

# **SEP Projektmappe**

***PROJEKTMAPPE DES PROJEKTES***

## ***CarMate***

### **Dokumentation des Projektes**

Gruppe G:

Luis Bleckmann

Lucy Brandt

Dario Connor Redding

Felix Maximilian Rieger

Konstandin Shuti

Dilay Topal

Noel von Stockum

## 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.*

CarMate ist eine Mitfahr-App, bei der Kunden Fahrtanfragen erstellen und Fahrtangebote von Fahrern erhalten. CarMate übernimmt die Routenplanung und bietet eine kartenbasierte Visualisierung zur Einrichtung von Fahrten, zur Anzeige der geplanten Route und zur Verfolgung der Fahrt, während sie sich vom Startpunkt zum Ziel bewegen.

Das Projekt soll als Webanwendung (bestehend aus einem Front- und Backend) umgesetzt werden. Front- und Backend sollen voneinander separat entwickelt werden und jeweils als eigenständige Docker-Container laufen. Die Kommunikation zwischen Backend und Frontend soll über eine Netzwerkverbindung erfolgen.

# Zyklus I

## Spezifikationsplanung

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
1.	Frontend Planung			
1.1.1	Registrierung Fahrer	User Story	Dilay Topal	Abgeschlossen ▾
1.1.2	Registrierung Kunde	User Story	Felix Maximilian Rieger	Abgeschlossen ▾
1.1.3	Login	User Story	Felix Maximilian Rieger	Abgeschlossen ▾
1.1.4	Multi-Faktor-Auth.	User Story	Felix Maximilian Rieger	Abgeschlossen ▾
1.1.5	Super Sicherheitscode	User Story	Felix Maximilian Rieger	Abgeschlossen ▾
1.1.6	Profil Eigenes Fahrer	User Story	Dilay Topal	Abgeschlossen ▾
1.1.7	Profil Eigenes Kunde	User Story	Dilay Topal	Abgeschlossen ▾
1.1.8	Profil Suche Fahrer	User Story	Konstandin Shuti	Abgeschlossen ▾
1.1.9	Profil Suche Kunde	User Story	Dilay Topal	Abgeschlossen ▾
1.1.10	Karte Sichtbarkeit Startpunkt	User Story	Konstandin Shuti	Abgeschlossen ▾
1.1.11	Karte Sichtbarkeit Ziel	User Story	Luis Bleckmann	Abgeschlossen ▾
1.1.12	Karte Sichtbarkeit Zwischenstopps	User Story	Luis Bleckmann	Abgeschlossen ▾
1.1.13	Karte Sichtbarkeit Strecke	User Story	Luis Bleckmann	Abgeschlossen ▾
1.1.14	Karte Steuerung Zoom	User Story	Luis Bleckmann	Abgeschlossen ▾
1.1.15	Karte Steuerung Verschieben	User Story	Luis Bleckmann	Abgeschlossen ▾
1.1.16	Fahranfrage Angabe Startpunkt	User Story	Luis Bleckmann	Abgeschlossen ▾
1.1.17	Fahranfrage Angabe Ziel	User Story	Luis Bleckmann	Abgeschlossen ▾
1.1.18	Fahranfrage Angabe Fahrzeugklasse	User Story	Konstandin Shuti	Abgeschlossen ▾
1.1.19	Fahranfrage Startpunkt aktuelle Position	User Story	Lucy Brandt	Abgeschlossen ▾

1.1.20	Fahranfrage Startpunkt Adresse	User Story	Dario Connor Reddig	Abgeschlossen ▾
1.1.21	Fahranfrage Startpunkt POI	User Story	Noel von Stockum	Abgeschlossen ▾
1.1.22	Fahranfrage Startpunkt Koordinate	User Story	Dario Connor Reddig	Abgeschlossen ▾
1.1.23	Fahranfrage Zielpunkt Adresse	User Story	Dario Connor Reddig	Abgeschlossen ▾
1.1.24	Fahranfrage Zielpunkt POI	User Story	Noel von Stockum	Abgeschlossen ▾
1.1.25	Fahranfrage Zielpunkt Koordinate	User Story	Dario Connor Reddig	Abgeschlossen ▾
1.1.26	Fahranfrage Aktiv nach Erstellung	User Story	Noel von Stockum	Abgeschlossen ▾
1.1.27	Fahranfrage Maximal 1 aktive Fahranfrage	User Story	Noel von Stockum	Abgeschlossen ▾
1.1.28	Fahranfrage Aktive Fahranfrage einsehen	User Story	Noel von Stockum	Abgeschlossen ▾
1.1.29	Fahranfrage Aktive Fahranfrage löschen	User Story	Noel von Stockum	Abgeschlossen ▾
1.2.1	Design Login	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.2	Design Register	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.3	Design Search	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.4	Design Profil-Fahrer	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.5	Design Profil-Nutzer	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.6	Design Fahrdaten	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.7	Design Fahranfrage	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.8	Design Aktuelle Fahrt	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.9	Design Karte	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
1.2.10	Multi-Faktor-Auth.	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
<b>2.</b>	Backend Architektur			
2.1	Komponentendiagramm Backend	Strukturdiagramm	Felix Maximilian Rieger	Abgeschlossen ▾
2.2	Klassendiagramm Backend	Strukturdiagramm	Luis Bleckmann	Abgeschlossen ▾

2.3.1	Kommunikationsdiagramm Login	Verhaltensdiagramm	Noel von Stockum	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.3.2	Kommunikationsdiagramm Registration	Verhaltensdiagramm	Lucy Brandt	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.3.3	Kommunikationsdiagramm Profil Ansehen	Verhaltensdiagramm	Dario Connor Reddig	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.3.4	Kommunikationsdiagramm Kartenvisualisierung	Verhaltensdiagramm	Konstandin Shuti	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.3.5	Kommunikationsdiagramm Fahranfrage Erstellen	Verhaltensdiagramm	Dilay Topal	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.4	Datenbank Modell	Strukturdiagramm	Felix Maximilian Rieger	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>

Jedes Artefakt, das im Rahmen des SEP erstellt wird, muss mit dem Namen genau einer [Verantwortlichen/einen](#) 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 obenstehenden Tabelle entsprechend eingetragen werden muss. Natürlich kann die Gruppe gemeinsam an einem Artefakt arbeiten, als Verantwortlicher sollte aber genau eine Person eingetragen werden.

## User-Stories

<b>User Story-ID</b>	1.1.1
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich mich registrieren können, sodass ich Kunden finden und interagieren kann und somit Fahrten durchführe, um Geld verdienen zu können.
<b>Geschätzter Realisierungsaufwand</b>	5 Tage
<b>Priorität</b>	hoch
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.3, 1.1.6

<b>User Story-ID</b>	1.1.2
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich mich registrieren können, sodass andere Nutzer mich finden können.
<b>Geschätzter Realisierungsaufwand</b>	7 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Felix Maximilian Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	1.1.3
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich mich einloggen können, um meine Daten einsehen zu können.
<b>Geschätzter Realisierungsaufwand</b>	7 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Felix Maximilian Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.2

<b>User Story-ID</b>	1.1.4
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich mich mittels zwei Faktor Authentifizierung anmelden können, um mich sicherer mit meinen Daten zu fühlen.
<b>Geschätzter Realisierungsaufwand</b>	7 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Felix Maximilian Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.1; 1.1.2; 1.1.3

<b>User Story-ID</b>	1.1.5
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich einen Super-Code haben, um weiter den Zugriff zu meinem Account zu behalten selbst wenn ich keinen Zugriff zu meiner E-Mail habe.
<b>Geschätzter Realisierungsaufwand</b>	1 Tage
<b>Priorität</b>	Niedrig
<b>Autor</b>	Felix Maximilian Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.1; 1.1.2; 1.1.3; 1.1.4

<b>User Story-ID</b>	1.1.6
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich mein eigenes Profil erstellen und bearbeiten können, damit ich meine persönlichen Daten und Fahrzeuginformationen anpassen und aktualisieren kann.
<b>Geschätzter Realisierungsaufwand</b>	5 Tage
<b>Priorität</b>	mittel
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.1, 1.1.3, 1.1.8

<b>User Story-ID</b>	1.1.7
<b>User Story-Beschreibung</b>	Als Kunde möchte ich mein eigenes Profil einsehen und bearbeiten können, damit ich meine persönlichen Informationen wie Name, E-Mail Adresse, Gesamtzahl der Fahrten benennen und aktuell halten kann.
<b>Geschätzter Realisierungsaufwand</b>	3 Tage
<b>Priorität</b>	mittel
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.2, 1.1.3

<b>User Story-ID</b>	1.1.8
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich nach Fahrer suchen können, damit ich einen passenden Fahrer basierend auf Bewertungen oder anderen Kriterien finden kann.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.1, 1.1.6, 1.1.9

<b>User Story-ID</b>	1.1.9
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich nach Kunden suchen und relevante Informationen einsehen können, damit ich mir den Kunden aussuchen kann, dessen Profil für mich am besten passt.
<b>Geschätzter Realisierungsaufwand</b>	4 Tage
<b>Priorität</b>	mittel
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.2, 1.1.7

<b>User Story-ID</b>	1.1.10
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich meinen Startpunkt auf einer Karte sehen, damit ich den gewünschten Abholort sicherstellen kann.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.13, 1.1.16, 1.1.19, 1.1.20, 1.1.21, 1.1.22

<b>User Story-ID</b>	1.1.11
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich mein Ziel auf der Karte sehen können, damit ich nachvollziehen kann, wo die Fahrt endet und wie weit es noch ist.
<b>Geschätzter Realisierungsaufwand</b>	2 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.17

<b>User Story-ID</b>	1.1.12
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich alle meine geplanten Zwischenstopps auf der Karte sehen können, damit ich den gesamten Verlauf meiner Route besser nachvollziehen kann
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.11

<b>User Story-ID</b>	1.1.13
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich die gesamte geplante Strecke auf der Karte sehen können, damit ich die Route nachvollziehen und mich besser orientieren kann.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.11, 1.1.12

<b>User Story-ID</b>	1.1.14
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich auf der Karte rein- und rauszoomen können, um mir bestimmte Streckenabschnitte oder Details besser ansehen zu können.
<b>Geschätzter Realisierungsaufwand</b>	0.5 Tage
<b>Priorität</b>	Niedrig
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.11

<b>User Story-ID</b>	1.1.15
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich die Karte per Touch oder Maus verschieben können, um verschiedene Bereiche der Route flexibel einsehen zu können.
<b>Geschätzter Realisierungsaufwand</b>	0.5 Tage
<b>Priorität</b>	Niedrig
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	1.1.16
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich einen Startpunkt für meine Fahrt angeben können, damit der Fahrer weiß, wo er mich abholen soll.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	1.1.17
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich ein Ziel für meine Fahrt angeben können, damit der Fahrer weiß, wohin ich gefahren werden möchte.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	1.1.18
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich bei der Fahranfrage die gewünschte Fahrzeugklasse angeben können, damit ich ein Fahrzeug buche, das zu meinen Bedürfnissen passt.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Konstantin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.26

<b>User Story-ID</b>	1.1.19
<b>User Story-Beschreibung</b>	Als Nutzer der Fahrdienst-App möchte ich eine Fahranfrage mit meinem aktuellen Standort als Startpunkt stellen, sodass ich schnell und bequem eine Fahrt von meinem momentanen Aufenthaltsort aus buchen kann.
<b>Geschätzter Realisierungsaufwand</b>	2 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Lucy Brandt
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16

<b>User Story-ID</b>	1.1.20
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich meine Fahranfrage mit meiner Adresse, die nicht mein aktueller Standort ist, als Startpunkt verschicken, damit ich einen lokal verfügbaren Fahrer erhalte.
<b>Geschätzter Realisierungsaufwand</b>	0,5 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16

<b>User Story-ID</b>	1.1.21
<b>User Story-Beschreibung</b>	Als Kunde möchte ich von bestimmten Orten(POI) meiner Stadt navigieren können, um vom Krankenhaus abgeholt werden zu können, ohne die Adresse raus zu suchen.
<b>Geschätzter Realisierungsaufwand</b>	3 Tage
<b>Priorität</b>	Mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16

<b>User Story-ID</b>	1.1.22
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich meine Fahranfrage mit Koordinaten als Startpunkt verschicken, damit ich eine höhere Sicherheit habe, dass ich die richtige Adresse angebe.
<b>Geschätzter Realisierungsaufwand</b>	0,5 Tage
<b>Priorität</b>	Niedrig
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16

<b>User Story-ID</b>	1.1.23
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich bei meiner Fahranfrage einen Adresse eingeben können, damit ich nicht allzu viel Zeit verliere.
<b>Geschätzter Realisierungsaufwand</b>	0,5 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16

<b>User Story-ID</b>	1.1.24
<b>User Story-Beschreibung</b>	Als Kunde möchte ich zu den wichtigen Orten (POI) meiner Stadt navigieren können, um zum Krankenhaus zu kommen ohne die Adresse raus zu suchen.
<b>Geschätzter Realisierungsaufwand</b>	3 Tage
<b>Priorität</b>	Mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.17 ; 1.1.21

<b>User Story-ID</b>	1.1.25
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich bei meiner Fahranfrage Koordinaten für meine Zieladresse eingeben können, damit ich sicherstellen kann, dass ich an der richtigen Adresse ankomme.
<b>Geschätzter Realisierungsaufwand</b>	0,5 Tage
<b>Priorität</b>	Niedrig
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16

<b>User Story-ID</b>	1.1.26
<b>User Story-Beschreibung</b>	Als Kunde möchte ich, dass meine letzte Fahranfrage als aktive Fahranfrage übernommen wird, um klicks zu sparen.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16 ; 1.1.17 ; 1.1.18

<b>User Story-ID</b>	1.1.27
<b>User Story-Beschreibung</b>	Als Kunde möchte ich nur eine aktive Fahranfrage haben können, um nicht ausversehen zu viel zu buchen und dann mehr zahlen zu müssen.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.26

<b>User Story-ID</b>	1.1.28
<b>User Story-Beschreibung</b>	Als Kunde möchte ich meine aktive Fahranfrage sehen können, um sicherzugehen, dass meine Angaben richtig waren.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.26

<b>User Story-ID</b>	1.1.29
<b>User Story-Beschreibung</b>	Als Kunde möchte ich meine aktive Fahranfrage löschen können, um fehler zu korrigieren oder nicht mehr benötigte Fahrten abzubrechen.
<b>Geschätzter Realisierungsaufwand</b>	2 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.28

## Papierprototypen

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

The image displays four separate paper prototypes of a user interface for a car booking service, arranged in a 2x2 grid. Each prototype is a hand-drawn sketch with various UI elements labeled for clarity.

- Top Left (Login Screen):** A modal window titled "Sicherheitscode" (Security Code) with the sub-instruction "Bitte Code eingeben" (Please enter code). It features a 4x4 grid input field and a "Code erneut an email senden" (Send code again via email) button. Below the modal is the main "Login" screen with "Benutzername" and "Passwort" fields, a "Registrieren" (Register) button with a checkmark, and a "Logo" placeholder.
- Top Right (Registration Screen):** A modal window titled "PopUp-Registrierung als Fahrer" (PopUp-Registration as Driver) with the question "Welches fahrzeug führen sie?" (Which vehicle do you drive?). It includes a dropdown menu set to "Klein" (Small), a "Ausklappmenü" (Drop-down menu) with options "Medium" and "Deluxe", and a "Logo" placeholder. Below the modal is the main "Register" screen with fields for "Name", "Nachname", "Benutzername", "Email", "Geburtsdatum", "Passwort", and a "Registrieren" (Register) button.
- Bottom Left (Route Planning Screen):** A "fahrdaten" (Driving Data) section with tabs for "Home", "Fahrt Planen" (Plan Journey), "Statistik" (Statistics), "Suche" (Search), and "Einstellungen" (Settings). It shows a route planning interface with a map of Altendorf showing a blue route line. On the left, there are fields for "Startpunkt" (Musterstraße 44), "Ziel" (Uni Duisburg-Essen), and "Fahrzeug" (Klein). In the center, a vertical stack of locations is shown: Musterstraße 44, + McDonalds, + Disneyland, and Uni Duisburg-Essen. At the bottom are "Logo" and "Suchen" buttons.
- Bottom Right (Booking Confirmation Screen):** A "Bestätigung" (Confirmation) section with tabs for "Home", "Fahrt Planen", "Statistik", "Suche", and "Einstellungen". It displays a message "Es wurde eine Fahranfrage gesendet" (A ride request has been sent) and "Antwort: ausstehend" (Answer: pending). It lists the trip details: "Von: jhsadkdjnjejhajhdah", "Nach: hsgdha kejdajhajhdja", "Auto: dashjsdkjwal", and "Kosten: jdjhkjas". A "Anfrage Rückziehen" (Revoke Request) button is at the bottom, along with a small note about canceling the booking.

The image displays three screenshots of a mobile application interface for a car sharing platform.

**Top Left Screenshot:** A driver profile page titled "Profil - Fahrer". It shows a placeholder profile picture, the name "Nutzername" (Nachname, Name), a 3-star rating, and statistics: 13 trips, a mail address (kdjahkljskd), birth date (dsfhjdshfkjhjkdks), and driving license (jdjshjdhjsfk). Buttons for "Chat Starten" and "Logo" are visible.

