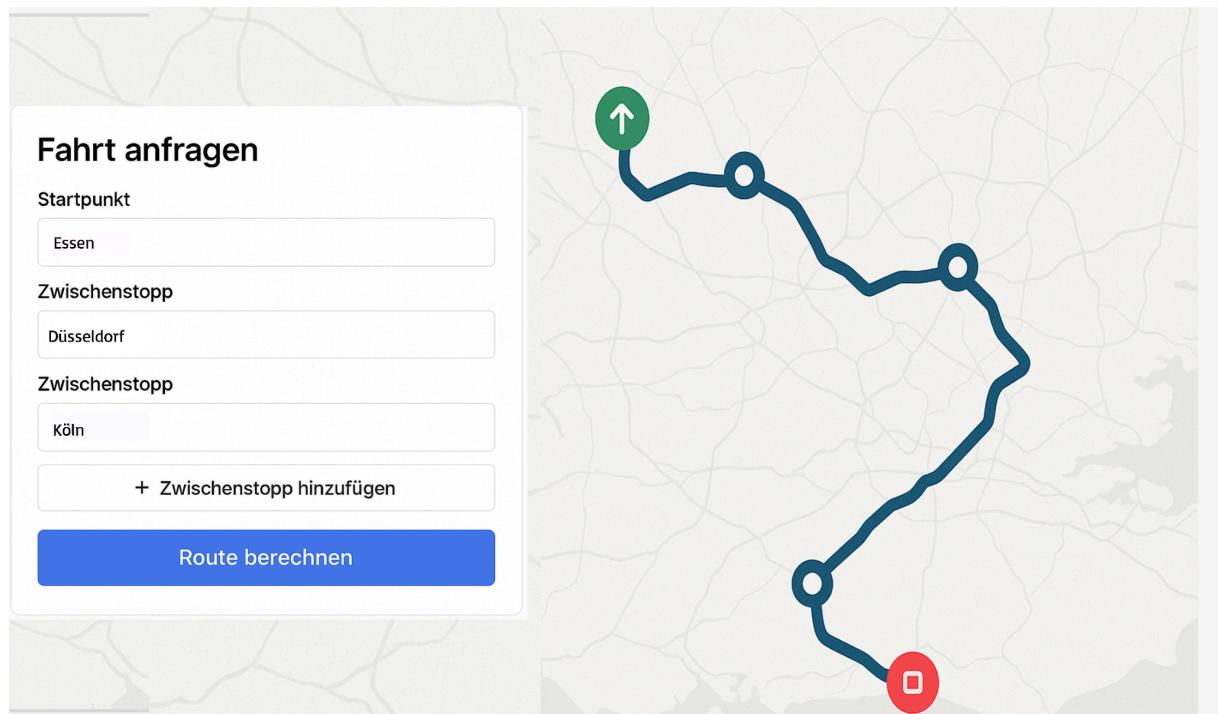


Fahrer Leaderboard

Driver Leaderboard

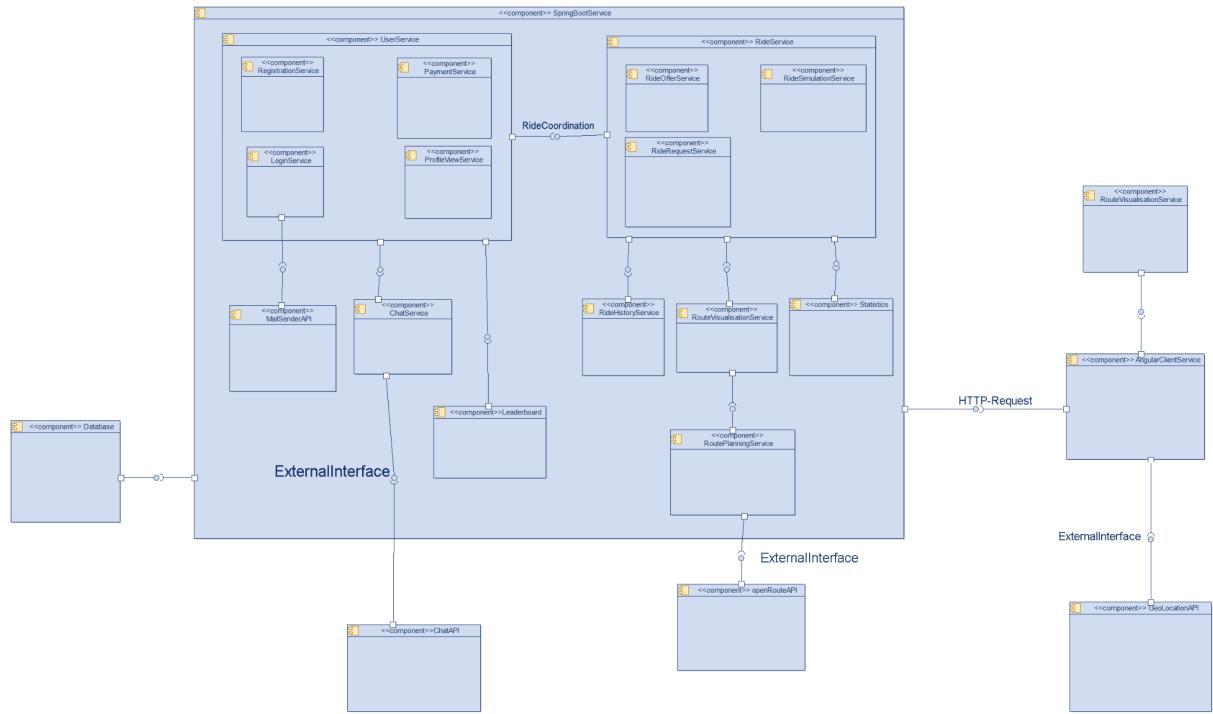
Driver Leaderboard						
Username	Name	Total Distance	Avg. Rating	Total Time	Ride Count	Earnings (€)
jdoe	John Doe	20.008 km	⭐ 4.9	285 h	30	€18,645
msmith	Michael Smith	19.275 km	⭐ 4.8	276 h	8	€17,970
djohnson	David Johnson	16.443 km	⭐ 4.9	240 h	30	€15,322.50
ewilliams	Emilly Williams	15.897 km	⭐ 4.7	205 h	20	€14,812.50
bthomas	Benjamin Thomas	15.211 km	⭐ 4.7	205 h	10	€14,175
swilson	Sophia Wilson	14.357 km	⭐ 4.6	198 h	8	€13,380
cbrown	Christopher Brown	14.106 km	⭐ 4.5	182 h	1	€13,147.50
amadison	Ava Madison	13.728 km	⭐ 4.4	175 h	0	€12,795

Fahrten mit mehreren Zwischenstopps

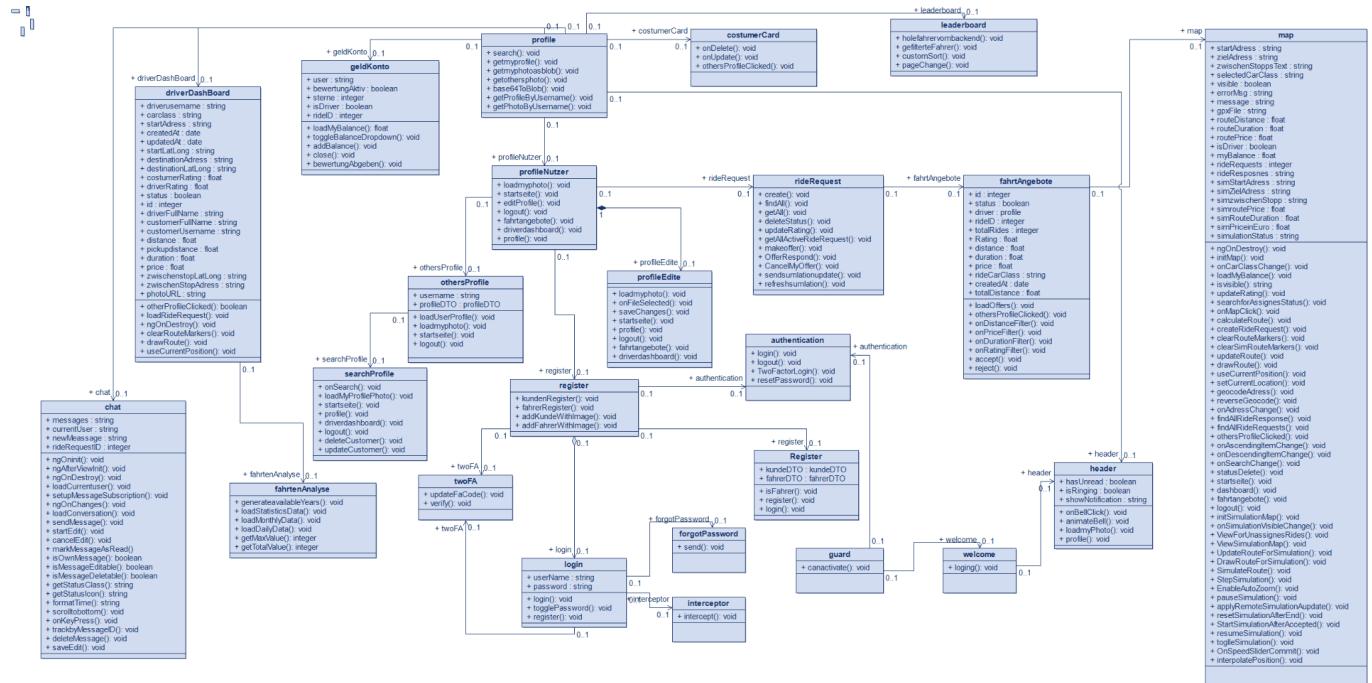


→ Zu allen Prototypen gelangt man, indem man die jeweiligen Buttons an klickt, die zu der jeweiligen Seite navigieren im header

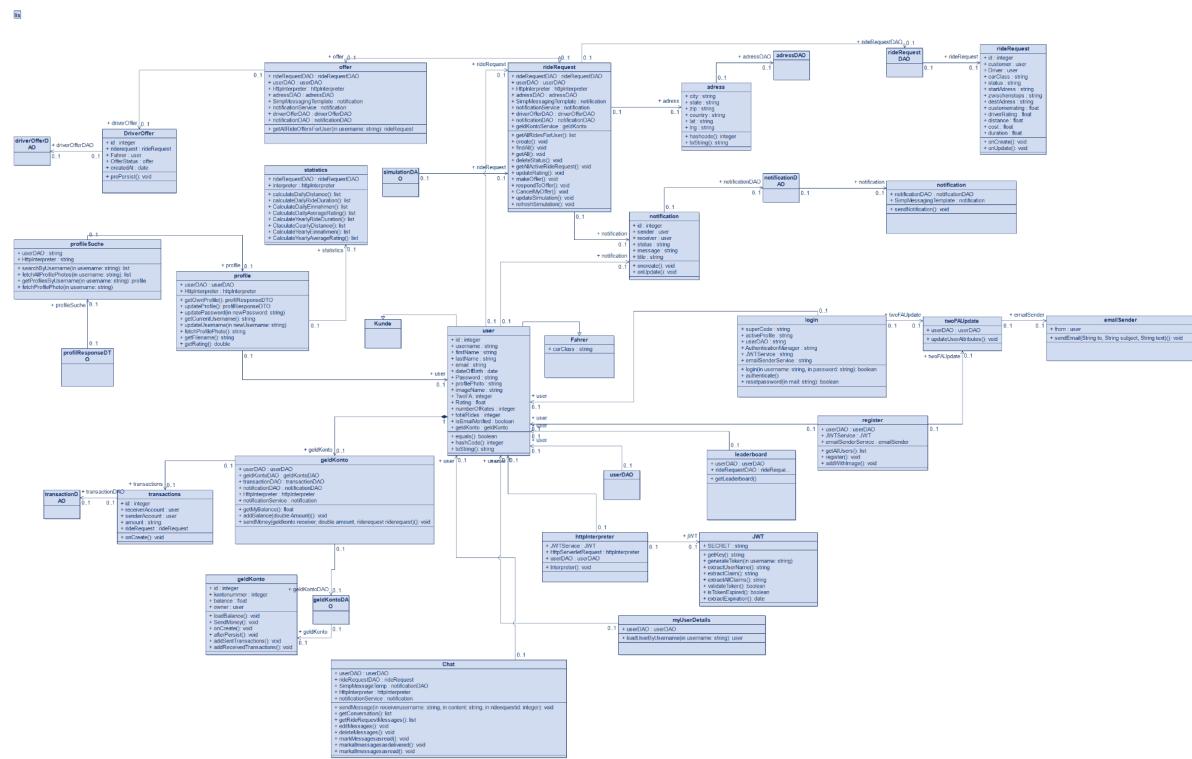
Strukturdiagramme (Komponenten- und Klassendiagramme)