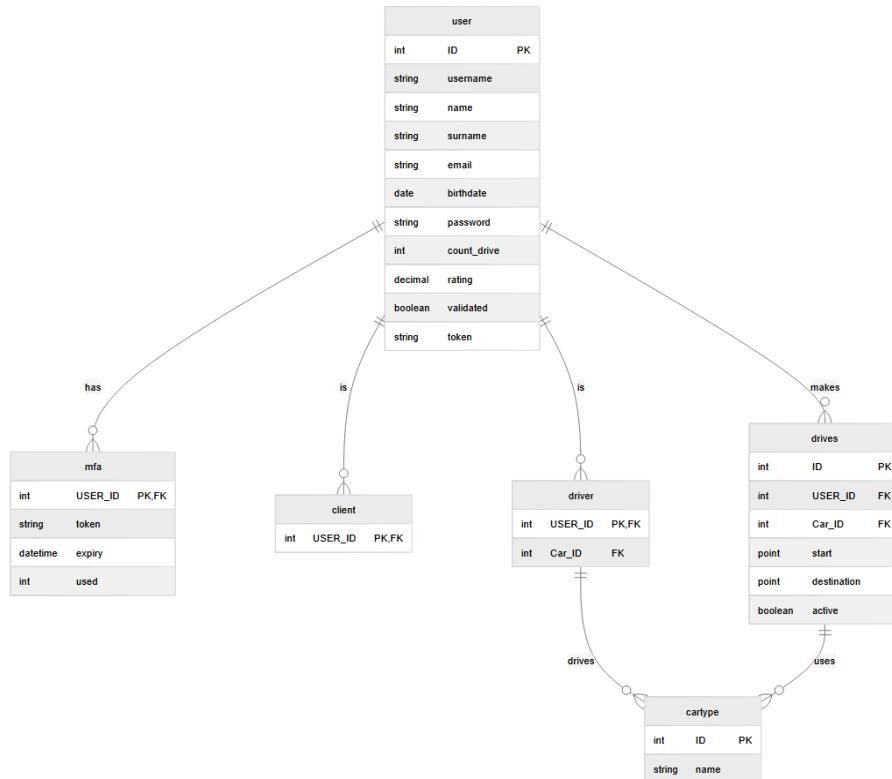
**Top Right Screenshot:** A user profile page titled "Profil - Nutzer". It shows a placeholder profile picture, the name "Nutzername" (Nachname, Name), a 5-star rating, and statistics: 13 trips, a mail address (kdjahkljskd), birth date (dsfhjdshfkjhjkdks). It also shows 164 reviews. Buttons for "Chat Starten" and "Logo" are visible.

**Bottom Screenshot:** A search results page titled "Search". It lists four profiles, each with a placeholder profile picture, the name "name", a 3-star rating, and a "Profil ansehen" button. The profiles are identical to the ones in the top screenshots. A "Logo" button is visible at the bottom left.

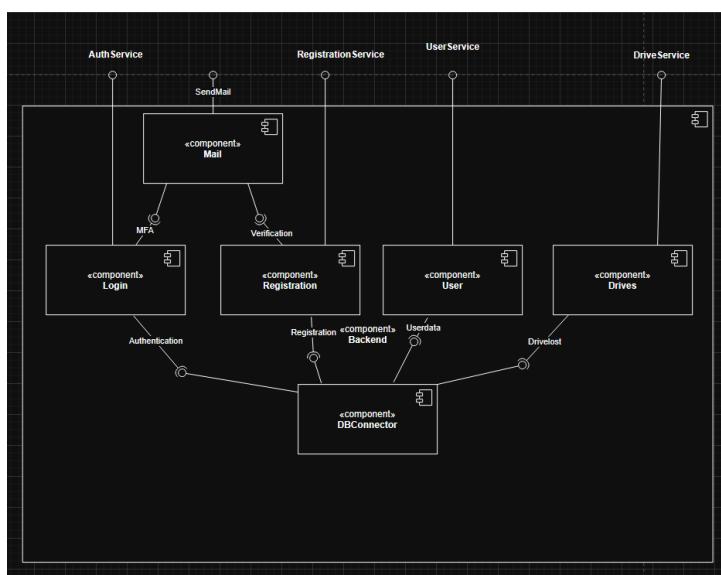
## Strukturdiagramme (Komponenten- und Klassendiagramme)

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.

**Datenbankmodell (Felix Maximilian Rieger):**



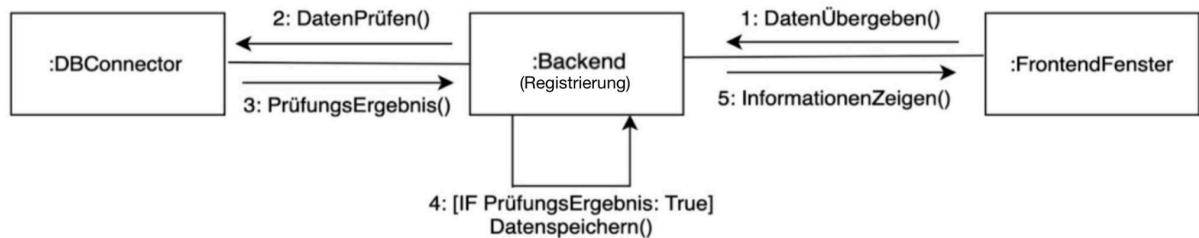
**Komponentendiagramm (Felix Maximilian Rieger):**



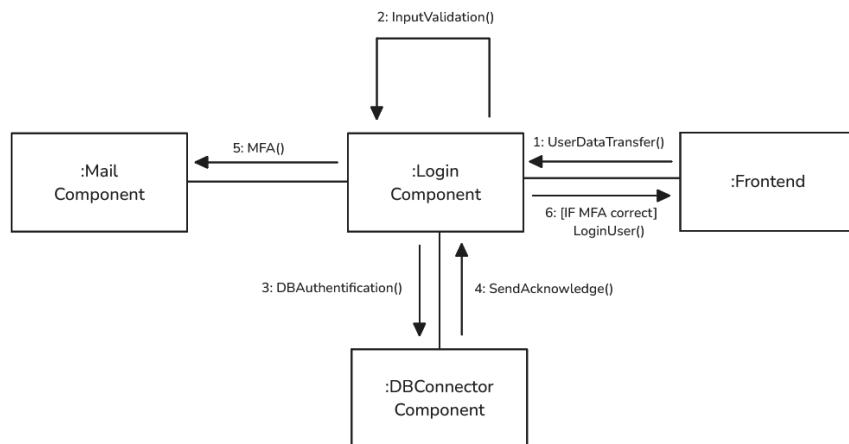
## Verhaltensdiagramme (Kommunikationsdiagramme)

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

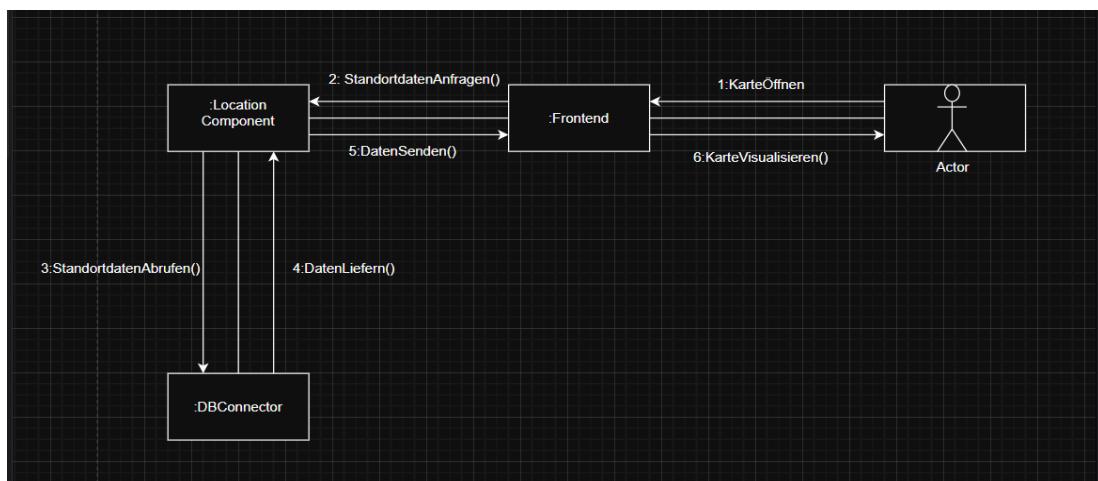
### Registrierung (Lucy Brandt):



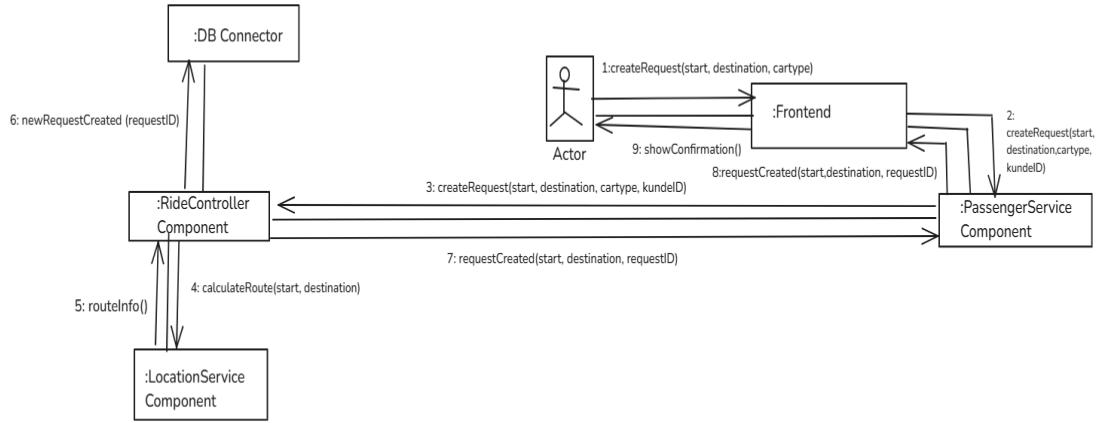
### Login (Noel von Stockum):



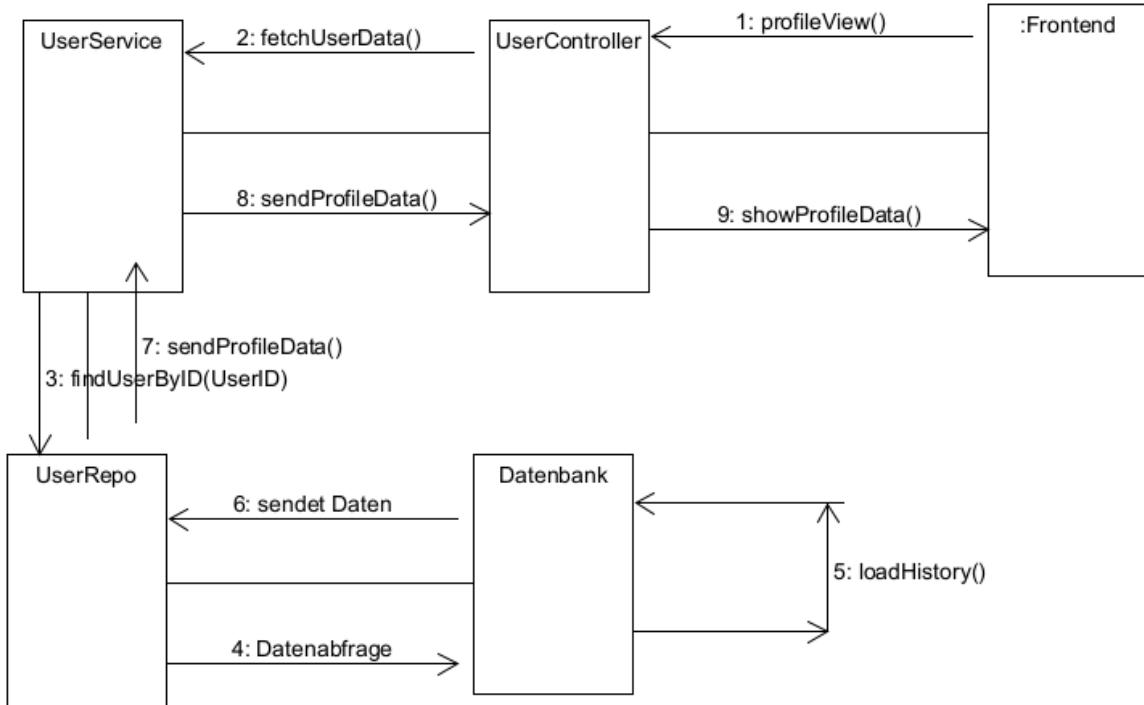
### Kartenvisualisierung (Konstandin Shuti):



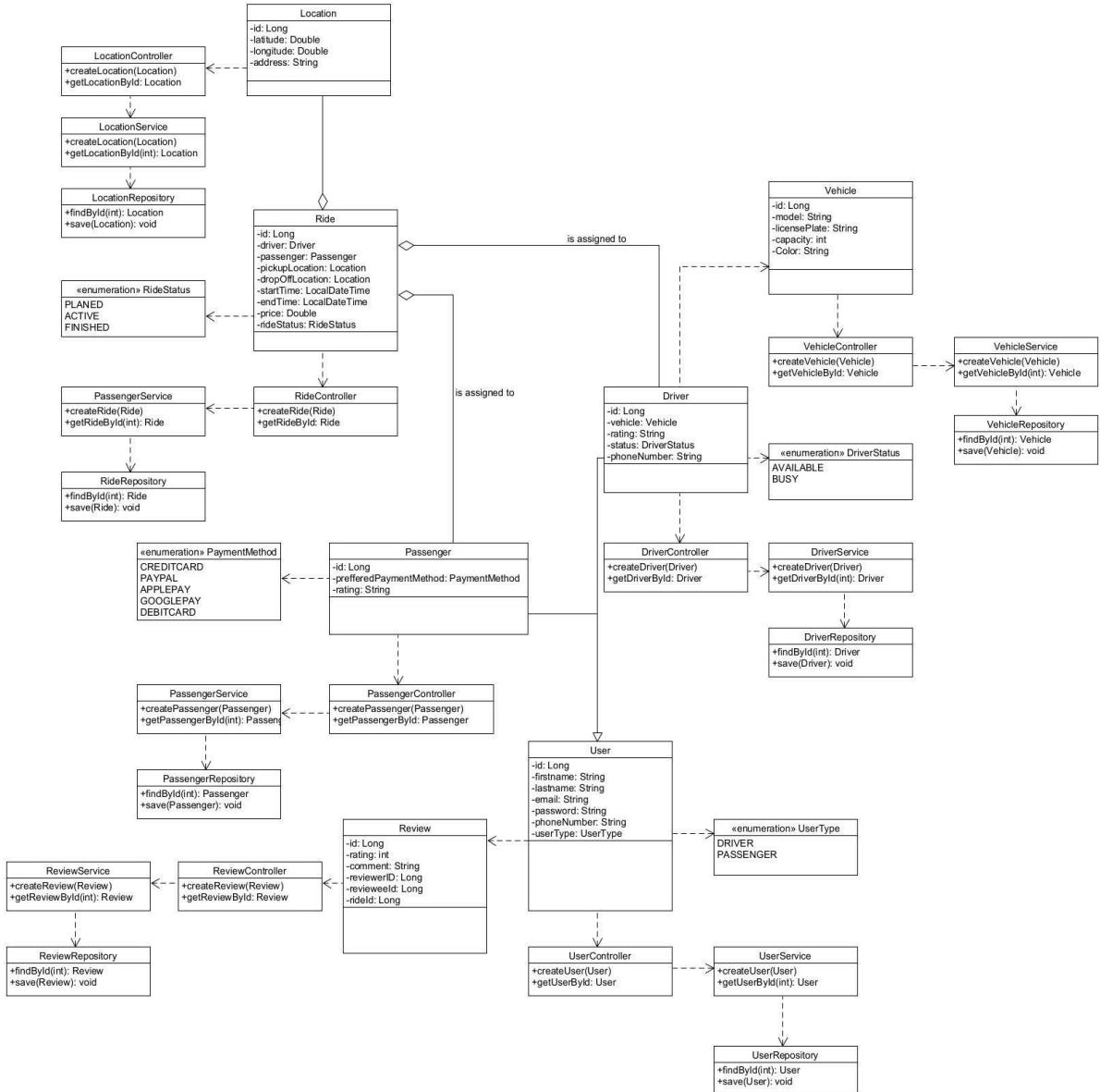
## Fahranfrage erstellen (Dilay Topal)



## Profilansicht (Dario Connor Reddig) :



## UML-Classdiagram (Luis Bleckmann)



## Funktionalitätsplanung

ID	Funktionalität	Verantwortliche/-r	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
<b>1.</b>	<b>Frontend</b>					
1.1	Registration Komponente	Noel von Stockum	2.2 ; 2.3 ; 2.10	1.1.1 ; 1.1.2	Frontend/src/app/components/register	Abgeschlossen ▾
1.2	Login Komponente	Dilay Topal	1.1 ; 2.1 ; 2.3	1.1.3 ; 1.1.4 ; 1.1.5	Frontend/src/app/components/login	Abgeschlossen ▾
1.3	Karte Komponente	Noel von Stockum		1.1.10 bis 1.1.15	Frontend/src/app/components/map	Abgeschlossen ▾
1.4	Fahrdaten Komponente	Dilay Topal	2.6	1.1.16 bis 1.1.25	Frontend/src/app/components/drivedata	Abgeschlossen ▾
1.5	Buchung Komponente	Dilay Topal	2.6	1.1.28	Frontend/src/app/components/booking	Abgeschlossen ▾
1.6	Aktuelle Fahrt Komponente	Noel von Stockum	2.7 ; 2.8	1.1.26 bis 1.1.29	Frontend/src/app/components/current-drive	Abgeschlossen ▾
1.7	Profil Fahrer Komponente	Lucy Brandt	2.4 ; 2.11	1.1.6	Frontend/src/app/components/profile	Abgeschlossen ▾
1.8	Profil Nutzer Komponente	Lucy Brandt	2.4 ; 2.12	1.1.7	Frontend/src/app/components/profile	Abgeschlossen ▾
1.9	Suche Komponente	Lucy Brandt	2.4 ; 2.5	1.1.8 ; 1.1.9	Frontend/src/app/components/search	Abgeschlossen ▾
1.10	App Komponente	Noel von Stockum			Frontend/src/app	Abgeschlossen ▾
1.11	MenuBar Komponente	Lucy Brandt			Frontend/src/app/components/menu-bar	Abgeschlossen ▾
1.12	Ride Service	Dilay Topal			Frontend/src/app/services/ride.service	Abgeschlossen ▾
<b>2.</b>	<b>Backend</b>					
2.1	Login Endpoints	Felix Rieger	2.2	1.1.3 ; 1.1.4 ; 1.1.5	AuthController.java, AuthRepo.java, User.java	Abgeschlossen ▾
2.2	Register Endpoint	Felix Rieger		1.1.1 ; 1.1.2	RegisterController.java, RegisterRepo.java, admin	Abgeschlossen ▾

2.3	Mail Connector	Felix Rieger			MailData.java, MailService.java, SpamService.java, SpamRepo.java	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.4	User Profil Speicherung	Konstandin Shuti		1.1.6 ; 1.1.7	saveUser in UserRepo, UserService	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.5	SearchUserById	Luis Bleckmann		1.1.8 ; 1.1.9	GetUserByID Methoden: UserController, UserService, UserRepo	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.6	CreateNewRide	Luis Bleckmann		1.1.16 ; 1.1.17	createDrive Methoden: RideController, RideService, RideRepo	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.7	getUsersCurrentDrive	Luis Bleckmann		1.1.28	GetRideById Methoden: RideController, RideService, RideRepo	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.8	delete drive	Dario Connor Reddig			deleteRideByID Methoden: RideRepo, RideService, RideController	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.9	Token Generator	Dario Connor Reddig			generateRegistrationToken in TokenService	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.10	Profile Picture save	Felix Rieger			RegisterController.java docker-compose.yml	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.11	Driver display	Konstandin Shuti		1.1.6 ; 1.1.8	UserDisplay, UserDisplayRepo, UserDisplayController, UserDisplayService	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.12	Client display	Konstandin Shuti		1.1.7 ; 1.1.9	UserDisplay, UserDisplayRepo, UserDisplayController, UserDisplayService	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.13	eigenes Profil anpassen	Dario Connor Reddig			updateUser Methoden in:	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>

					UserController, UserRepo, UserService	
2.14	aktive Session prüfen	Dario Connor Reddig			UserService.java a Methode: hasActiveSession	Abgeschlossen ▾
3.	<b>Docker</b>					
3.1	Dockerfile	Felix Rieger			all Dockerfile	Abgeschlossen ▾
3.2	docker-compose	Felix Rieger			docker-compose.yml	Abgeschlossen ▾
3.3	nginx config	Felix Rieger			nginx.conf	Abgeschlossen ▾

Anhand der Funktionalitätsplanung werdet Ihr geprüft. Diese sollte dementsprechend **immer** aktuell gehalten werden und pro Funktionalität **nur einen** Verantwortlichen enthalten. Als Quellcodereferenz solltet Ihr immer euer Package, eure Klasse und die dazugehörige Methode angeben. ggf. könnt Ihr auch Zeilenangaben machen.