Frontend Klassendiagramm:

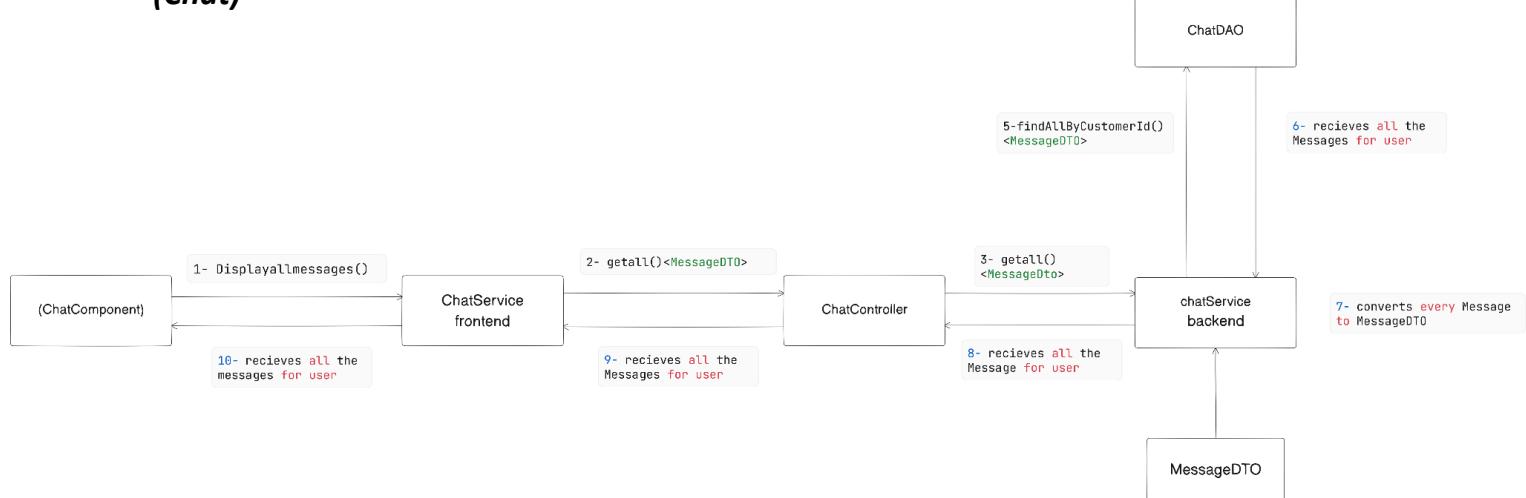


Backend Klassendiagramm:

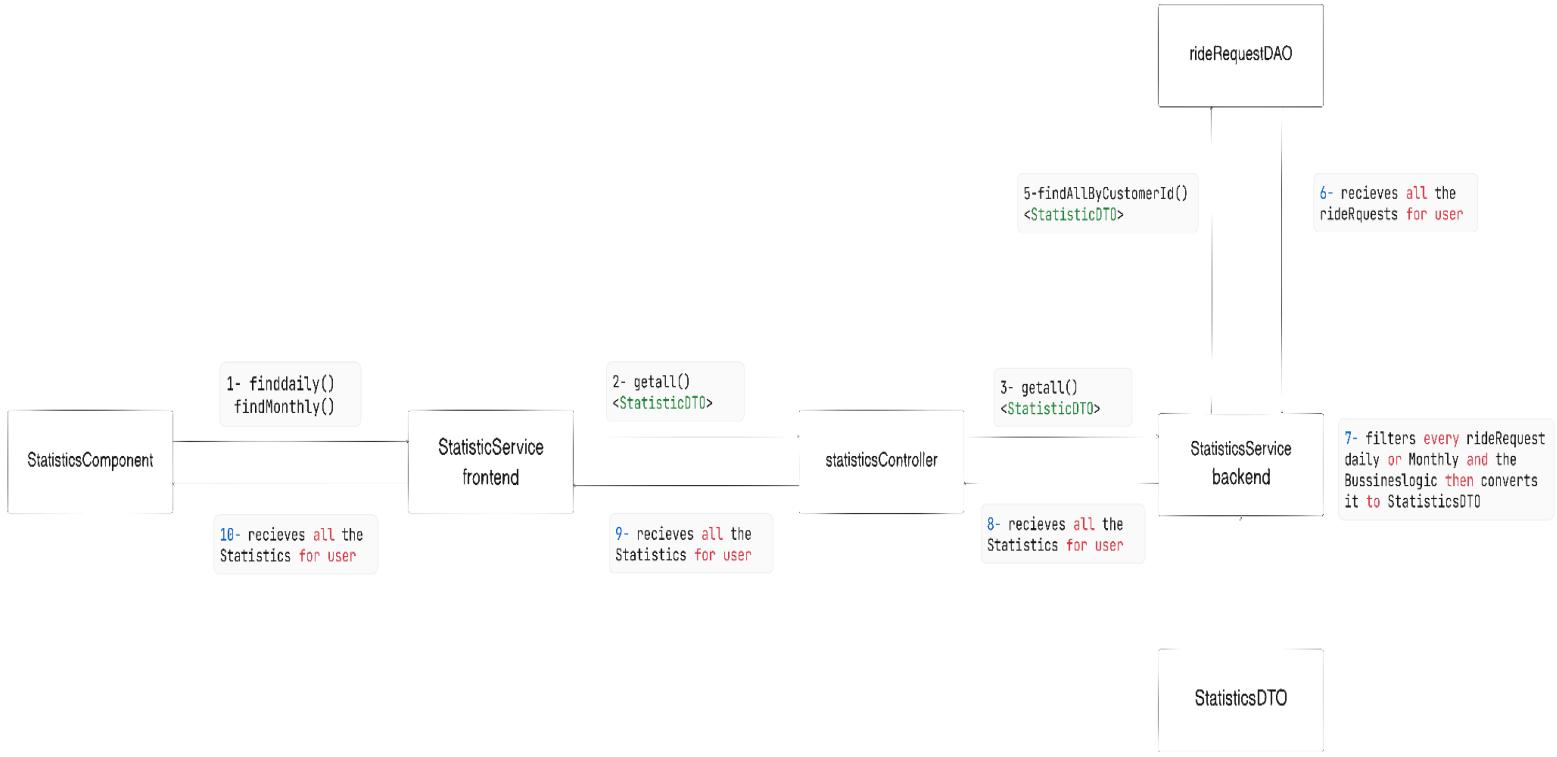


Verhaltensdiagramme (Kommunikationsdiagramme)

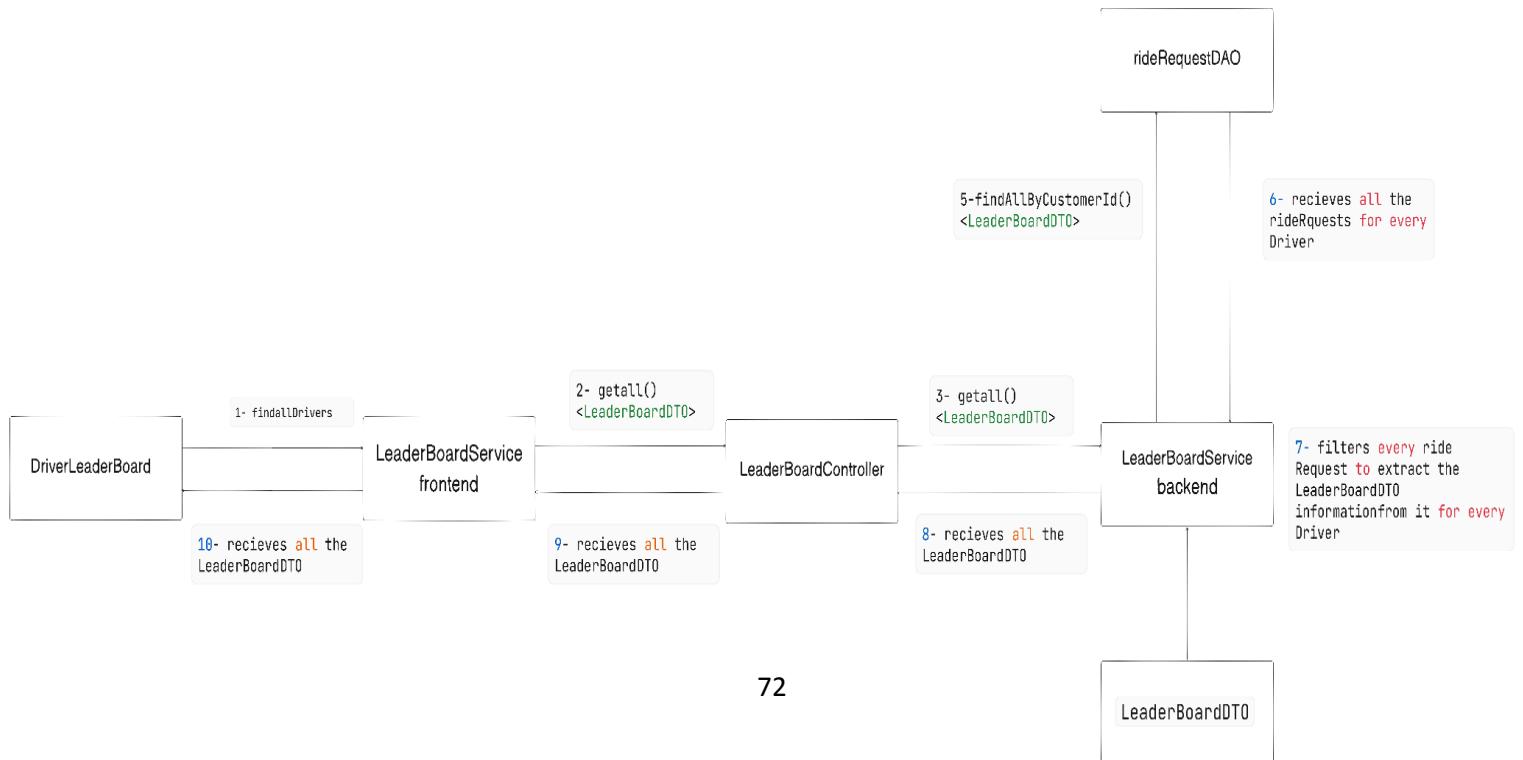
(Chat)