## Systemtests

<b>Datum</b>	18.05.2025		
<b>Tester</b>	Konstandin Shuti		
<b>SW-Version</b>	Software-Stand: 18.05.2025 Commit: 73c31be7		
<b>Vorbedingung(en)</b>	Der Benutzer ist angemeldet. Der Nutzer „kosh“ ist in der Datenbank vorhanden. Die Suchseite ist geöffnet und bereit für Eingaben.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer gibt den Benutzernamen „kosh“ in das Suchfeld ein.	Das System zeigt eine Liste mit allen User, deren Benutzername “kosh” entspricht oder diesen enthält.	✓
2	Der Benutzer klickt auf “Profil ansehen”.	Das System öffnet die Profilseite und stellt den Benutzer mit alle relevanten Attribute dar.	✓
<b>Nachbedingung(en)</b>	Das Benutzerprofil von „kosh“ wird korrekt auf der Benutzeroberfläche angezeigt.		
<b>Testurteil</b>	<b>Test bestanden.</b>		

<b>Datum</b>	18.05.2025		
<b>Tester</b>	Noel von Stockum		
<b>SW-Version</b>	Software-Stand: 18.05.2025 Commit: 73c31be7		
<b>Vorbedingung(en)</b>	Die User Tabelle der Datenbank ist leer bzw. beinhaltet keine User mit den gewünschten Namen oder E-Mail Adresse. Das Profilbild ist kleiner als 1MB und vom Typ PNG.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer klickt auf "Registrieren".	Das System zeigt die Registrierungs View an.	✓
2	Es werden die gewünschten Eingaben vorgenommen. (E-Mail, Geburtsdatum und Profilbild sind im korrekten Format)	Das System zeigt das Passwort durch „*“-Symbole zensiert an.  Es kann nur entweder "Fahrer" oder "Kunde" angekreuzt werden.	✓
3	Es wird auf "Registrieren" geklickt.	"Ihre Registrierung wird verarbeitet" wird als 'Pop-up' angezeigt, bevor der User an die E-Mail Verifikation weitergeleitet wird.  Es wird eine E-Mail mit dem Token an den vorgegebenen Mail-Client gesendet.	✓
4	Es wird der E-Mail-Token eingetragen.	"Registrierung und Verifikation erfolgreich" wird angezeigt.  validated in der user tabelle wird auf 1 (True) gesetzt.	✓
5	Es wird auf "Zurück zum Login" geklickt.	Das System zeigt wieder die Login View an.	✓
<b>Nachbedingung(en)</b>	Der User ist in der Tabelle wieder zu finden und wurde erfolgreich Registriert.		
<b>Testurteil</b>	<b>Test bestanden.</b>		

<b>Datum</b>	18.05.2025		
<b>Tester</b>	Felix Maximilian Rieger		
<b>SW-Version</b>	Software-Stand: 18.05.2025 Commit: 73c31be7		
<b>Vorbedingung(en)</b>	Ein Nutzer mit dem Benutzernamen "user" und dem Passwort "password"		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<input checked="" type="checkbox"/> / <input type="checkbox"/>
1	Der Benutzer meldet sich mit den Zugangsdaten (user:password) an.	Das System leitet den Nutzer zur Zwei Faktor Eingabe weiter und versendet eine Mail an den Nutzer.	<input checked="" type="checkbox"/>
2	Der Benutzer gibt den Code aus der Mail ein.	Das System leitet den Nutzer auf die Map weiter.	<input checked="" type="checkbox"/>
<b>Nachbedingung(en)</b>	Nutzer ist am System angemeldet.		
<b>Testurteil</b>	<b>Test bestanden.</b>		

<b>Datum</b>	18.05.2025		
<b>Tester</b>	Noel von Stockum		
<b>SW-Version</b>	Software-Stand: 18.05.2025 Commit: 73c31be7		
<b>Vorbedingung(en)</b>	Der User ist erfolgreich registriert, angemeldet und hat auf die "Fahrt Planen"-View navigiert.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Es wird ein Startpunkt und ein Ziel in Form eines POI ("Essen Hbf" und "Duisburg Hbf") eingetragen und auf "Weiter" geklickt.	Nach einer Ladezeit, in welcher die Eingaben durch Koordinaten ersetzt werden, wird die "Ihre Route"-View angezeigt.	✓
2	Es wird auf "Buchen" geklickt, da der Nutzer mit den Eingaben zufrieden ist	Die Daten werden an das Backend übertragen und die Nächste ("Fahranfrage gesendet")-View kann diese korrekt wiedergeben.	✓
3	Es wird auf "Home" in der Navbar geklickt.	Es wird auf die Map-View gewechselt, wo der Nutzer seine Route sowie den Start und Zielpunkt sehen kann.	✓
<b>Nachbedingung(en)</b>	In der drives Tabelle wurde die Fahrt persistent gespeichert.		
<b>Testurteil</b>	<b>Test bestanden.</b>		

# Zyklus II

## Spezifikationsplanung

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
<b>3.</b>	Frontend Planung 2. Zyklus			
3.1.1	Geldkonto Einsehen	User Story	Luis Bleckmann	Abgeschlossen ▾
3.1.2	Geldkonto Kunde Zahlen	User Story	Luis Bleckmann	Abgeschlossen ▾
3.1.3	Geldkonto Fahrer Verdienen	User Story	Luis Bleckmann	Abgeschlossen ▾
3.1.4	Fahranfragen Liste anzeigen	User Story	Luis Bleckmann	Abgeschlossen ▾
3.1.5	Fahranfragen Liste sortieren	User Story	Luis Bleckmann	Abgeschlossen ▾
3.1.6	Fahranfragen aktuelle Position	User Story	Konstandin Shuti	Abgeschlossen ▾
3.1.7	Fahrplanung karte während planung	User Story	Konstandin Shuti	Abgeschlossen ▾
3.1.8	Fahrplanung Distanz	User Story	Felix Rieger	Abgeschlossen ▾
3.1.9	Fahrplanung Dauer	User Story	Felix Rieger	Abgeschlossen ▾
3.1.10	Fahrplanung Preis	User Story	Felix Rieger	Abgeschlossen ▾
3.1.11	Fahrplanung Fahrzeugklassen-Faktor	User Story	Felix Rieger	Abgeschlossen ▾
3.1.12	Fahrtangebote Kunde Benachrichtigung	User Story	Dilay Topal	Abgeschlossen ▾
3.1.13	Fahrtangebote Fahrer maximal 1 Fahranfrage	User Story	Dilay Topal	Abgeschlossen ▾
3.1.14	Fahrtangebote Fahrer zurückziehen	User Story	Dilay Topal	Abgeschlossen ▾
3.1.15	Fahrtangebote Kunde Liste	User Story	Dilay Topal	Abgeschlossen ▾
3.1.16	Fahrt Simulation Karte	User Story	Noel von Stockum	Abgeschlossen ▾
3.1.17	Fahrt Simulation Start & Stop	User Story	Noel von Stockum	Abgeschlossen ▾
3.1.18	Fahrt Simulation Geschwindigkeitsregler	User Story	Noel von Stockum	Abgeschlossen ▾
3.1.19	Fahrt Simulation Synchron	User Story	Noel von Stockum	Abgeschlossen ▾
3.1.20	Fahrt Simulation Fahrt automatisch abschließen	User Story	Konstandin Shuti	Abgeschlossen ▾

3.1.21	Fahrtabschluss Zahlungsabwicklung	User Story	Dario Connor Reddig	Abgeschlossen ▾
3.1.22	Fahrtabschluss Fahrer Bewertung an Kunde	User Story	Konstandin Shuti	Abgeschlossen ▾
3.1.23	Fahrtabschluss Kunde Bewertung an Fahrer	User Story	Konstandin Shuti	Abgeschlossen ▾
3.1.24	Fahrthistorie anzeigen	User Story	Felix Rieger	Abgeschlossen ▾
3.1.25	Fahrthistorie sortieren	User Story	Dario Connor Reddig	Abgeschlossen ▾
3.1.26	Fahrthistorie Kunde, Fahrer am Namen suchen	User Story	Dario Connor Reddig	Abgeschlossen ▾
3.1.27	Fahrthistorie Fahrer, Kunden am Namen suchen	User Story	Dario Connor Reddig	Abgeschlossen ▾
3.2.1	Geldkonten	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
3.2.2	Liste: Fahr Anfragen	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
3.2.3	Fahrtplanung	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
3.2.4	Fahrtangebote	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
3.2.5	Fahrt Durchführung	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
3.2.6	Fahrthistorie	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
3.2.7	Benachrichtigung	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
3.2.8	Bewertung	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
(3.3.1)	Einstellungen	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
(3.3.2)	PopUps / Benachrichtigung	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
<b>4.</b>	Backend Planung 2. Zyklus			
4.1	Datenbankdiagramm	Diagramm	Felix Rieger	Abgeschlossen ▾
4.2	Klassendiagramm	Diagramm	Luis Bleckmann	Abgeschlossen ▾
4.3	Komponentendiagramm	Diagramm	Dario Connor Reddig	Abgeschlossen ▾
4.4	Verhaltensdiagramm Geldkonto	Diagramm	Dilay Topal	Abgeschlossen ▾
4.5	Verhaltensdiagramm Fahrt Fahrer	Diagramm	Noel von Stockum	Abgeschlossen ▾
4.6	Verhaltensdiagramm Fahrt Kunde	Diagramm	Lucy Brandt	Abgeschlossen ▾
4.7	Verhaltensdiagramm Vergangene Fahrt	Diagramm	Konstandin Shuti	Abgeschlossen ▾

## User-Stories

<b>User Story-ID</b>	3.1.1
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich mein Geldkonto einsehen können, um den aktuellen Kontostand und vergangene Transaktionen nachvollziehen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.1; 1.1.2; 1.1.3; 1.1.4

<b>User Story-ID</b>	3.1.2
<b>User Story-Beschreibung</b>	Als Kunde möchte ich mein Geldkonto nutzen können, um eine Fahrt zu bezahlen und sicherzustellen, dass der Betrag korrekt abgebucht wird.
<b>Geschätzter Realisierungsaufwand</b>	2 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.1

<b>User Story-ID</b>	3.1.3
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich Geld auf mein Konto gutgeschrieben bekommen, wenn ich eine Fahrt abschließe, damit ich für meine Dienstleistung entlohnt werde.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.1

<b>User Story-ID</b>	3.1.4
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich eine Liste offener Fahranfragen sehen können, um auszuwählen, welche Fahrt ich annehmen möchte.
<b>Geschätzter Realisierungsaufwand</b>	2 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.26; 1.1.27; 1.1.28; 1.1.29

<b>User Story-ID</b>	3.1.5
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich die Liste der Fahrtenfragen nach Kriterien wie Entfernung, Preis oder Zeit sortieren können, um bessere Entscheidungen zu treffen.
<b>Geschätzter Realisierungsaufwand</b>	3 Tage
<b>Priorität</b>	Mittel
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.4; 3.1.6

<b>User Story-ID</b>	3.1.6
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich meine aktuelle Position angeben können, um eine Fahrt in meiner Nähe annehmen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.4

<b>User Story-ID</b>	3.1.7
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich während der Fahrplanung eine Karte sehen, damit ich Start, Ziel und Route visuell überprüfen kann.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.6; 1.1.10; 1.1.11; 1.1.13

<b>User Story-ID</b>	3.1.8
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich die Distanz der Fahrt sehen können, um den Preis einschätzen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.16; 1.1.17; 1.1.13

<b>User Story-ID</b>	3.1.9
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich die Dauer der Fahrt sehen können, um besser einzuschätzen, wann ich ankommen werde.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.8

<b>User Story-ID</b>	3.1.10
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich den Preis der Fahrt sehen können, um mein Budget zu planen.
<b>Geschätzter Realisierungsaufwand</b>	2 Tage
<b>Priorität</b>	Mittel
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.8

<b>User Story-ID</b>	3.1.11
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich für weniger komfortable Fahrzeugklassen einen angepassten Preis zahlen.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.10

<b>User Story-ID</b>	3.1.12
<b>User Story-Beschreibung</b>	Als Kunde möchte ich bei einer Fahranfrage darüber benachrichtigt werden, dass ein Fahrer meine Fahrt übernehmen möchte, damit ich sehen kann, wer mein Fahrer ist und letztendlich die Fahrt entweder annehme oder ablehne.
<b>Geschätzter Realisierungsaufwand</b>	3 Tage
<b>Priorität</b>	Mittel
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.1, 1.1.2, 1.1.26, 1.1.28, 3.1.4

<b>User Story-ID</b>	3.1.13
<b>User Story-Beschreibung</b>	Als Fahrer kann ich aus der Liste nur eine Fahranfrage auswählen und dem Kunden ein Angebot bieten, damit ich Fahrten übernehmen und gleichzeitig Doppelbuchungen vermeide.
<b>Geschätzter Realisierungsaufwand</b>	3-4 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.1, 1.1.2, 1.1.26, 1.1.27, 3.1.4, 3.1.5

<b>User Story-ID</b>	3.1.14
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich mein ausgewähltes Fahrangebot zurückziehen können, solange es noch nicht vom Kunden angenommen worden ist, damit ich flexibel bleiben kann und sinnvollere, nähere Fahrten übernehme.
<b>Geschätzter Realisierungsaufwand</b>	3 Tage
<b>Priorität</b>	Mittel
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.2, 1.1.29, 3.1.5, 3.1.12, 3.1.13

<b>User Story-ID</b>	3.1.15
<b>User Story-Beschreibung</b>	Als Kunde möchte ich bei einer Fahranfrage mehrere gleichzeitige Angebote von verschiedenen Fahrern, als Tabelle, erhalten und nach Fahrerbewertung etc. informiert werden, damit ich das für mich beste Angebot auswählen kann.
<b>Geschätzter Realisierungsaufwand</b>	3-4 Tage
<b>Priorität</b>	Mittel
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	1.1.2, 1.1.8, 1.1.27, 1.1.28, 3.1.12, 3.1.13

<b>User Story-ID</b>	3.1.16
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich auf der Karte sehen können, wie weit ich mit meiner Fahrt bin.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.8; 1.1.13

<b>User Story-ID</b>	3.1.17
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich jederzeit die Fahrt starten und anhalten können, um ungeplante Stopps mit einzubringen.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.16; 1.1.13

<b>User Story-ID</b>	3.1.18
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich die Geschwindigkeit einstellen können, um die Fahrt an der Realität anzupassen.
<b>Geschätzter Realisierungsaufwand</b>	2 Tage
<b>Priorität</b>	Niedrig
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.16; 3.1.17

<b>User Story-ID</b>	3.1.19
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich, dass die Anzeige der Fahrt so wie die Kontrollelemente mit meinem Fahrer/Kunden synchronisiert ist, damit wir eine Single Source of Truth haben und uns nicht über das Entgelt streiten.
<b>Geschätzter Realisierungsaufwand</b>	3 Tage
<b>Priorität</b>	Hoch
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.16 - 3.1.18; 1.1.14

<b>User Story-ID</b>	3.1.20
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich, dass eine simulierte Fahrt automatisch abgeschlossen wird, sobald das Ziel erreicht ist, damit ich den Abschluss nicht manuell durchführen muss.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Mittel
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.16 - 3.1.19; 1.1.13

<b>User Story-ID</b>	3.1.23
<b>User Story-Beschreibung</b>	Als Kunde möchte ich nach Fahrtabschluss den Fahrer bewerten können, damit ich meine Erfahrung teilen kann.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.20