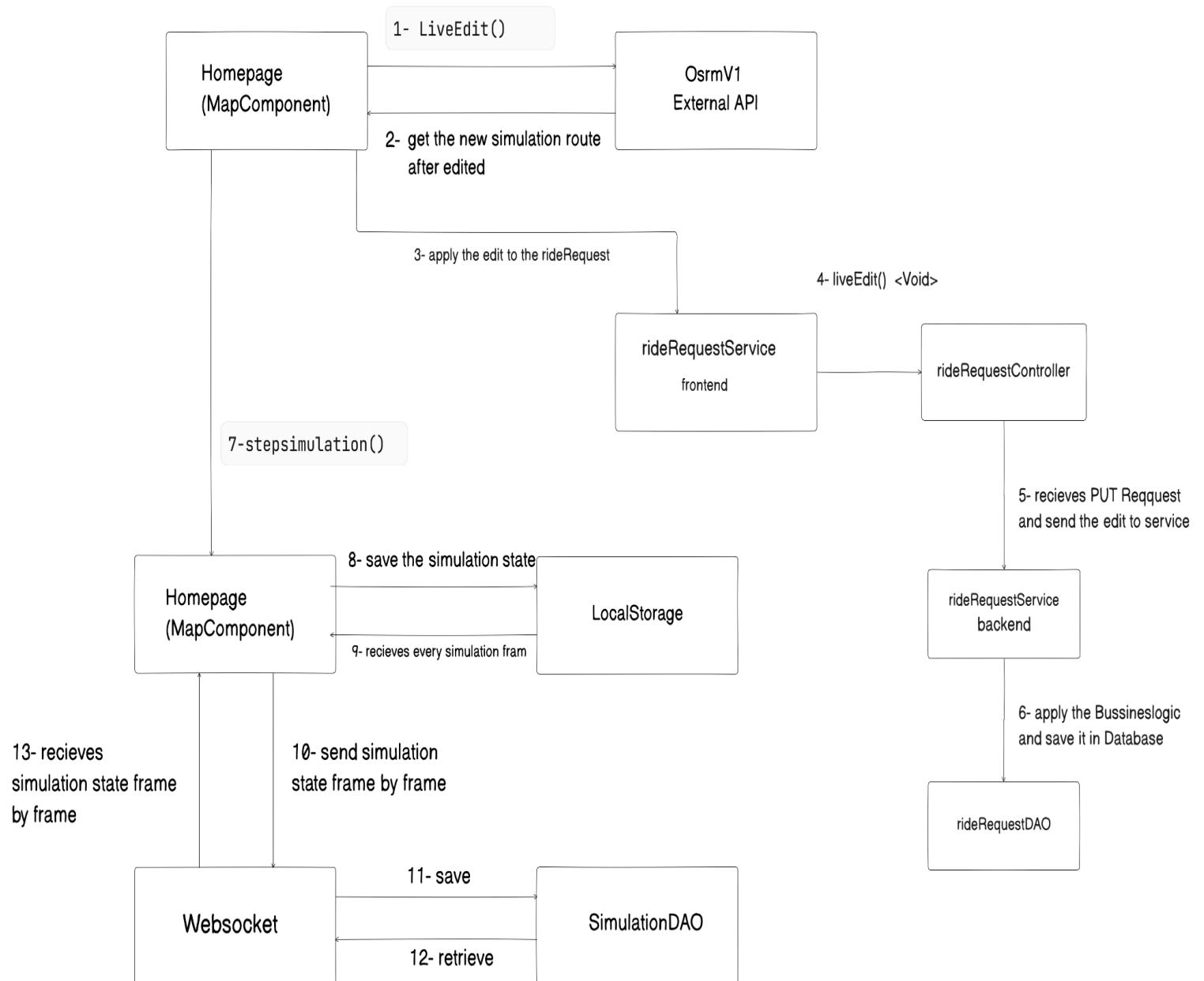
(Grafische Statistiken)



(Fahrer Leaderboard)



(Live Änderungen während der Fahrt)



Funktionalitätsplanung

ID	Funktionalität	Verantwortlicher	Verknüpfte User Stories	Quellcode-referenz	Status
12	Chatfunktion zwischen Fahrer und Kunde				
12. 1	Echtzeit-Übertragung von Nachrichten	Sara Akgüvercin	5.1, 5.2, , 10.2, 11.1, 11.2	gruppeC/frontend/Angular/src/app/components/chat.component.ts	fertig
12. 2	Chatfunktion zwischen Fahrer und Kunde	Sara Akgüvercin	14.1	gruppeC/frontend/Angular/src/app/components/chat.component.ts	fertig
12. 3	Methoden : deleteMessage (), saveEdit ()	Aya Almallal	14.1	gruppeC/frontend/Angular/src/app/components/chat.component.ts (Zeile 360-404)	fertig
13	Fahrten mit mehreren Zwischenstopps				
13. 1	Kunde kann zusätzliche Wegpunkte festlegen und Reihenfolge der Wegpunkte wird berücksichtigt	Ibrahim Sayed (schon fertig im 1.Zyklus)	15.1	gruppeC/frontend/Angular/src/app/components/MAPComponent	fertig
13. 2	UI für Zwischenstops Eingabe	Ibrahim Sayed (schon fertig im 1.Zyklus)	15.1	gruppeC/frontend/Angular/src/app/components/MAPComponent	fertig
14	Grafische Statistiken für Fahrer				
14. 1	Anzeige von Einnahmen, Distanz, Zeit und Bewertung	Rawan Almallal	16.1	gruppeC/frontend/Angular/src/app/components/statistics	fertig
14. 2	Auswahl zwischen täglicher und monatlicher Ansicht	Rawan Almallal	16.2	gruppeC/frontend/Angular/src/app/components/statistics	fertig
15	Live Änderung				

15. 2	UI zur Bearbeitung der Route während der Simulation	Ibrahim Sayed	17.1	gruppeC/frontend/Angular/src/app/components/MAPComponent	fertig
16	Fahrer-Leaderboard				
16. 1	Anzeige aller Fahrer mit Leistungsdaten	Aya Almallal	18.1	gruppeC/frontend/Angular/src/app/components/leaderboard	fertig
16. 2	Sortier- und Filterfunktion der Tabelle	Aya Almallal	18.2	gruppeC/frontend/Angular/src/app/components/leaderboard	fertig
16. 3	Leaderboard Ansicht im Frontend	Aya Almallal	18.1, 18.2	gruppeC/frontend/Angular/src/app/components/leaderboard	fertig

Funktionalitätsplanung(Backend)

ID	Funktionalität	Verantwortlicher	Verknüpfte User Stories	Quellcode-referenz	Status
1	Live Änderung während der Fahrt				
1.1	Kunde kann Ziel/Wegpunkte während Fahrt ändern und System berechnet Route bei Änderung neu	Ibrahim Sayed	17.1	gruppeC/backend/src/main/java/com.example.SEPDrive/Service/riderRequestService gruppeC/backend/src/main/java/com.example.SEPDrive/Service/riderRequestService	fertig
1.2	echt Zeit notification nach der Speicherung der Änderung	Ibrahim Sayed	17.1	gruppeC/backend/src/main/java/com.example.SEPDrive/Service/riderRequestService	fertig
2	Fahrten mit mehreren Zwischenstopps				
2.1	Kunde kann zusätzliche Wegpunkte festlegen und Reihenfolge der Wegpunkte wird berücksichtigt	Ibrahim Sayed	15.1	gruppeC/backend/src/main/java/com.example.SEPDrive/Service/riderRequestService	fertig
3	Grafische Statistiken für Fahrer				