<b>User Story-ID</b>	3.1.22
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich den Kunden nach der Fahrt bewerten, damit ich meine Erfahrung teilen und das Verhalten des Kunden rückmelden kann.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.20

<b>User Story-ID</b>	3.1.24
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich meine Fahrhistorie sehen können, um meine Transaktionen nachvollziehen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	Hoch
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.20

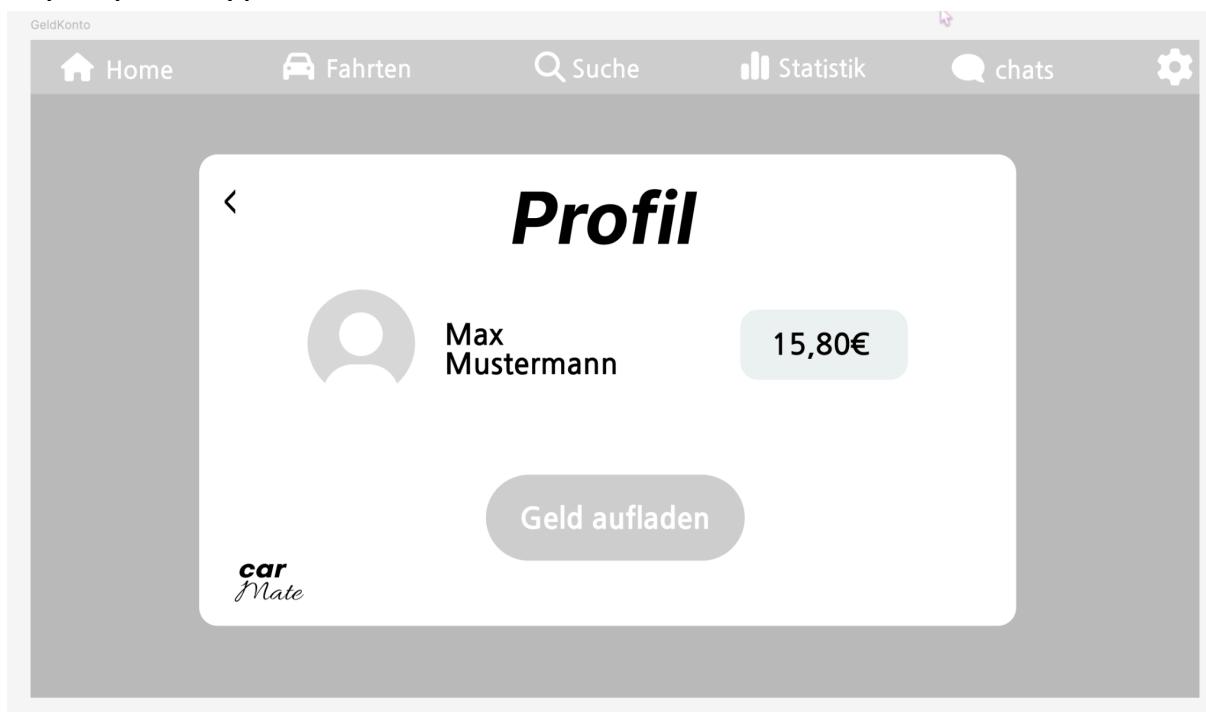
<b>User Story-ID</b>	3.1.21
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich, dass nach einer erfolgreichen Fahrt die Zahlung automatisch abgewickelt wird, damit dies nicht vergessen wird.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	hoch
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.1; 3.1.2; 3.1.3

<b>User Story-ID</b>	3.1.25
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich meine Fahrthistorie anhand von gewissen Kriterien sortieren, um nicht durch alle Fahrten scrollen zu müssen.
<b>Geschätzter Realisierungsaufwand</b>	0,5 Tage
<b>Priorität</b>	niedrig
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.24

<b>User Story-ID</b>	3.1.26
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich in meiner Fahrthistorie anhand eines Fahrernamens suchen können, um nicht durch alle Fahrten scrollen zu müssen.
<b>Geschätzter Realisierungsaufwand</b>	2 Std
<b>Priorität</b>	niedrig
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.24

<b>User Story-ID</b>	3.1.27
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich in meiner Fahrthistorie anhand eines Kundennamens suchen können, um nicht durch alle Fahrten scrollen zu müssen.
<b>Geschätzter Realisierungsaufwand</b>	2 Std
<b>Priorität</b>	niedrig
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	3.1.24

## Papierprototypen



FahrAnfragen (fahrer)

The screenshot shows a table of ride requests with the following columns: Fahrt-ID, Datum-Uhrzeit, Kunde, Bewertung, Fzg. Klasse, Entfernung, Distanz, Dauer, and Preis. One row is highlighted in green with the message "Anfrage Gesendet".

Fahrt-ID	Datum-Uhrzeit	Kunde	Bewertung	Fzg. Klasse	Entfernung	Distanz	Dauer	Preis
1234	02.02.2022 22:00 Uhr	Max muster nutzernname	★★★★★	Mittel	13 km	302 km	1 std. 30 min	15,80€
1234	02.02.2022 22:00 Uhr	Max muster nutzernname	<b>Anfrage Gesendet</b>					

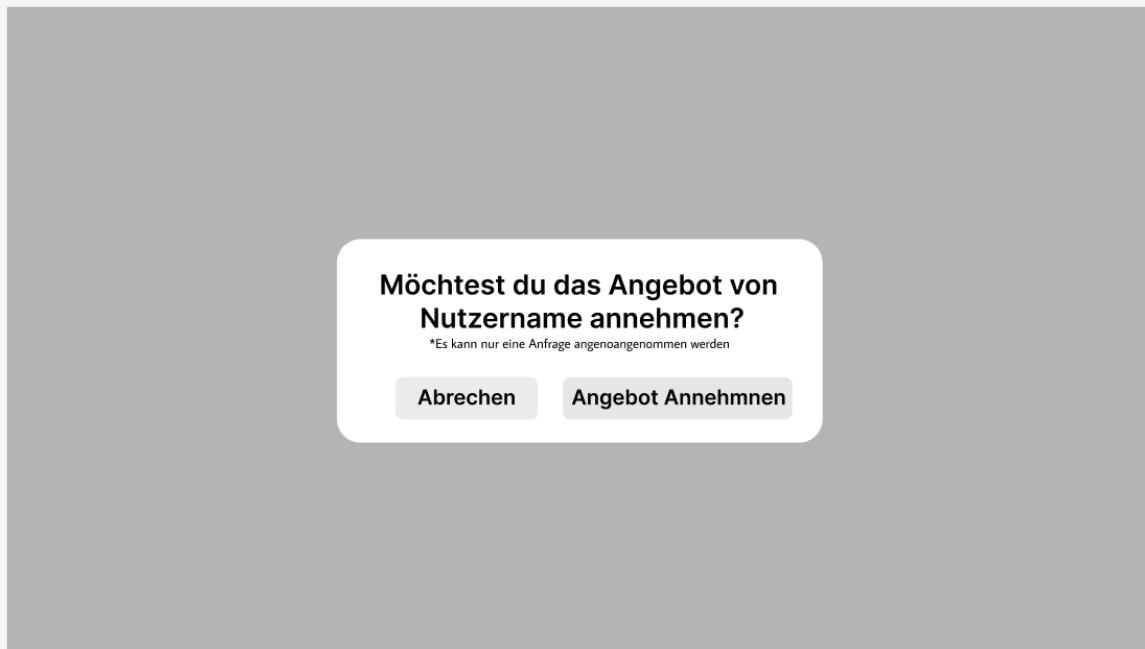
benachrichtigung

Fahrtangebote (kunde)

The screenshot shows three offer cards. Each card includes a red 'X' icon, the driver's name, the number of trips, distance, and a checked checkbox.

	<b>Nutzername</b>	Anzahl Fahrten	Fahrdistanz
	★★★★★	26	360 km
	<input checked="" type="checkbox"/>		
	<b>Nutzername</b>	Anzahl Fahrten	Fahrdistanz
	★★★★★	26	360 km
	<input checked="" type="checkbox"/>		
	<b>Nutzername</b>	Anzahl Fahrten	Fahrdistanz
	★★★★★	26	360 km
	<input checked="" type="checkbox"/>		

PopUp Annehmen



popUp Ablehnen



Fahrtplanung

Home    fahren    Suche    Statistik    Einstellungen

Startpunkt  
Aktueller Standort

+ Zwischenstopp  
Disneyland

Ziel  
McDonalds

Fahrzeug  
Klein

**car Mate**

Suchen

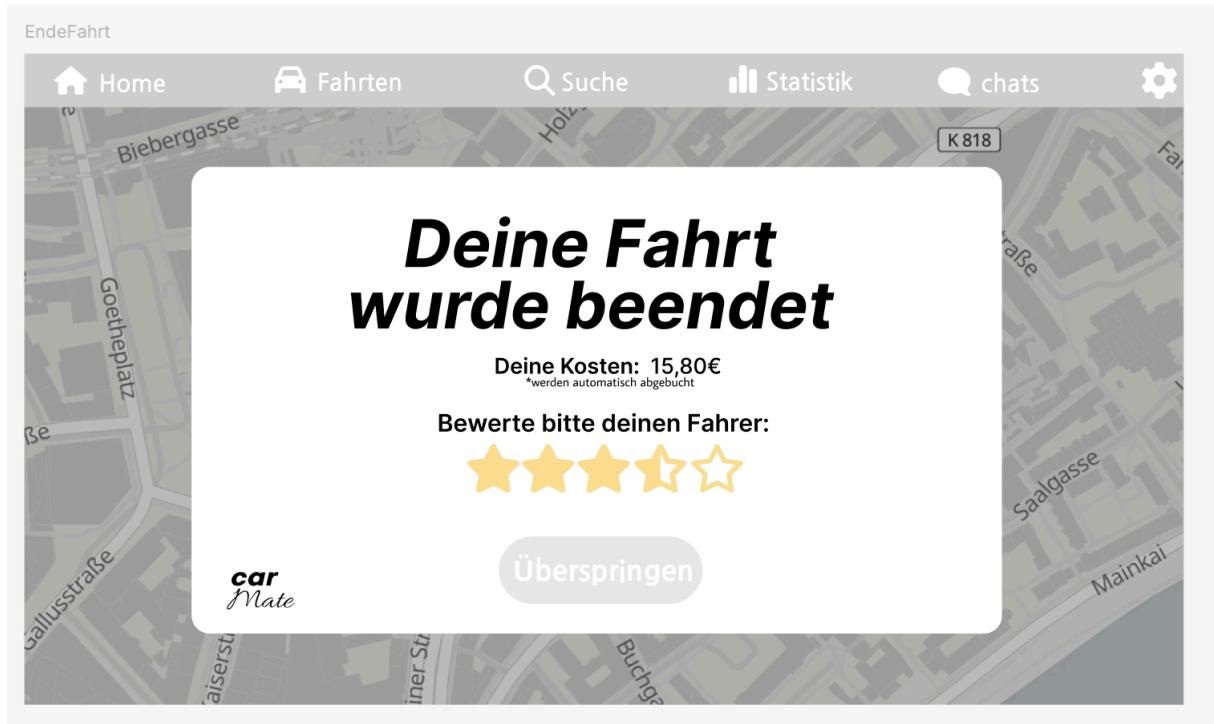
Gesamtdistanz    Gesamtdauer    Erw. Fahrpreis  
24 km    3 Std.    36,80 €

A map of a city area showing a purple route line. The route starts at a car icon, goes north along Bleidenstraße, turns east onto Roßmarkt, and then turns south along Bleidenstraße again to reach a checkered flag icon labeled 'K 818'. Street names like Goetheplatz, Bleidenstraße, and Kornmarkt are visible.

FahrtDurchführung

Home    Fahrten    Suche    Statistik    chats

A map of the same route as the first screenshot, showing the purple line and key locations like McDonalds, Disneyland, and the checkered flag. A control bar at the bottom includes a play/pause button, a timer set to '3 sek', and a progress bar indicating '30 sek' of the total route length.

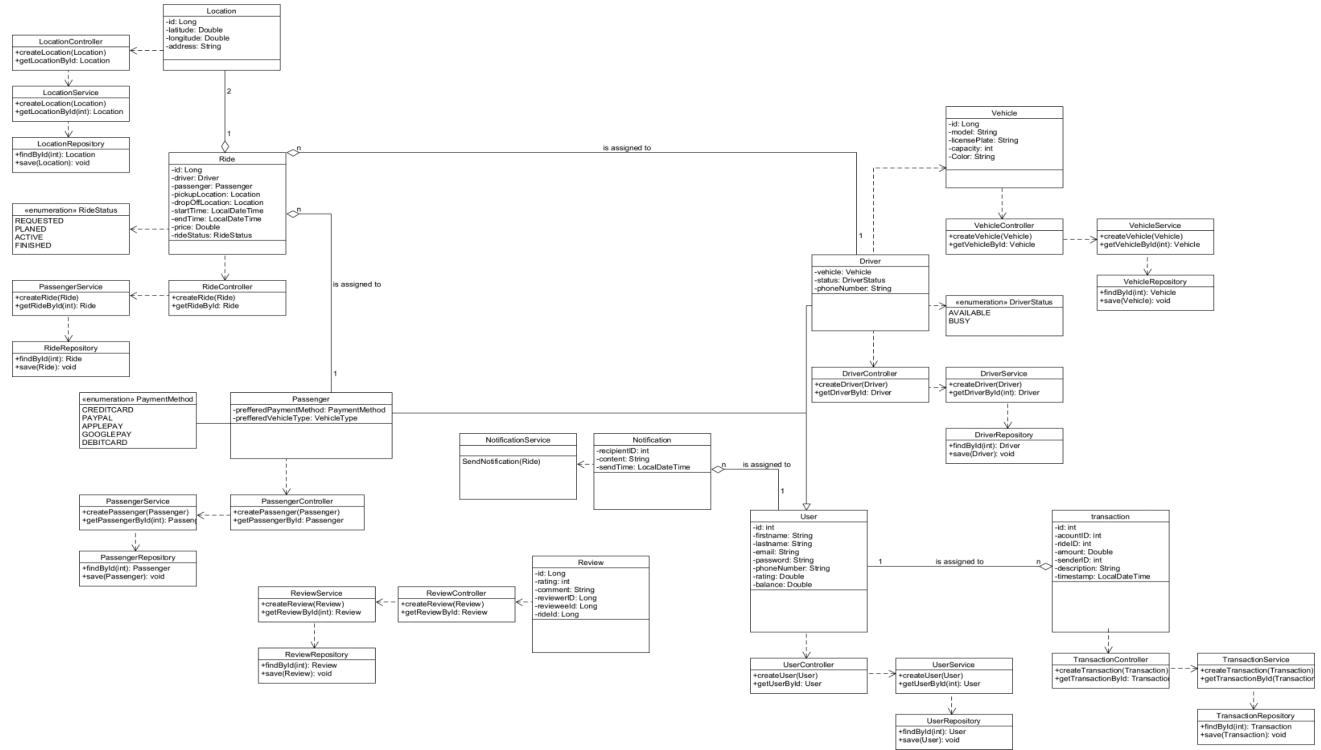


Fahrthistorie								
Fahrt-ID	Kunden	Fahrer	Distanz	Fahrtdauer	Abschluss	Fahrer Rating	Kunden Rating	Betrag
1234	Max muster nutzername	Maxine muster nutzername	302 KM	4 Minuten	02.02.2022 22:00 Uhr	★★★★★	★★★★★	0,20€
4321	Max muster nutzername	Maxine muster nutzername	302 KM	4 Minuten	02.02.2022 22:00 Uhr	★★★★★	★★★★★	0,20€

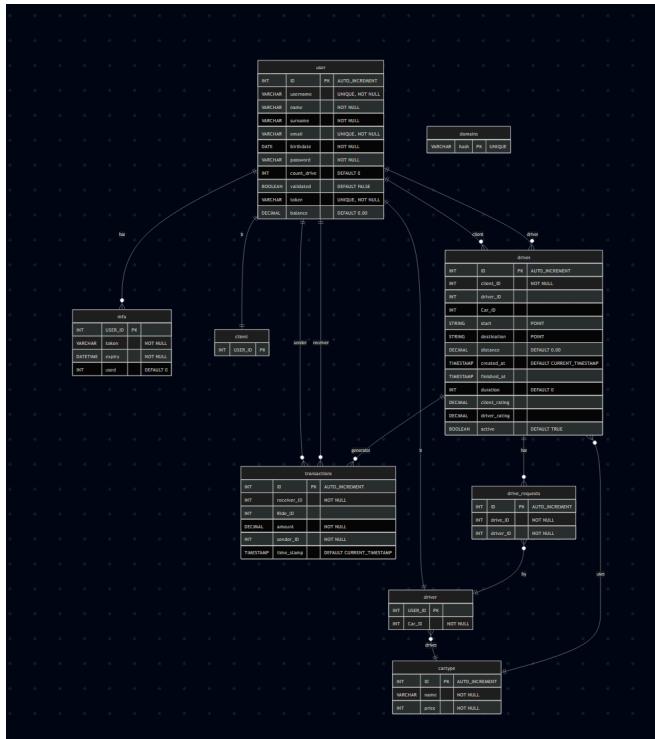


# Strukturdiagramme (Komponenten- und Klassendiagramme)

## UML-Klassendiagramm (Luis Bleckmann):

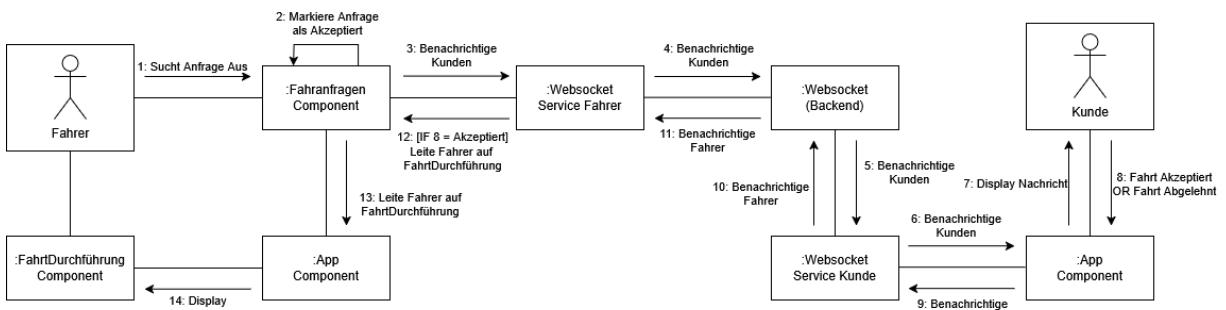


## Datenbankmodell (Felix Maximilian Rieger):

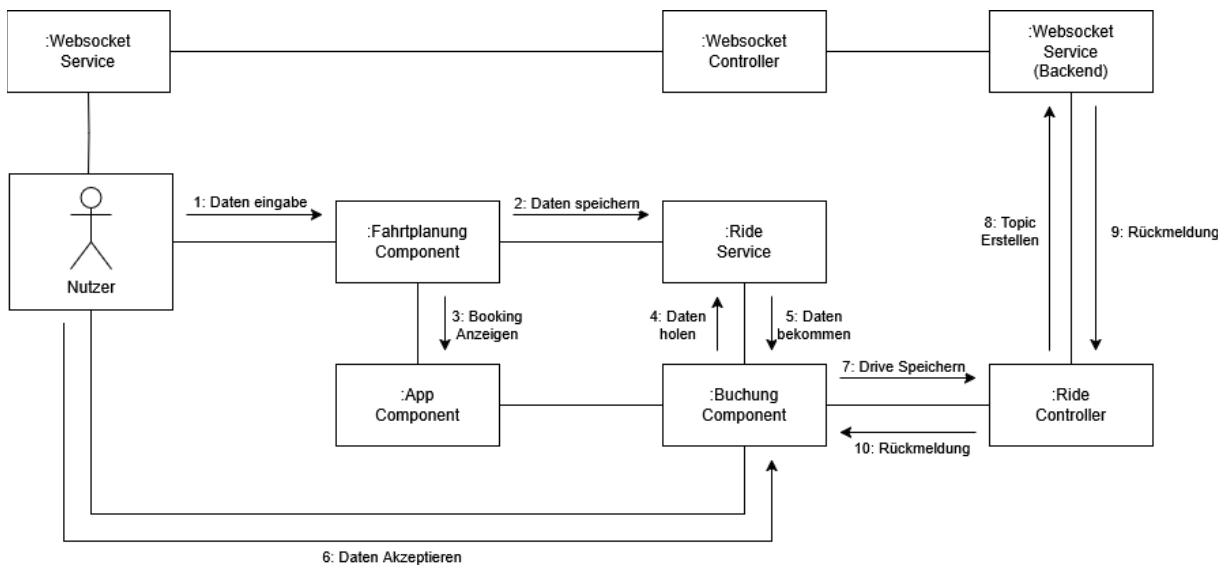


## Verhaltensdiagramme (Kommunikationsdiagramme)

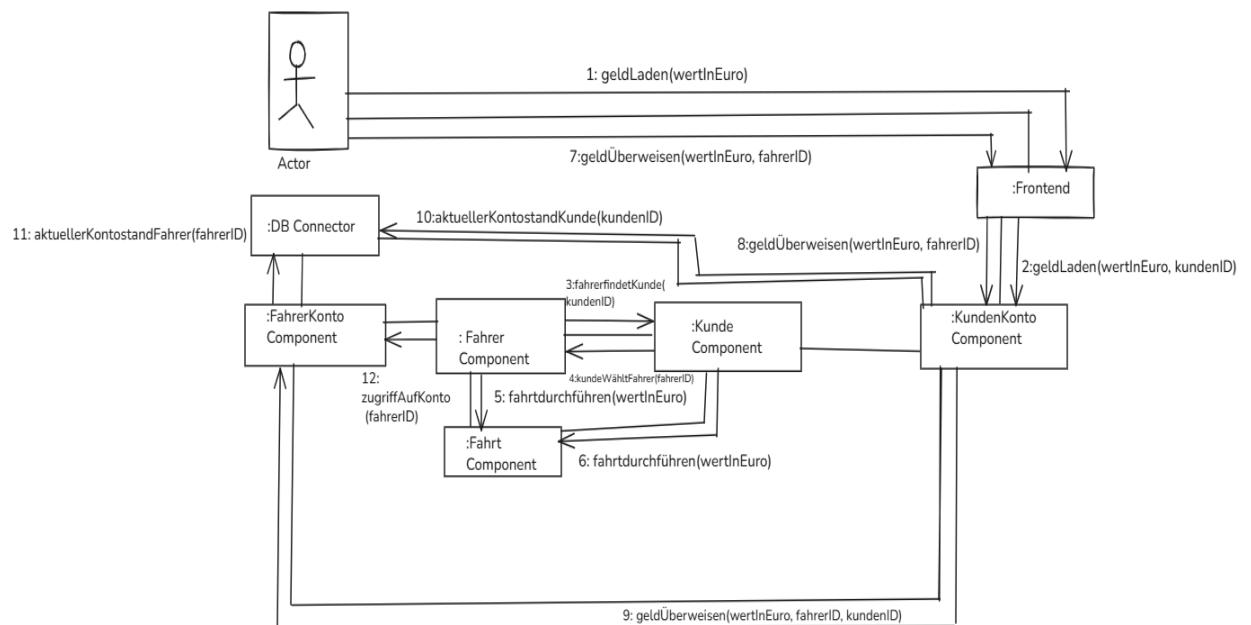
### Verhaltensdiagramm Fahrt Fahrer (Noel von Stockum):



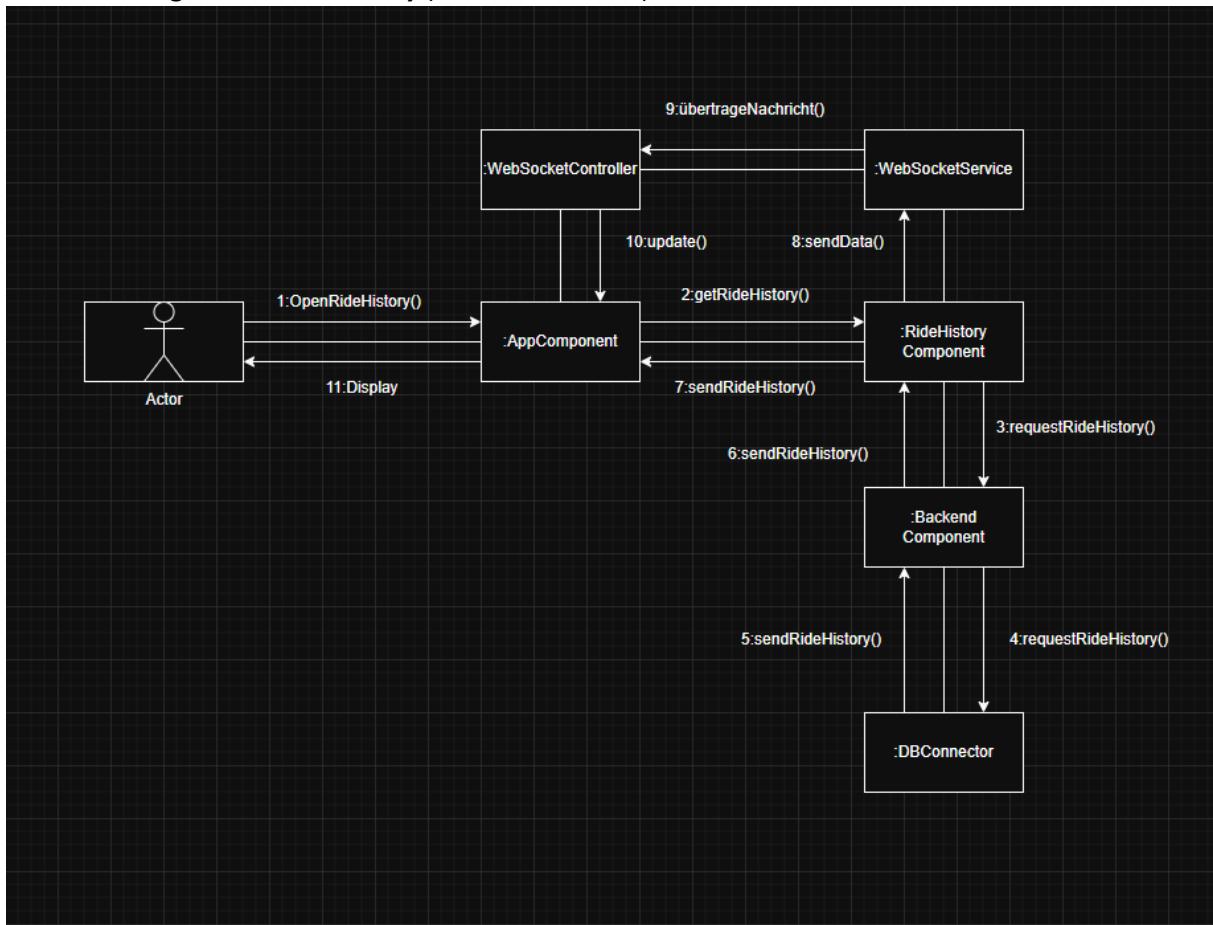
### Verhaltensdiagramm Fahrt Kunde (Lucy Brandt):



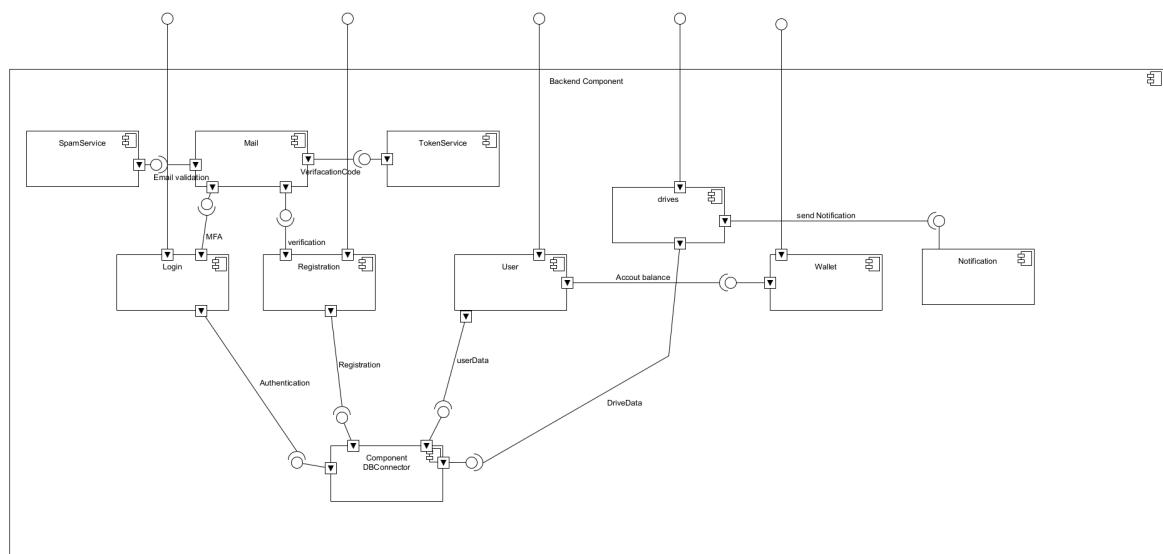
### Verhaltensdiagramm: Geldkonto (Dilay Topal):



### Verhaltensdiagramm: Ride History (Konstantin Shutil):



### Komponentendiagramm (Dario Connor Reddig)



## Funktionalitätsplanung

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode - referenz	Status
1.	<b>Frontend</b>					
1.1	Geldkonten + Aufladen	Lucy Brandt	2.1.1, 2.1.2, 2.1.3, 2.1.4	3.1.1, 3.1.2, 3.1.3	\Frontend\src\app\components\account.component	Abgeschlossen ▾
1.2	Liste: Fahr Anfragen	Dilay Topal	2.2.3, 2.2.4, 2.2.5, 2.2.6	3.1.4, 3.1.5, 3.1.6, 3.1.13, 3.1.14	\Frontend\src\app\components\driverequests.component	Abgeschlossen ▾
1.3	Fahrtplanung	Noel von Stockum	2.2.2	3.1.6, 3.1.7, 3.1.8, 3.1.9, 3.1.10	\Frontend\src\app\components\drivedata.component; booking.component; current-drive.component	Abgeschlossen ▾
1.4	Fahrtangebote	Lucy Brandt	2.2.8, 2.2.9, 2.2.10	3.1.12, 3.1.15,	\Frontend\src\app\components\driveoffercostumer.component	Abgeschlossen ▾
1.5	Fahrt Durchführung	Noel von Stockum	2.3.1	3.1.16, 3.1.17, 3.1.18, 3.1.19, 3.1.20	\Frontend\src\app\components\map.component	Abgeschlossen ▾
1.6	Fahrhistorie	Dilay Topal	2.4.2	3.1.24, 3.1.25, 3.1.26, 3.1.27	\Frontend\src\app\components\history.component	Abgeschlossen ▾
1.7.1	Benachrichtigung	Noel von Stockum	1.10, 2.2.5	3.1.12	\Frontend\src\app\	Abgeschlossen ▾

					app.component	
1.7.2	PopUps	Luis Bleckmann		-	\Frontend\src\app\components\popups.component; offer-dialog-accept.component; offer-dialog-reject.component	Abgeschlossen ▾
1.8	Bewertung	Dilay Topal	2.3.3	3.1.22, 3.1.23	\Frontend\src\app\components\rating.component	Abgeschlossen ▾
1.9	(Einstellungen)	Lucy Brandt	-		\Frontend\src\app\components\settings.component	Abgeschlossen ▾
1.10	Websocket Service	Noel von Stockum		3.1.12	\Frontend\src\app\services\websocket.service.ts	Abgeschlossen ▾
1.11	AppComponent	Noel von Stockum			\Frontend\src\app\components\app.component	Abgeschlossen ▾
<b>2.</b>	<b>Backend</b>					
2.1.1	(Kunde) Geldkonto aufladen	Luis Bleckmann	2.1.4	-	UserController:addBalance, UserService:addBalance, UserRepo:addBalance	Abgeschlossen ▾
2.1.2	(Kunde) Geldkonto wird zur Bezahlung genutzt	Luis Bleckmann	2.1.4	3.1.2	Rideservice:payforRide, RideRepo:recordTransaction	Abgeschlossen ▾

2.1.3	Nutzer können ihren Kontostand einsehen	Luis Bleckmann	2.1.4	3.1.1	UserController:getBalance, UserService:getbalance, UserRepo:getBalance	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.1.4	Nutzer Besitzen ein Geldkonto	Luis Bleckmann		3.1.1	NaN	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.1.5	Fahrer wird für seine Fahrt entlohnt	Luis Bleckmann		3.1.3	Rideservice:payforRide, RideRepo:recordTransaction	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.2.2	Berechnung der Distanz der Fahrt, der erwarteten Fahrtdauer und des Preis'	Dario Connor Reddig		3.1.8, 3.1.9, 3.1.10	RideService:fahrpreisErmitteln, RideService:distanceAndDuration	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.2.3	(Fahrer) Sortierung der Fahranfragen	Konstandin Shuti		3.1.5	DriveReqController.java, DriveReqService.java, DriveReqRepo.java	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.2.4	Fahrer kann (maximal) eine Fahranfrage annehmen	Felix Rieger		3.1.13		<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.2.5	Kunde wird über die angenommene Fahranfrage benachrichtigt (websocket)	Felix Rieger		3.1.12	NotificationHandler.java, RideService:hasRideRequests, RideRepo:hasRideRequests	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
2.2.6	Fahrer kann sein Angebot zurückziehen, solange es nicht vom Kunden	Felix Rieger		3.1.14	RideController:deleteOffer, RideService:deleteDr	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>

	angenommen wurde				iverOffer, RideRepo: deleteDriverOffer	
2.3.1	Kunde und Fahrer fahrdaten für die Karte werden Synchronisiert (Websocket)	Felix Rieger		3.1.19	RideWebS ockethand ler.java,RideCache.j ava, RideRepo: acceptOff er	Abgeschlossen ▾
2.3.2	Nach Abschluss werden die Konten verrechnet	Konstandin Shuti		3.1.21	UserServic e:verRech neKonten, UserRepo: verrechne Konten	Abgeschlossen ▾
2.3.3	Fahrer und Kunde können sich nach der Fahrt gegenseitig bewerten	Dario Connor Reddig		3.1.22, 3.1.23	RideContr olle:rateR ide,RideSe rvice:rate User, RideRepo: rateUser	Abgeschlossen ▾
2.4.2	Fahrer und Kunde können ihre Fahrhistorie einsehen	Felix Rieger		3.1.24, 3.1.25	RideContr olle:getHi story, Rideservic e:getHisto ryByID, RideRepo: getHistory ByUserna me	Abgeschlossen ▾
2.2.8	Kunde kann eine Liste der Fahrangebote einsehen	Felix Rieger		3.1.15	RideContr olle:getO ffers, RideServic e:getOffer sByUser, RideRepo: getOffersB yUser	Abgeschlossen ▾
2.2.9	Kunde kann Fahrangebot annehmen	Felix Rieger		3.1.12	RideContr olle:acce ptOffer, RideServic e:acceptO ffer,RideR epo:accep tOffer	Abgeschlossen ▾

2.2.1 0	Kunde kann Fahrangebot ablehnen	Felix Rieger		3.1.12	RideContr oller:delet Offer	Abgeschlossen ▾
------------	---------------------------------------	--------------	--	--------	-----------------------------------	-----------------

## 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	Preisberechnung	preis1	passed
2	Preisberechnung	preis2	passed
3	Preisberechnung	preis3	passed
4	Auszahlen	auszahlen	passed