3.1	StatisticService	Bashar Herzallah	16.2	gruppeC/backend /src/main/java/c om.example.SEP Drive/statisticSer vice	fertig
3.2	StatisticController	Bashar Herzallah	16.2	gruppeC/backend /src/main/java/c om.example.SEP Drive/statisticCon troller	fertig
3.3					
4	Fahrer-Leaderboard				
4.1	LeaderboardService	Rias Safi	18.2 , 18.1	gruppeC/backend /src/main/java/c om/example/SEP Drive/Service/Lea derboardService	fertig
4.2	LeaderboardController	Rias Safi	18.2 , 18.1	gruppeC/backend /src/main/java/c om/example/SEP Drive/Controller/ LeaderboardCont roller	fertig
4.3	LeaderboardDTO	Rias Safi	18.2, 18.1	gruppeC/backend /src/main/java/c om/example/SEP Drive/Controller/ LeaderboardDTO	fertig
5	Chatfunktion zwischen Fahrer und Kunde				
5.1	ChatService	Ibrahim Sayed	14.1, 5.1, 5.2, , 10.2, 11.1, 11.2	gruppeC/backend /src/main/java/c om/example/SEP Drive/Service/Ch atService	fertig
5.2	ChatController	Ibrahim Sayed	14.1, 5.1, 5.2, , 10.2, 11.1, 11.2	gruppeC/backend /src/main/java/c om/example/SEP Drive/Controller/ ChatController	fertig
5.3	ChatMessage model	Ibrahim Sayed	14.1, 5.1, 5.2, , 10.2, 11.1, 11.2	gruppeC/backend /src/main/java/c om/example/SEP Drive/model/Cha tMessage.java	fertig

Modultests

ID	Getestete Funktionalität	Quellcode Referenz	Status
2.0	Das ganze leaderboard.Class mit dem benötigte getLeaderboard methode.	gruppeC/backend/src/test/java/com.example.SEPDrive/service/leaderboardServiceTest	fertig

Systemtests

Datum	07.07.2025		
Tester	Sara Akgüvercin		
SW-Version	V 0.2.0		
Vorbedingung(en)	Fahrer hat mindestens 5 abgeschlossene Fahrten mit hinterlegten Daten.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Fahrer öffnet den Statistik-Bereich im Dashboard	Übersicht mit Einnahmen, Distanz, Zeit und Bewertung wird geladen.	✓
2	Fahrer wechselt auf den Tab „Monat“.	System zeigt aggregierte Monatswerte korrekt an.	✓
3	Fahrer bewegt die Maus über die Diagramme.	Tooltip mit genauen Zahlenwerten (z. B. Einnahmen an einem Tag) wird angezeigt.	✓
Nachbedingung(en)	Daten bleiben für zukünftige Analysen verfügbar und korrekt gespeichert.		
Testurteil	Bestanden		

Datum	07.07.2025		
Tester	Sara Akgüvercin		
SW-Version	V 0.2.0		
Vorbedin-gung(en)	Fahrt ist noch nicht gestartet, Ziel ist definiert.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde öffnet Routenplanung und fügt einen Zwischenstopp hinzu.	Neuer Wegpunkt erscheint in der Liste der Stops.	✓
2	Kunde ändert Reihenfolge der Stops per Drag & Drop.	Reihenfolge wird sofort übernommen und angezeigt.	✓
3	Kunde speichert die aktualisierte Route.	System zeigt aktualisierte Route mit allen Stops korrekt auf der Karte an.	✓
Nachbe-dingung(en)	Neue Reihenfolge der Stopps ist in der Systemroute gespeichert.		
Testurteil	Bestanden		

Datum	07.07.2025		
Tester	Sara Akgüvercin		
SW-Version	V 0.2.0		
Vorbedin-gung(en)	Nachricht wurde gesendet, Empfänger hat sie noch nicht gelesen.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Fahrer sendet eine Nachricht an den Kunden.	Nachricht wird im Chatverlauf angezeigt (Status: "ungelesen").	✓
2	Fahrer klickt auf "Löschen".	Nachricht wird erfolgreich entfernt und für den Kunden nicht mehr sichtbar.	✓
3	Fahrer versucht eine gelesene Nachricht zu löschen.	System verweigert Löschung und zeigt Fehlermeldung „Nachricht bereits gelesen“.	✓
Nachbe-dingun(gen)	Nur nicht gelesene Nachrichten sind löschar, alle gelesenen bleiben bestehen.		
Testurteil	Bestanden		

Nutzerhandbuch

Technische Anforderungen

 Technische Voraussetzungen
1. Freie Ports:
- Port 4200 für das Frontend
- Port 8080 für das Backend
- Port 5332 für die PostgreSQL-Datenbank
2. Entwicklungsumgebung: IntelliJ IDEA empfohlen
3. Docker Engine: min. Version 27 (oder höher)
4. Docker Compose: min. Version 2.0 (oder höher)
5. Moderner Webbrowser (z. B. Chrome)
6. Internetverbindung

Installationsanleitung

1. Installieren Sie die folgenden Dateien, bereitgestellt über Sciebo: - gruppe-c-frontend.tar - gruppe-c-backend.tar - postgres.tar - docker-compose.yml
2. Konsole öffnen
3. In das Verzeichnis mit den Dateien navigieren: cd /pfad/zum/ordner

4. Docker-Images manuell laden:

```
docker image load -i gruppe-c-frontend.tar
```

```
docker image load -i gruppe-c-backend.tar
```

```
docker image load -i postgres.tar
```