## Systemtests

Datum	16.06.2025		
Tester	Konstandin Shuti		
SW-Version			
Vorbedingung(en)	Der Benutzer ist angemeldet. Fahrtanfragen existieren und können nach verschiedenen Kriterien sortiert werden.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Fahrer ruft Fahrtanfragen auf	Das System liefert eine Liste aller verfügbaren Fahrtanfragen zurück.	✓
2	Fahrer sortiert die Liste nach Kriterien	Die Liste der Fahrtanfragen wird entsprechend dem gewählten Kriterium sortiert und dem Fahrer aktualisiert angezeigt.	✓
Nachbedingung(en)	Dem Benutzer wird die sortierte Liste der Fahrtanfragen entsprechend seiner Auswahl korrekt angezeigt.		
Testurteil	Test bestanden		

<b>Datum</b>	14.06.2025		
<b>Tester</b>	Felix Rieger		
<b>SW-Version</b>			
<b>Vorbedingung(en)</b>	Der Benutzer ist angemeldet. Abgeschlossene Fahrten für den Nutzer existieren.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Nutzer ruft Fahrthistorie auf	Das System liefert eine Liste aller abgeschlossenen Fahrten zurück.	✓
2	Fahrer sortiert die Liste nach Kriterien	Die Fahrthistorie wird entsprechend dem gewählten Kriterium sortiert und dem Nutzer in aktualisierter Form angezeigt.	✓
<b>Nachbedingung(en)</b>	Dem Benutzer wird die sortierte Liste der Fahrthistorie entsprechend seiner Auswahl korrekt angezeigt.		
<b>Testurteil</b>	Test bestanden		

<b>Datum</b>	14.06.2025		
<b>Tester</b>	Felix Rieger		
<b>SW-Version</b>			
<b>Vorbedingung(en)</b>	Der Kunde ist angemeldet. Fahrtangebote für seine Fahrt existieren.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Kunde ruft Fahrtangebote auf	Das System liefert eine Liste aller verfügbaren Fahrtangebote für den Kunden zurück.	✓
2	Kunde sortiert die Liste nach Kriterien	Die Liste der Fahrtangebote wird entsprechend dem gewählten Kriterium sortiert und dem Kunden aktualisiert angezeigt.	✓
3	Kunde lehnt das Angebot ab	Das System löscht das Angebot und zeigt die aktualisierte View an.	✓
4	Kunde nimmt ein Angebot an	Kunde und Fahrer werden auf die Karte weitergeleitet. Das System aktualisiert die Datenbank.	✓
<b>Nachbedingung(en)</b>	Kunde und Fahrer sind auf der Kartenansicht.		
<b>Testurteil</b>	Test bestanden		

<b>Datum</b>	14.06.2025		
<b>Tester</b>	Felix Rieger		
<b>SW-Version</b>			
<b>Vorbedingung(en)</b>	Der Nutzer ist angemeldet. Eine durchgeführte Fahrt existiert .		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Der Nutzer klickt die passende Bewertung an.	Die entsprechende Anzahl an Sternen wird dargestellt.	✓
2	Der Nutzer schickt die Bewertung ab.	Der Nutzer wird auf Home weitergeleitet. Das System speichert die Bewertung, wenn beide bewertet haben, wird die Fahrt als fertig markiert.	✓
<b>Nachbedingung(en)</b>	Kunde und Fahrer befinden sich auf der Startansicht. Die Fahrt wird in der Fahrthistorie angezeigt.		
<b>Testurteil</b>	Test bestanden		

<b>Datum</b>	14.06.2025		
<b>Tester</b>	Felix Rieger		
<b>SW-Version</b>			
<b>Vorbedingung(en)</b>	Der Nutzer ist angemeldet. Der Nutzer befindet sich auf der Karte und hat eine aktive Fahrt.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Nutzer klickt auf Start.	Die Fahrt startet bei Fahrer und Kunde.	✓
2	Nutzer klickt auf Pause.	Die Fahrt wird bei beiden Nutzer pausiert und der Marker befindet sich an derselben Stelle.	✓
3	Nutzer startet die Fahrt wieder.	Die Fahrt läuft bei Fahrer und Kunde weiter.	✓
4	Kunde beendet die Fahrt.	Die Fahrt wird beendet und die Nutzer werden auf die Bewertung weitergeleitet. Das System aktualisiert die Datenbank.	✓
<b>Nachbedingung(en)</b>	Der Kunde und der Fahrer werden auf die Bewertung weitergeleitet.		
<b>Testurteil</b>	Test bestanden		

# Zyklus III

## Spezifikationsplanung

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
5.	Frontend Planung 3. Zyklus			
5.1.1	Chat Austausch	User Story	Luis Bleckmann	Abgeschlossen ▾
5.1.2	Chat Echtzeit	User Story	Luis Bleckmann	Abgeschlossen ▾
5.1.3	Chat Löschen	User Story	Luis Bleckmann	Abgeschlossen ▾
5.1.4	Chat Bearbeiten	User Story	Dilay Topal	Abgeschlossen ▾
5.1.5	Chat Lesebestätigung	User Story	Dilay Topal	Abgeschlossen ▾
5.1.6	Fahrplanung Zwischenstopps	User Story	Dilay Topal	Abgeschlossen ▾
5.1.7	Statistiken Einnahmen	User Story	Dario Connor Reddig	Abgeschlossen ▾
5.1.8	Statistiken Distanz	User Story	Felix Rieger	Abgeschlossen ▾
5.1.9	Statistiken Fahrzeit	User Story	Felix Rieger	Abgeschlossen ▾
5.1.10	Statistiken Bewertung	User Story	Felix Rieger	Abgeschlossen ▾
5.1.11	Statistiken Monatlich	User Story	Konstandin Shuti	Abgeschlossen ▾
5.1.12	Statistiken Jährlich	User Story	Konstandin Shuti	Abgeschlossen ▾
5.1.13	Live-Änderung Option bei Pause	User Story	Dario Connor Reddig	Abgeschlossen ▾
5.1.14	Live-Änderung Keine Weiterfahrt	User Story	Noel von Stockum	Abgeschlossen ▾
5.1.15	Live-Änderung Daten Neuberechnung	User Story	Noel von Stockum	Abgeschlossen ▾
5.1.16	Live-Änderung Synchronisierung	User Story	Noel von Stockum	Abgeschlossen ▾
5.1.17	Leaderboard Ansicht	User Story	Dario Connor Reddig	Abgeschlossen ▾
5.1.18	Leaderboard Sortierung	User Story	Konstandin Shuti	Abgeschlossen ▾
5.2.1	Chat	Papierprototyp	Lucy Brandt	Abgeschlossen ▾
5.2.2	Fahrplanung	Papierprototyp	Lucy Brandt	Abgeschlossen ▾

5.2.3	Live-Änderungen	Papierprototyp	Lucy Brandt	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
5.2.4	Statistiken	Papierprototyp	Lucy Brandt	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
5.2.5	Leaderboard	Papierprototyp	Lucy Brandt	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
<b>6.</b>	Backend Planung 3. Zyklus			
6.1	Datenbankdiagramm	Diagramm	Felix Rieger	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
6.2	Klassendiagramm	Diagramm	Luis Bleckmann	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
6.3	Komponentendiagramm	Diagramm	Dario Connor Reddig	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
6.4	Verhaltensdiagramm Chat	Diagramm	Konstandin Shuti	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
6.5	Verhaltensdiagramm Live-Änderung	Diagramm	Noel von Stockum	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
6.6	Verhaltensdiagramm Zwischenstopps Fahrplan	Diagramm	Dilay Topal	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
6.7	Verhaltensdiagramm Statistiken	Diagramm	Lucy Brandt	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
6.8	Verhaltensdiagramm Leaderboard	Diagramm	Lucy Brandt	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>

## User-Stories

<b>User Story-ID</b>	5.1.1
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich mit anderen Nutzern im System Nachrichten austauschen können, um eine direkte Kommunikation während der Nutzung der Plattform zu ermöglichen.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	5.1.2
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich Nachrichten in Echtzeit empfangen und senden können, damit Unterhaltungen ohne Verzögerung geführt werden können und die Kommunikation flüssig verläuft.
<b>Geschätzter Realisierungsaufwand</b>	2 Tage
<b>Priorität</b>	hoch
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.1

<b>User Story-ID</b>	5.1.3
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich gesendete Nachrichten löschen können, um meine Konversationen nachträglich bereinigen oder fehlerhafte Nachrichten entfernen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Luis Bleckmann
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.1

<b>User Story-ID</b>	5.1.11
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich monatliche Statistiken zu meinen Fahrten sehen, um meine Nutzung und Ausgaben besser nachverfolgen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.7, 5.1.8, 5.1.9, 5.1.10

<b>User Story-ID</b>	5.1.12
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich jährliche Statistiken zu meinen Fahrten sehen, um meine Nutzung und Ausgaben besser nachverfolgen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.7, 5.1.8, 5.1.9, 5.1.10

<b>User Story-ID</b>	5.1.18
<b>User Story-Beschreibung</b>	Als Nutzer*in möchte ich das Leaderboard sortieren können, um relevante Platzierungen einfacher zu sehen.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Konstandin Shuti
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.17

<b>User Story-ID</b>	5.1.8
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich gefahrene Distanzen einsehen können, um die Nutzung meines Fahrzeuges besser einschätzen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	niedrig
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	5.1.10
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich meine Bewertungen einsehen können, um mein Verhalten anpassen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	niedrig
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	5.1.9
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich meine gefahrene Zeit einsehen können, um meinen Tag besser planen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	niedrig
<b>Autor</b>	Felix Rieger
<b>Abhängigkeiten zu anderen User Stories</b>	

<b>User Story-ID</b>	5.1.14
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich, dass während meiner Änderungen andere Nutzer die Fahrt nicht wieder starten oder selbst Änderungen machen können, um konsistente Daten zu haben.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.13

<b>User Story-ID</b>	5.1.15
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich, dass nach meinen Änderungen die Route neu berechnet wird, um auch nur für das zu Zahlen, was wirklich gefahren wurde und die neue Route nachverfolgen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.13

<b>User Story-ID</b>	5.1.16
<b>User Story-Beschreibung</b>	Als Nutzer möchte ich, dass nach meinen Änderungen andere Nutzer die gleichen Daten haben, um danach auch mögliche Änderungen von deren Seite einpflegen zu können.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Noel von Stockum
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.13

<b>User Story-ID</b>	5.1.7
<b>User Story-Beschreibung</b>	Als Fahrer möchte ich Statistiken über meine Einnahmen bei CarMate sehen, damit ich sehen kann, ob es für mich profitabel ist.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	mittel
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.8, 5.1.9, 5.1.10, 5.1.11, 5.1.12,

<b>User Story-ID</b>	5.1.13
<b>User Story-Beschreibung</b>	Als Fahrgäst möchte ich Änderungen an der Route vornehmen können, wenn ich die Fahr pausiert habe
<b>Geschätzter Realisierungsaufwand</b>	1,5 Tage
<b>Priorität</b>	niedrig
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.14, 5.1.15, 5.1.16,

<b>User Story-ID</b>	5.1.17
<b>User Story-Beschreibung</b>	Als Fahrer oder Fahrgäst möchte ich ein Leaderboard für alle Fahrer als Tabelle sehen, damit ich die besten Fahrer sehen kann
<b>Geschätzter Realisierungsaufwand</b>	0,5 Tage
<b>Priorität</b>	niedrig
<b>Autor</b>	Dario Connor Reddig
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.18,

<b>User Story-ID</b>	5.1.4
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich gesendete Nachrichten im Chat bearbeiten können, solange der Empfänger die Nachricht noch nicht gelesen hat, damit ich Fehler und falsche Informationen korrigieren kann.
<b>Geschätzter Realisierungsaufwand</b>	1 Tag
<b>Priorität</b>	hoch
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.1, 5.1.2, 5.1.5

<b>User Story-ID</b>	5.1.5
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich eine Lesebestätigung für gesendete Nachrichten erhalten, damit ich erkennen kann, ob der Empfänger meine Nachricht gelesen hat.
<b>Geschätzter Realisierungsaufwand</b>	0,5 Tage
<b>Priorität</b>	hoch
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.1, 5.1.2, 5.1.3, 5.1.4

<b>User Story-ID</b>	5.1.6
<b>User Story-Beschreibung</b>	Als Kunde möchte ich beim Erstellen einer Fahranfrage zusätzliche Zwischenstopps zwischen Start- und Zielpunkt hinzufügen können, damit meine Route individuelle Bedürfnisse berücksichtigt
<b>Geschätzter Realisierungsaufwand</b>	2-3 Tage
<b>Priorität</b>	hoch
<b>Autor</b>	Dilay Topal
<b>Abhängigkeiten zu anderen User Stories</b>	5.1.14, 5.1.15

# Papierprototypen

The screenshot displays two main sections of the car Mate app.

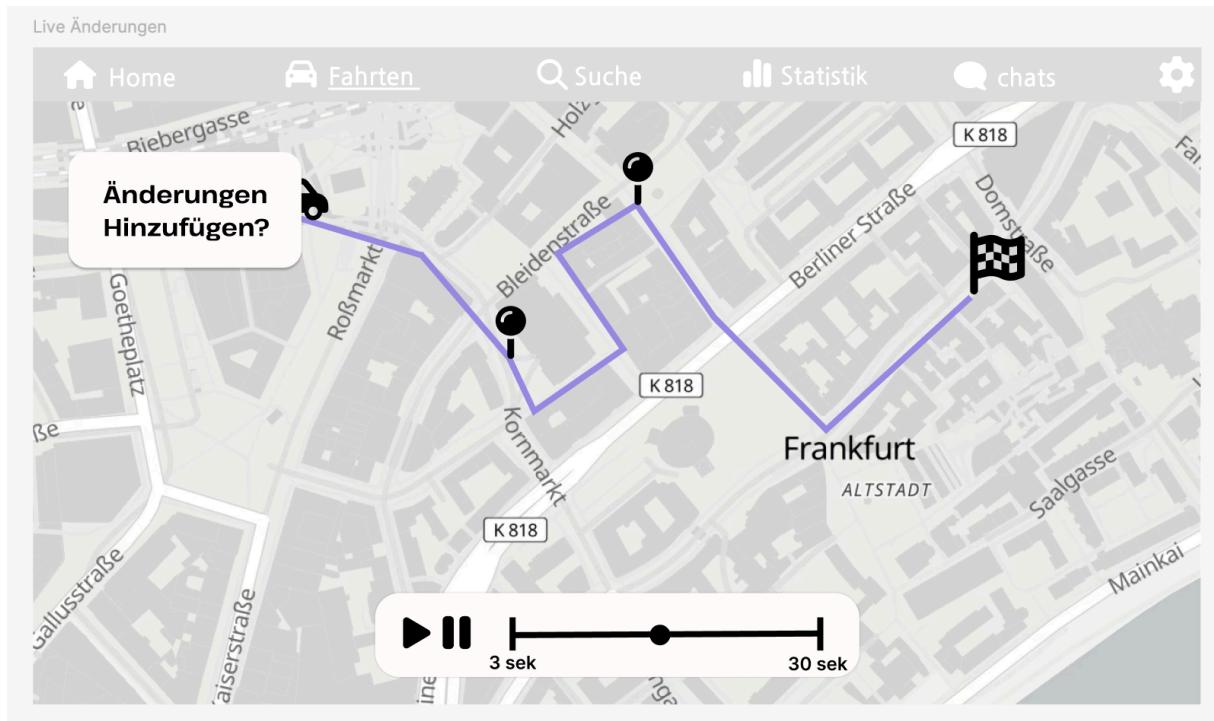
**Top Section (Chat Screen):**

- Header:** Chat, Home, Fahrten, Suche, Statistik, chats, Einstellungen.
- Left Column (Chats):** A list of four entries, each with a person icon and the word "Name".
- Right Column (User Profile):** Displays the user's name "Nutzername" and a message bubble containing "bla bla bla". The timestamp "13:30" is shown at the bottom right of the message area.

**Bottom Section (Fahrplanung Screen):**

- Header:** Fahrplanung, Home, Fahrten, Suche, Statistik, Einstellungen.
- Input Fields:**
  - Startpunkt: Aktueller Standort
  - + Zwischenstopp: Disneyland
  - Ziel: McDonalds
  - Fahrzeug: Klein
- Map:** A map showing a driving route marked with a purple line, starting from the user's current location and ending at McDonald's. The route passes through several streets including Goetheplatz, Rossmarkt, and Kommtstrasse. A small flag icon with "K 818" is visible on the map.
- Summary:** Shows the total distance, duration, and cost of the trip.

Gesamtdistanz	Gesamtdauer	Erw. Fahrtzeit
24 km	3 Std.	36,80 €
- Car Mate Logo:** Located at the bottom left.