5. Container starten:

```
docker-compose up -d
```

6. Laufende Container prüfen:

```
docker ps
```

Bedienungsanleitung

1. Öffnen Sie <http://localhost:4200>
2. Die Anwendung führt Sie zuerst zur Welcome-Seite
3. Klicken Sie unten auf Login, um sich einzuloggen
4. Haben Sie noch kein Konto? Dann klicken Sie auf Register
5. Füllen Sie das Formular aus:
 - Benutzername, Passwort
 - Vorname, Nachname
 - E-Mail
 - Geburtsdatum
6. Sie erhalten einen 2FA-Code per E-Mail
Hinweis: Die Mail mit dem Verifizierungscode landet oft im Junk-/Spamordner
Alternativ können Sie den Supercode 999999 eingeben (nur in Entwicklungsumgebung gültig)
7. Nach erfolgreicher Verifizierung wird die Anwendung freigeschaltet.
Sie gelangen zur Kartenansicht (Map)

Fahranfrage

- Geben Sie Ihr Start Adresse oder Ihren aktuellen Standort ein

- Geben Sie Ihr Ziel Adresse
- wählen Sie eine Auto Klasse aus
- dann klicken sie auf (Route Anzeigen), damit sicherzustellen, dass das App die Adresse gespeichert hat

- Klicken Sie auf „Fahranfrage“, um eine Anfrage zu senden
- Die Fahrt erscheint auf der Seite
- Wenn Sie die Anfrage löschen möchten, klicken Sie auf „Löschen“
- Danach das Fahranfrage Fenster schließen und wieder öffnen um die Änderung zu sehen

Profifunktion

- Sie können search bar nutzen um users mit username suchen
- Wenn Sie auf ein Profil klicken, erhalten Sie die Details dazu
- Über den Button „Mein Profil“ sehen Sie Ihre eigenen Daten

Abschluss

Vielen Dank für die Nutzung unserer Anwendung.

Dieses System wurde im Rahmen von SEP Zyklus 1 entwickelt und erfolgreich getestet.

README – SEPDrive (Zyklus 2)

Alles ist fast gleich von erstem Zyklus für Instillation und Technische Voraussetzungen.

Bedienungsanleitung (Anwendung)



Geld Konto auf laden

- Klicken Sie auf den Betrag neben dem Profilbild oben.
- Geben Sie in der Dropdown-Liste den Betrag ein, den Sie aufladen möchten.
- und auf „Add Balance“ Button klicken.



Fahrthistorie

- Auf der **Startseite** (Home-Seite) klicken Sie in der **Navigationsleiste** auf „**Fahranfragen**“. Dort erscheint eine Liste aller Fahranfragen. Oben wird die **aktuelle (aktive) Fahranfrage** angezeigt, darunter folgen alle anderen, die **nicht mehr aktiv** sind. Jede Box repräsentiert eine einzelne Fahranfrage. Die Liste kann **aufsteigend oder absteigend sortiert** werden, und Sie können auch **nach Namen suchen**.
- Zusätzlich können Sie **jede abgeschlossene Fahranfrage bewerten**, falls Sie dies **nicht während der Simulation** gemacht haben oder **wenn Sie Ihre Meinung geändert** haben und ein **besseres oder schlechteres Rating** abgeben möchten.
- Zusätzlich können Sie auch auf jeden Status-Button klicken – daraufhin öffnet sich von rechts eine Karte, die die Route dieser Fahranfrage anzeigt.



FahrtPlannung

- Geben Sie Ihr Start Adresse oder Ihren aktuellen Standort ein
- Geben Sie Ihr Ziel Adresse ein
- Oder Sie können einfach auf die Karte klicken – einmal auf die gewünschte Adresse, um einen Startpunkt zu setzen, und ein weiteres Mal, um das Zielpunkt festzulegen.
- wählen Sie eine Auto Klasse aus

- Dann klicken Sie auf „Route anzeigen“, um die Route auf der Karte darzustellen sowie die Entfernung, den Preis und die Dauer der Fahrt zu berechnen – noch bevor die Fahrt erstellt wird.
- Klicken Sie auf „Fahranfrage“, um eine Anfrage zu senden
- Die Fahrt erscheint auf der Seite wenn Sie auf „Fahranfrage“ Button Klicken
- Wenn Sie die Anfrage löschen möchten, klicken Sie auf „Löschen“
- Und Sie Können auf Status Button Klicken um die Route nochmal zu sehen



Liste der verfügbaren Fahranfragen

- Als Fahrer sehen Sie in der Navigationsleiste einen neuen Button mit dem Namen „Driver Dashboard“. Klicken Sie darauf, um eine Tabelle zu öffnen, die alle aktiven Fahranfragen aller Nutzer enthält.**
- Sie können jede Spalte auf- oder absteigend sortieren. Außerdem steht Ihnen eine globale Suchfunktion zur Verfügung, mit der Sie fahrtenübergreifend nach Benutzernamen, vollständigen Namen oder Adressen suchen können. Zusätzlich gibt es für jede Spalte eine eigene Filtersuche, mit der Sie gezielt innerhalb dieser Spalte filtern können.**
- Die "Pickup-Distanz" steht für die Entfernung zwischen Ihnen und der Abholadresse des Kunden. Anfangs beträgt dieser Wert 0 – bis Sie Ihre Adresse eingeben und auf den Emoji-Button mit dem Marker auf der Karte klicken. Dann wird die Entfernung automatisch berechnet.**
- Falls Sie Ihre aktuelle Position verwenden möchten, lassen Sie das Adressfeld leer und klicken Sie direkt auf den Emoji-Button. In diesem Fall wird die Entfernung zwischen Ihrem aktuellen Standort und der Abholadresse ermittelt.**
- Wenn Sie – so wie ich – ein visueller Mensch sind, gibt es am Ende jeder Zeile einen Karten-Button. Wenn Sie darauf klicken, wird die Route auf der Karte angezeigt. Dort sehen Sie Ihre aktuelle Position sowie den gesamten Fahrweg der jeweiligen Fahrt.**
- Sie können jede beliebige Fahrt auswählen und ein Angebot dafür abgeben. Der Kunde wird sofort über Ihr Angebot benachrichtigt. Solange der Kunde das Angebot nicht**

annimmt oder ablehnt, können Sie kein weiteres Angebot abgeben – es ist nur ein Angebot pro Fahrt möglich.