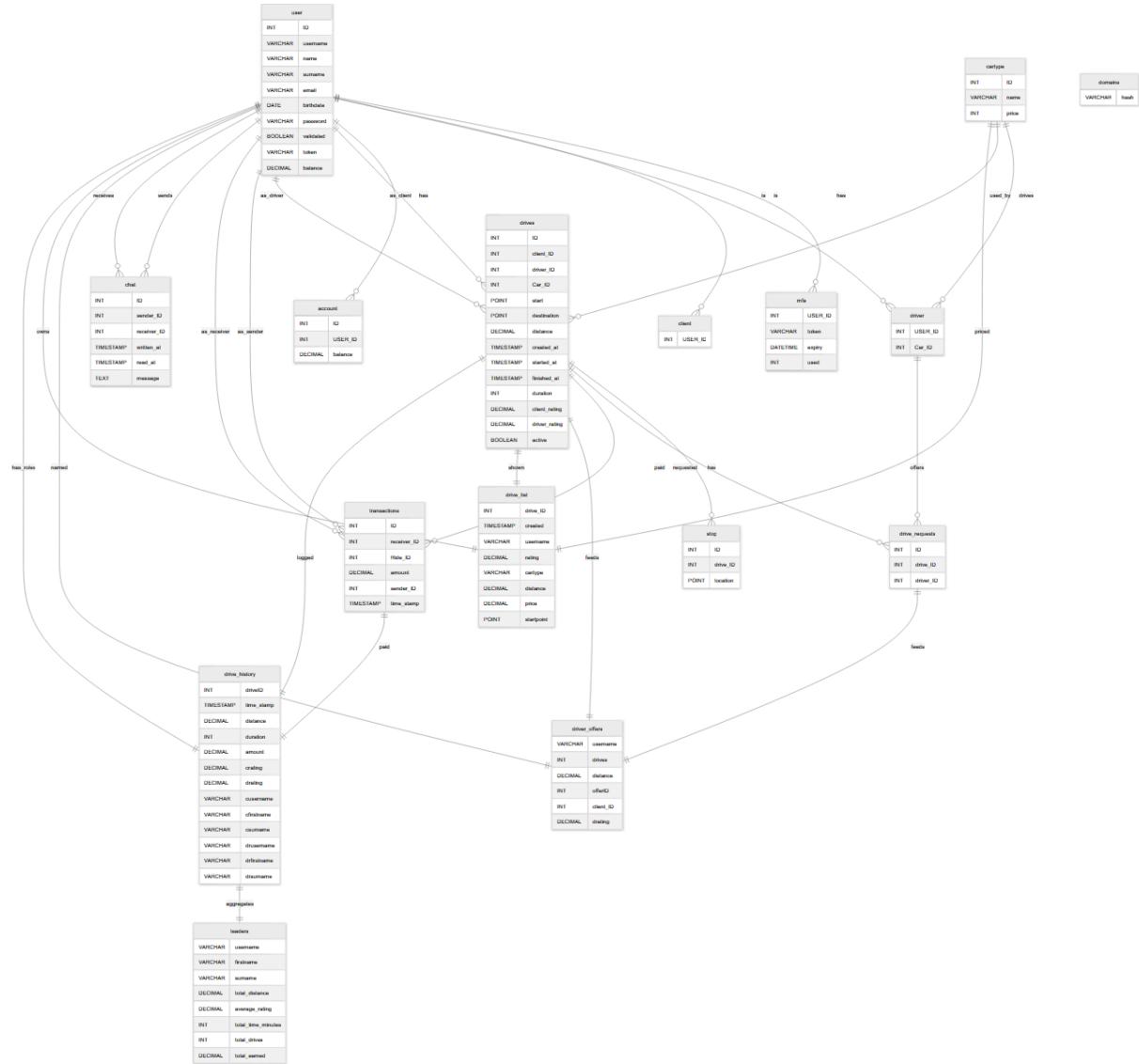


Fahrer Leaderboard

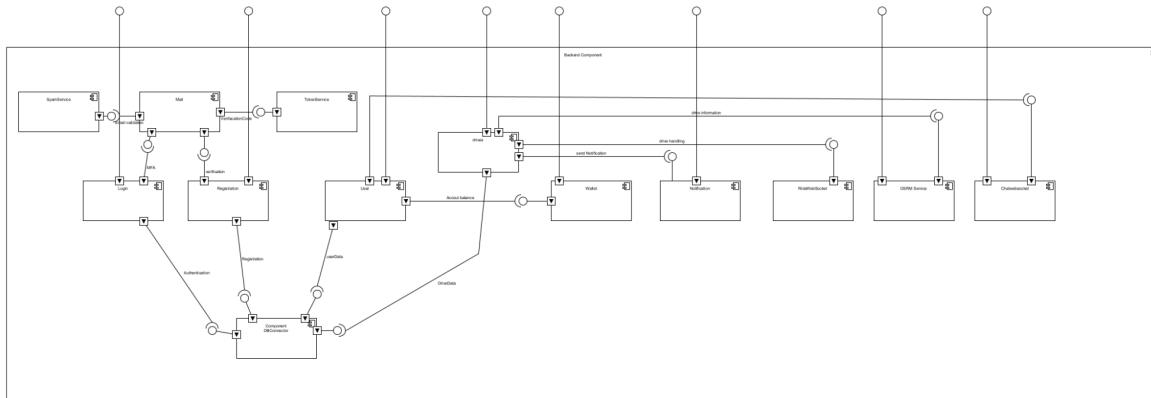
Nutzername	Fahrer	Distanz	Fahrtzeit	Anzahl Fahrten	Bewertung	Betrag
nutzername	Maxine muster	302 KM	4 Minuten	21	★★★★★☆	0,20€
nutzername	Maxine muster	302 KM	4 Minuten	21	★★★★★☆	0,20€

# Strukturdiagramme (Komponenten- und Klassendiagramme)

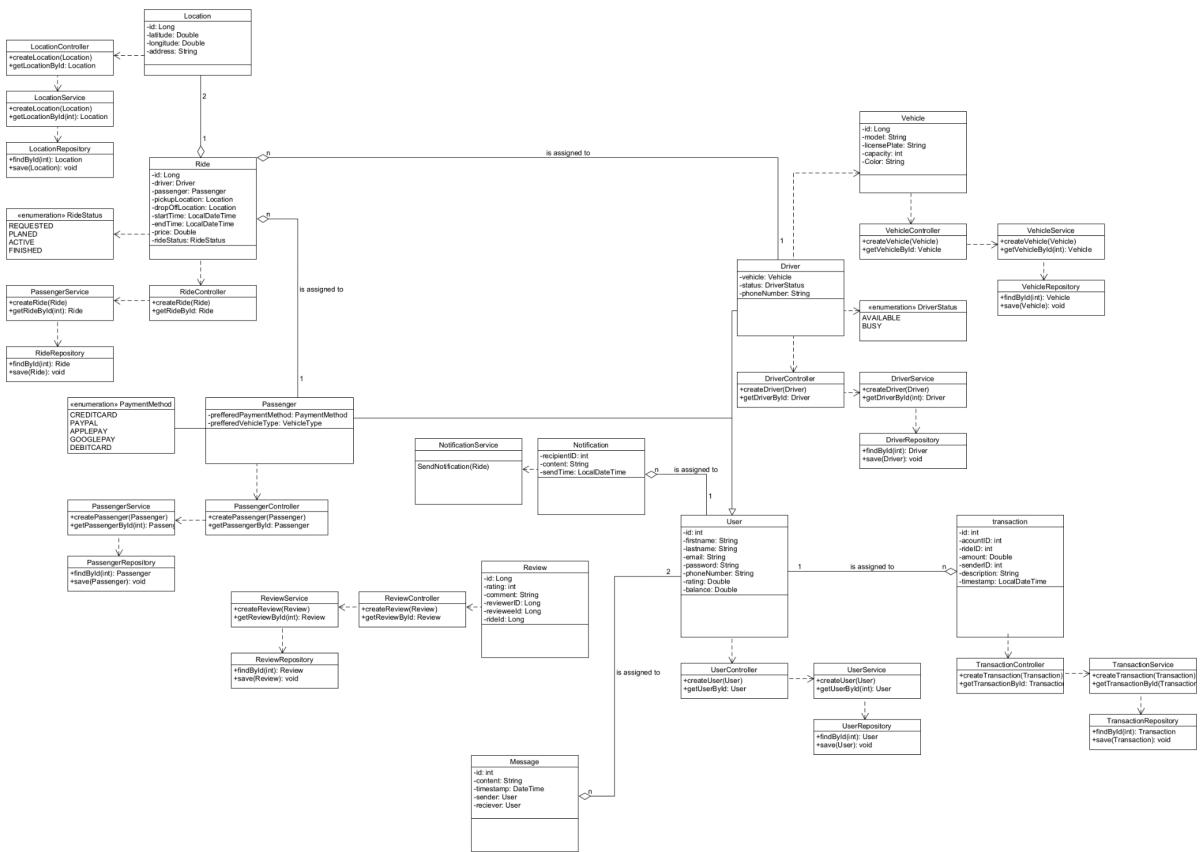
Datenbankdiagramm (Felix Rieger):



## Komponentendiagramm (Dario Connor Reddig):

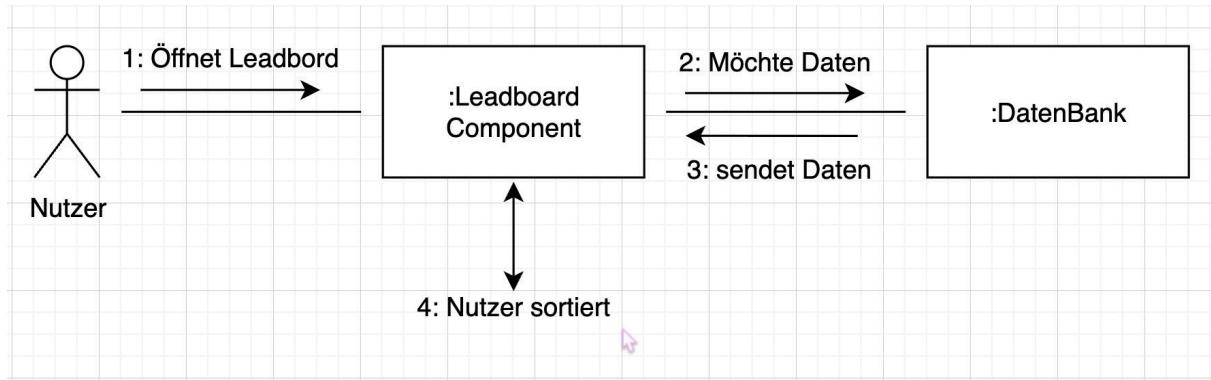


## Klassen Diagramm (Luis Bleckmann):

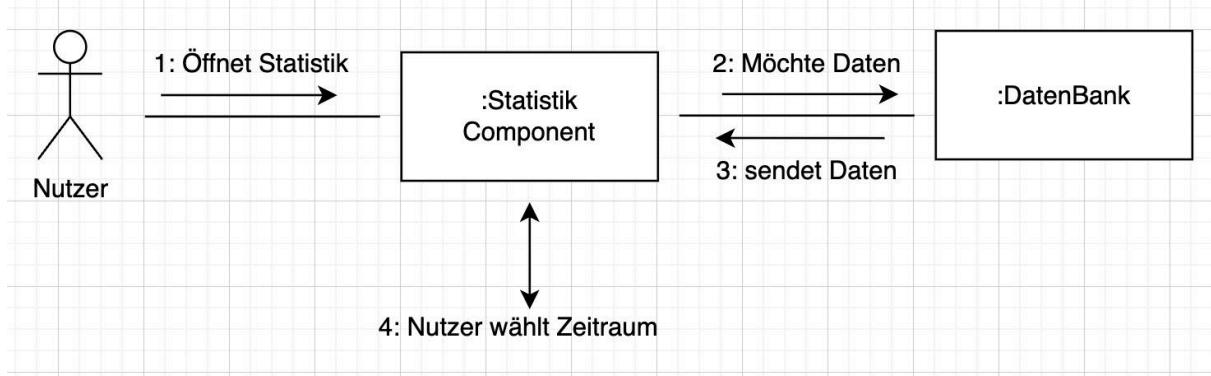


## Verhaltensdiagramme (Kommunikationsdiagramme)

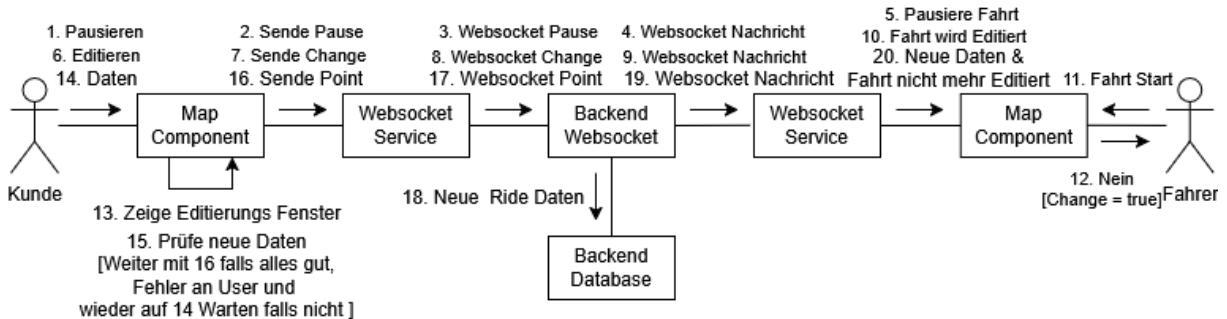
Verhaltensdiagramm Leaderboard (Lucy Brandt):



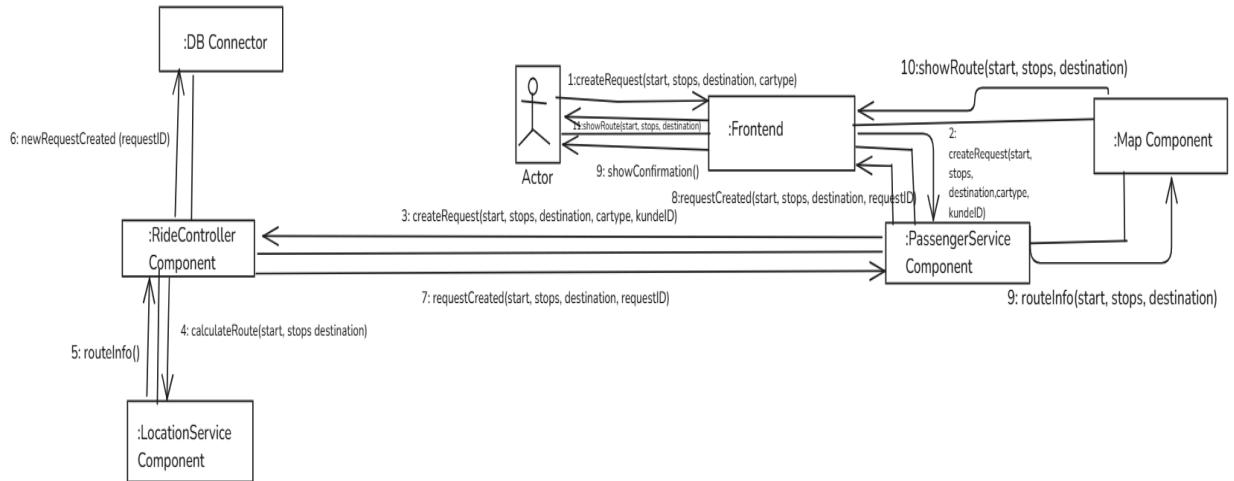
Verhaltensdiagramm Statistik (Lucy Brandt):



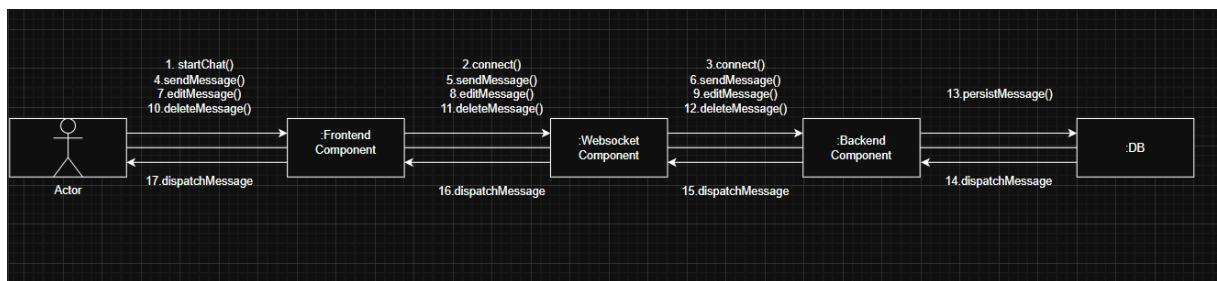
Verhaltensdiagramm Live-Änderungen (Noel von Stockum):



## Verhaltensdiagramm Zwischenstopps Fahrtpplan: Dilay Topal



## Verhaltensdiagramm Chat (Konstantin Shuti):



## Funktionalitätsplanung

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
<b>1.</b>						
1.1	Live Änderungen	Noel von Stockum		5.1.13; 5.1.14; 5.1.15; 5.1.16	Frontend/src/app/components/ <a href="#">map.component.*</a> (alles außer was bei Dilay steht)	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>
1.2	Zwischenstopps	Dilay Topal		5.1.6	Frontend/src/app/components/ <a href="#">map.component.ts</a> (zeilen: 25, 142-144, 149-156, tryLoadFromStorage() ); Frontend/src/app/components/drivedata.component.html (zeilen: 18-28); Frontend/src/app/components/drivedata.component.css (zeilen: 62-87); Frontend/src/app/components/drivedata.component.ts (zeilen: 25, 28-40, 119-133, 136, 148-149); Frontend/src/app/com	<span style="background-color: #c8f7e4; border-radius: 10px; padding: 2px 10px;">Abgeschlossen ▾</span>

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
					<p>ponents/bookmark.component.html (zeilen: 8-13) ; Frontend/src/app/components/bookmark.component.css (zeilen: 58-67) ; Frontend/src/app/components/bookmark.component.ts (zeilen: 31-33, 49-53, 64-71, 79, toggleStopText()) ; Frontend/src/app/components/current-drive.component.html (zeilen: 12-17) ; Frontend/src/app/components/current-drive.component.css (zeilen: 76-86) ; Frontend/src/app/components/current-drive.component.ts (zeilen: 11, 18, 33, 59-61, 70-81, 87, 94, 99-102,</p>	

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
					toggleStopText() ; Frontend/src/app/services/ <a href="#">ride.service.ts</a> (zeilen: 11-12, 27-28, 41-42)	
1.3	Statistiken	Lucy Brandt		5.1.8; 5.1.9; 5.1.10; 5.1.11; 5.1.12	Frontend/src/app/components/statistics.component.*	Abgeschlossen ▾
1.4	Leaderboard	Dilay Topal		5.1.17; 5.1.18;	Frontend/src/app/components/leaderboard.component.*	Abgeschlossen ▾
1.5	Chat	Noel von Stockum		5.1.1; 5.1.2; 5.1.3; 5.1.4; 5.1.5	Frontend/src/app/components/chat. <a href="#">component</a> .*	Abgeschlossen ▾
1.6	Websocket Service	Dilay Topal		5.1.1; 5.1.3; 5.1.4; 5.1.5	Frontend/src/app/services/ <a href="#">websocket.service.ts</a> (zeilen: 16, 19, 23, 28, 38-40, sendChatMessage(), editChatMessage(), deleteChatMessage(), changingRoute(), sendPoints()))	Abgeschlossen ▾
2.						
2.1	Chat Websocket	Felix Rieger		5.1.1	Backend/car-mate/src/main/java/de/car_mate/car_mate	Abgeschlossen ▾