-Sie haben jedoch die Möglichkeit, Ihr Angebot wieder zu stornieren – zum Beispiel, wenn der Kunde zu lange nicht reagiert oder wenn Sie einfach Ihre Meinung geändert haben. Nach der Stornierung können Sie ein neues Angebot abgeben.

-Wichtig: Sie können nur dann ein Angebot machen, wenn Sie über die passende Fahrzeugklasse verfügen, die der Kunde in seiner Fahranfrage angegeben hat.



Fahrtangebote

Sobald ein Fahrer Ihnen ein Angebot macht, werden Sie sofort benachrichtigt. Über den Button „Fahrtangebote“ in der Navigationsleiste gelangen Sie zur Übersicht Ihrer aktuellen Angebote. Dort sehen Sie eine Tabelle mit allen erhaltenen Fahrtangeboten, die Sie – wie gewohnt – durchsuchen und sortieren können.

Sie haben die Möglichkeit, ein Angebot anzunehmen oder abzulehnen – basierend auf der Bewertung und den Informationen des Fahrers. Bitte beachten Sie: Sobald Sie ein Angebot annehmen, werden alle anderen automatisch abgelehnt.

Nach der Annahme eines Angebots startet die Fahrsimulation direkt. Wo genau Sie diese finden, erfahren Sie im Abschnitt „Durchführung einer Fahrt“.



Durchführung einer Fahrt

Die Simulation startet automatisch, sobald ein Kunde ein Angebot eines Fahrers annimmt.

Um den aktuellen Stand der Simulation zu verfolgen, gehen Sie zur Startseite und klicken Sie auf den Button „Fahranfragen“. Dort sehen Sie, dass sich der Status der aktiven Fahrt von einem grünen Button zu einem roten „Live“-Button geändert hat. Wenn Sie darauf klicken, öffnet sich auf der rechten Seite eine Karte.

Auf dieser Karte sehen Sie einen sich bewegenden Marker, der die Route der Fahrt darstellt. Sie können in die Karte hinein- oder herauszoomen oder auf den Button „Fokus“ klicken, um die Karte automatisch mit dem Marker mitbewegen zu lassen. Außerdem können Sie die Simulation jederzeit pausieren oder fortsetzen.

Es gibt auch einen Schieberegler (Slider), mit dem Sie die Geschwindigkeit der Simulation anpassen können. Bitte beachten Sie: Ziehen Sie den Regler nicht, da dabei jede Zwischenstufe als neue Geschwindigkeit übermittelt wird – was zu Verkehrsproblemen in der Simulation führen kann. Klicken Sie stattdessen einfach auf die gewünschte Geschwindigkeit, damit sie korrekt übernommen wird.

Bitte beachten Sie, dass die Simulation bei jedem Zwischenstopp automatisch pausiert – denn logisch betrachtet wurde dieser Zwischenstopp eingeplant, um dort tatsächlich anzuhalten. Sie werden benachrichtigt, sobald die Fahrt an einem Zwischenstopp anhält. Sowohl der Fahrer als auch der Kunde können die Simulation anschließend manuell wieder starten.

Nach dem Ende der Simulation wird der Fahrpreis automatisch vom Kundenkonto abgebucht und dem Fahrerkonto gutgeschrieben. Beide Parteien erhalten darüber eine Benachrichtigung.

Die Fahrt wechselt anschließend vom Status „aktiv“ zu „abgeschlossen“. Direkt im Anschluss erscheint für Fahrer und Kunde ein Bewertungsfenster, in dem sie einander bewerten können.

! ! - Wichtig

Beim Erstellen einer Fahranfrage und dem Zeichnen der Route kann es gelegentlich vorkommen, dass die Straße nicht korrekt angezeigt wird – dies liegt meist daran, dass die Karte nicht richtig geladen wurde. In diesem Fall aktualisieren Sie bitte die Seite und versuchen es erneut.

Wenn Sie zwei Fenster gleichzeitig verwenden, um die Simulation zu beobachten – insbesondere im privaten oder Inkognito-Modus – kann es vorkommen, dass die Karte nicht korrekt geladen wird.

Wenn Sie auf den Simulations-Button klicken und zwar die Karte erscheint, aber keine Bewegung sichtbar ist, läuft die Simulation im Hintergrund dennoch weiter. In diesem Fall

hilft es, die Seite zu aktualisieren, damit die Karte korrekt geladen wird und Sie die Simulation wieder sehen können.

Sie können auf jede Stelle klicken, an der ein Benutzername oder Name angezeigt wird – Sie werden dann zur Profilseite des jeweiligen Nutzers weitergeleitet.

Um zu Ihrem eigenen Profil zu gelangen, klicken Sie einfach auf Ihr Profilbild oder Ihren Namen, die in der Kopfzeile angezeigt werden.

README – SEPDrive (Zyklus 3)

· Leaderboard & Statistiken

Wenn Sie die **Leaderboard** oder die **Statistiken** erreichen möchten, klicken Sie einfach auf Ihr Profil. Dort finden Sie drei Icons:

1. Such-Icon
2. Leaderboard-Icon
3. Statistik-Icon

· Chat öffnen

Möchten Sie den Chat starten, öffnen Sie Ihre **aktiven Fahranfragen**. Dort finden Sie den **Chat-Button** – klicken Sie darauf, um direkt mit dem Fahrer bzw. Kunden zu kommunizieren.

Live Änderung:

Möchten Sie einen Zwischenstopp in den **Live-Änderungen** hinzufügen, geben Sie eine Zahl gefolgt von einem Bindestrich "-" und dem neuen Zwischenstopp ein, z. B. **1- Essen**. Zwischen jedem Zwischenstopp muss eine leere Zeile stehen.

Hinweis: Die Nummern dienen nur zur Kennzeichnung und beeinflussen nicht die Anzeigereihenfolge der Zwischenstopps. Zum Beispiel wird **3- Wittenscheid** zwischen den anderen Stopps angezeigt, obwohl es die Zahl 3 trägt.