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
					/controller/ChatWebSockethandle.r.java	
2.2	Chat Socket History OnInit	Dario Connor Reddig	2.1	5.1.1; 5.1.5	Backend/car-mate/src/main/java/de/car_mate/car_mate/controller/ChatWebSockethandle.r.java/loadChatHistoryMessages	Abgeschlossen ▾
2.3	Chat Socket Command Nachricht Send	Dario Connor Reddig	2.1	5.1.1; 5.1.5	Backend/car-mate/src/main/java/de/car_mate/car_mate/controller/ChatWebSockethandle.r.java/handleSendMessage	Abgeschlossen ▾
2.4	Chat Socket Command Nachricht Edit	Dario Connor Reddig	2.1; 2.13	5.1.4; 5.1.5	Backend/car-mate/src/main/java/de/car_mate/car_mate/controller/ChatWebSockethandle.r.java/handleEditMessage	Abgeschlossen ▾
2.5	Chat Socket Command Nachricht Delete	Konstandin Shuti	2.1; 2.13	5.1.3; 5.1.5	Backend/car-mate/src/main/java/de/car_mate/car_mate/controller/ChatWebSockethandle.r.java/handleDeleteMessage	Abgeschlossen ▾

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
2.6	Live-Änderung Ride Websocket Point	Luis Bleckmann		5.1.13; 5.1.15; 5.1.16	gruppe-g\B ackend\car -mate\src\ main\java\ de\car_mat e\car_mate\ controller\ RideWebSo ckethandle r.java	Abgeschlossen ▾
2.7	Live-Änderung Ride Websocket Change	Luis Bleckmann		5.1.14; 5.1.15; 5.1.16	gruppe-g\B ackend\car -mate\src\ main\java\ de\car_mat e\car_mate\ controller\ RideWebSo ckethandle r.java	Abgeschlossen ▾
2.8	CreateRide Zwischenstopps	Luis Bleckmann		5.1.6	Backend\ca r-mate\src\ main\java\ de\car_mat e\car_mate\ \Service\Ri deService.j ava	Abgeschlossen ▾
2.9	GetRide Zwischenstopps	Konstandin Shuti		5.1.6	Backend/ca r-mate/src/ main/java/ de/car_mat e/car_mate/ repo/Ride Repo.java:g etRideStop s	Abgeschlossen ▾
2.10	GetCurrentRide Zwischenstopps	Konstandin Shuti		5.1.6	Backend/ca r-mate/src/ main/java/ de/car_mat e/car_mate/ repo/Ride Repo.java:g etCurrentRi deStops	Abgeschlossen ▾
2.11	getStats	Felix Rieger		5.1.7, 5.1.8, 5.1.9,	Backend/ca r-mate/src/ main/java/	Abgeschlossen ▾

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
				5.1.10, 5.1.11, 5.1.12	de/car_mate/car_mate/controller/RideController.java:getStats,Backend/car-mate/src/main/java/de/car_mate/car_mate/model/StatsElement.java,Backend/car-mate/src/main/java/de/car_mate/car_mate/Repo/RideRepo.java:getStats,Backend/car-mate/src/main/java/de/car_mate/car_mate/Service/RideService.java:getStats,Backend/car-mate/src/main/java/de/car_mate/car_mate/Service/RideService.java:getStats	
2.12	Leaderboard	Felix Rieger		5.1.17, 5.1.18	Backend/car-mate/src/main/java/de/car_mate/car_mate/controller/RideController.java:getLeaderboard,	Abgeschlossen ▾

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
					Backend/car-mate/src/main/java/de/car_mate/car_mate/model/Leader.java, Backend/car-mate/src/main/java/de/car_mate/car_mate/Repo/RideRepo.java:getLeader, Backend/car-mate/src/main/java/de/car_mate/car_mate/Service/RideService.java:getleaders	
2.13	Chat Nachricht wurde gelesen	Dario Connor Reddig	2.1	5.1.1; 5.1.5	Backend/car-mate/src/main/java/de/car_mate/car_mate/Repo/ChatRepo.java/messagesRead	Abgeschlossen ▾

## Modultests

ID	Getestete Funktionalität	Quellcode Referenz	Status
1	leaderboard	leader()	passed

## Systemtests

Datum	13.07.2025		
Tester	Noel von Stockum		
SW-Version			
Vorbedingung(en)	<p>Fahrer und Kunde existieren in der Datenbank. Mindestens der Kunde ist angemeldet. Es gibt eine Fahrt mit diesem Fahrer und Kunden, die noch nicht beendet wurde. Beide sind auf der "Home"-View (Menübar).</p> <p>Die Fahrt ist pausiert.</p>		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde klickt auf den "Änderungen Hinzufügen?" Button	<p>Kunde sieht Eingabefelder für neue Zwischenstopps und das Ziel.</p> <p>Fahrer kann die Fahrt nicht mehr starten oder zurücksetzen.</p>	✓
2	Kunde klickt n mal auf den "+" Button über "Zwischenstopp"	<p>Es werden n viele Eingabefelder erstellt.</p> <p>Falls zu viele erstellt werden, gibt es eine scrollbar.</p>	✓
3	Kunde gibt die gewünschten Änderungen ein, löscht gegebenenfalls nicht benötigte Zwischenstopp Eingabefelder und klickt dann auf den "Weiter" Button	<p>Kunde wird wieder auf die Karten view weitergeleitet</p> <p>Beide sehen, nach einer kurzen berechnungszeit, die neuen Marker und die neue route von der aktuellen Position des Fahrzeugs zum ersten neuen/noch nicht passierten Zwischenstopp oder dem Ziel</p>	✓
4	Fahrer oder Kunde klickt auf den "Play" Button.	Die Fahrt wird entlang der neuen Route fortgesetzt	✓
Nachbedingung(en)	<p>In der Datenbank, in der drives Tabelle, für diese Fahrt, wird active auf 0 gesetzt.</p> <p>Die Fahrt ist nun für beide in der Historie und für den Fahrer in der Statistik sichtbar.</p>		
Testurteil	Bestanden		

<b>Datum</b>	13.07.2025		
<b>Tester</b>	Noel von Stockum		
<b>SW-Version</b>			
<b>Vorbedingung(en)</b>	Der Fahrer existiert in der Datenbank. Fahrer ist angemeldet, hat mindestens eine Fahrt absolviert und hat zur "Statistik"-View (Menübar) navigiert.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / ✗</b>
1	Fahrer schaut sich die Statistik und Leaderboard an	Beide zeigen die korrekten Daten aus dem Backend	✓
2	Fahrer wechselt zwischen Monatlicher und Täglicher ansicht der Statistiken	Die X und (sofern nötig) Y Achse zeigen die korrekten Einheiten und Werte	✓
3	Fahrer wählt Datum einer älteren Fahrt aus (z.b. durch admin panel erstellt)	Statistiken spiegeln diese korrekt wieder	✓
4	Fahrer sortiert das Leaderboard anhand unterschiedlicher Spalten, Auf- und Absteigend	Leaderboard wird jedesmal mit korrekt sortieren Daten angezeigt	✓
<b>Nachbedingung(en)</b>	Daten waren jederzeit korrekt für den User einsehbar		
<b>Testurteil</b>	Bestanden		

<b>Datum</b>	13.07.2025		
<b>Tester</b>	Noel von Stockum		
<b>SW-Version</b>			
<b>Vorbedingung(en)</b>	Der Kunde existiert in der Datenbank. Kunde ist angemeldet und hat auf die "Fahrt"-View (Menübar) navigiert.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Es wird beliebig oft auf den "+" Button über "Zwischenstopp" geklickt.	Für jeden Klick wird ein neues Eingabefeld und "-" Button für den neuen Zwischenstopp erstellt.	✓
2	Es werden die gewünschten Orte eingetragen (Hier zum test als start: "Essen", Zwischenstopp: "Duisburg", Ziel: "Düsseldorf") und auf den "Weiter" Button geklickt	Gewünschte Orte werden übernommen und in der nächsten Ansicht nochmal zur Bestätigung sowohl als Adressen als auch auf der Karte angezeigt.	✓
3	Der Kunde klickt auf den "Buchen"-Button.	Route wird im Backend gespeichert und nochmals in einem ähnlichen View angezeigt.	✓
<b>Nachbedingung(en)</b>	In der "drives" Tabelle wurde die Fahrt persistent gespeichert.		
<b>Testurteil</b>	Bestanden		

<b>Datum</b>	13.07.2025		
<b>Tester</b>	Noel von Stockum		
<b>SW-Version</b>			
<b>Vorbedingung(en)</b>	Kunde und Fahrer existieren in der Datenbank. Es gibt mindestens eine offene oder angenommene Anfrage durch einen Fahrer. Kunde ist angemeldet und hat auf die "Fahrt"-View (Menübar) navigiert.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Kunde klickt auf den Usernamen des Fahrers	Wird auf das Profil weitergeleitet	✓
2	Kunde klickt auf den "Chat" Button unten rechts	Wird auf einen Chat mit dieser Person weitergeleitet.	✓
3	Kunde schreibt Nachricht und drückt Enter	Nachricht wird bei beiden angezeigt, sofern der Fahrer sich auch im Chat befindet. Wenn sich nur der Kunde im Chat befindet (bzw. der Fahrer ausgeloggt ist), sind diese Nachrichten zunächst als nicht gelesen markiert.	✓
4	Kunde schreibt und sendet eine weitere nachricht, aber Löscht diese wieder und der Fahrer ist zurzeit ausgeloggt	Nachricht wird angezeigt, wieder gelöscht und kommt beim Einloggen des Fahrers auch nicht auf.	✓
5	Kunde schreibt und sendet eine weitere nachricht, aber versucht diese wieder zu Löschen und der Fahrer ist zurzeit im Chat	Nachricht wird bei beiden angezeigt und nicht gelöscht.	✓
6	Kunde schreibt und sendet eine weitere nachricht, aber Editiert diese wieder (indem er nochmals etwas in das Eingabefeld einträgt und auf den edit Knopf unter der Nachricht drückt) und	Nachricht wird angezeigt, überarbeitet und kommt beim Einloggen des Fahrers im neuen Stand an.	✓

	der Fahrer ist zurzeit ausgeloggt		
7	Kunde schreibt und sendet eine weitere nachricht, aber versucht diese zu Editieren und der Fahrer ist zurzeit im Chat	Nachricht wird bei beiden angezeigt und nicht überarbeitet.	√
<b>Nachbedingung(en)</b>	Nachrichten sind in der Datenbank Persistent gespeichert		
<b>Testurteil</b>	Bestanden		

# Nutzerhandbuch

## Technische Anforderungen

Technische Mindestanforderungen, welche das System benötigt, um wie gewünscht bedienbar zu sein.

- Internetanbindung
- Browser
- Möglichkeit Docker zu bedienen
- Docker Compose Unterstützung

## Installationsanleitung

Genaue Erläuterung, wie das entwickelte System vollkommen funktionsfähig auf einem Rechner in Betrieb genommen werden kann.

1. Nach dem Download auf ihrem System entzippen
2. test.env datei zu .env umbenennen (da Sciebo diese sonst nicht akzeptiert)
3. Im Terminal in das Verzeichnis navigieren
4. Dort folgende Befehle nacheinander ausführen:
  - a. “docker load -i gruppe-g-admin.tar”
  - b. “docker load -i gruppe-g-frontend.tar”
  - c. “docker load -i gruppe-g-app.tar”
  - d. “docker load -i gruppe-g-grafana.tar”
  - e. “docker load -i mariadb.tar”
  - f. “docker load -i redis.tar”
  - g. “docker load -i mailhog.tar”
  - h. “docker compose up -d”
5. Nachdem alle Container auf “Built”, “Created” oder “Started” stehen:
6. Im Browser: “[localhost](http://localhost)” für die Anwendung und “[localhost/nav](http://localhost/nav)” für alle weiteren Dienste.

(optional): Im Browser: “[localhost/admin](http://localhost/admin)”, wo Sie die Credentials: “password” anwenden können.

Die generierten Nutzer haben alle das Passwort “Test1234”.

## Bedienungsanleitung

Genaue Erläuterung, wie das entwickelte System zu bedienen ist.

### Vorbedingungen

Um gleichzeitig als Fahrer und Kunde diese App zu bedienen, wird empfohlen, einen normalen und einen privaten Tab zu öffnen, da sich zwischen diesen die Sessions nicht synchronisieren.

## 1. Registrierung und Anmeldung

Falls noch kein Benutzerkonto vorhanden ist, klicken Sie in der Login-Ansicht auf „**Registrieren**“. Wählen Sie dort aus, ob Sie sich als Fahrer oder Kunde registrieren möchten, und füllen Sie anschließend das Formular aus. Optional kann ein Profilbild hochgeladen werden – dieses darf maximal **1 MB groß** und muss vom Dateitypen **.png** sein. Das Adminpanel kann hier sehr hilfreich sein (path: **localhost/admin**, passwort: **password**), User welche dort erstellt wurden haben das Passwort: **Test1234**.

## 2. E-Mail-Verifikation und MFA

Für die E-Mail-Verifikation sowie den Multi-Faktor-Authentifizierungs-Code (MFA) öffnen Sie den lokalen E-Mail-Client unter „**localhost/mail**“. Klicken Sie auf die **aktuellste/oberste Nachricht** und entnehmen Sie den dort angegebenen Code. Alternativ kann für den MFA auch der Supercode „**1111**“ verwendet werden.

## 3. Navigation

Im oberen Bereich der Anwendung befindet sich eine Menüleiste, über die Sie folgende Bereiche aufrufen können:

- **Home** (auch Kartenansicht)
- **Fahrten** (zur Fahrt Planung)
- **Chat** (hier auch die Möglichkeit nach Usern zu suchen und deren Profil zu sehen)
- **Statistik** (auch Historie und Leaderboard)
- **Profilbild** (das eigene Profil ansehen, auch Geldkonto einsehen und aufladen)

## 4. Kartenansicht

In der **Home**-Ansicht sehen Sie, sofern vorhanden, Ihre aktuell aktive Fahrt auf der Karte inklusive Fahrt durchführung.

## 5. Fahrten

Im Bereich „**Fahrten**“ geben Sie den **Startpunkt**, die **Zwischenstopps**, das **Ziel** und den gewünschten **Fahrzeugtyp** an. Diese Eingaben können als **Adresse**, **Koordinate** oder **Point of Interest (POI)** erfolgen. Alternativ können Sie Ihren **aktuellen Standort** automatisch ermitteln lassen, indem Sie auf den entsprechenden Button klicken.

Nach Eingabe aller Daten klicken Sie auf „**Weiter**“, um zur Buchungsansicht zu gelangen. Dort können Sie Ihre Angaben nochmals überprüfen.

## 6. Aktive Fahrt

Nach dem Absenden einer Fahranfrage wird diese nochmals in einer separaten Ansicht angezeigt. Sie können die Anfrage dort jederzeit **zurückziehen**.

Wenn Sie zwischenzeitlich zu einer anderen Ansicht und wieder zurück wechseln, wird Ihnen die aktuelle Fahrt weiterhin angezeigt. In diesem Zustand ist es nicht möglich, eine neue Fahrt zu planen, solange die bestehende Anfrage aktiv ist.

Bei einem gestellten Angebot an ihre Fahrt wird diese Sicht durch das Neuladen, einen Klick auf das Pop-up, einen Klick auf “Aktuelle Angebote” oder durch das Später dahin Navigieren mit einer Liste von Angeboten ersetzt. Dort sieht der Kunde alle Angebote und kann diese sortieren sowie durch Klick auf den grünen Haken annehmen. Durch ein Klicken auf das rote X kann ein Angebot abgelehnt werden.

## 7. Chat

Im Bereich "**Chat**" können Sie nach registrierten Benutzern suchen, indem Sie oben links den Benutzernamen eingeben. Sofern es schon Chats mit anderen Nutzern gab, wird einer der Alten Chats geöffnet und bei keiner Eingabe sind unter dem Eingabefeld weitere Nutzer, mit denen Sie schon geschrieben haben, zu sehen.

Um Nachrichten zu verfassen, schreiben Sie in das untere Eingabefeld und drücken Sie die Enter-Taste zum Abschicken (oder klicken Sie auf den Papierflieger am rechten Rand des Eingabefeldes). Sofern der andere Nutzer während seiner aktuellen Session schon mal im Chat war, werden diese Nachrichten als gelesen gehandhabt. Um sicherzugehen, dass Sie mit nicht gelesenen Nachrichten arbeiten können, muss der andere Nutzer abgemeldet sein.

Nachrichten editieren funktioniert, indem Sie den neuen Nachricht-Inhalt in das Eingabefeld unten eintragen und danach auf den Stift und Papier Knopf zwischen Uhrzeit und Mülleimer unter der Nachricht klicken. Sofern der andere Nutzer während seiner aktuellen Session schon mal im Chat war, wird diese Nachricht als gelesen gehandhabt und nicht editiert. Um sicherzugehen, dass Sie mit nicht gelesenen Nachrichten arbeiten können, muss der andere Nutzer abgemeldet sein, während die Nachricht verschickt wurde.

Nachrichten löschen funktioniert, indem Sie auf den Mülleimer Knopf am rechten Rand unter der Nachricht klicken. Sofern der andere Nutzer während seiner aktuellen Session schon mal im Chat war, wird diese Nachricht als gelesen gehandhabt und nicht gelöscht. Um sicherzugehen, dass Sie mit nicht gelesenen Nachrichten arbeiten können, muss der andere Nutzer abgemeldet sein, während die Nachricht verschickt wurde.

## **8. Angebot erstellen**

Als Fahrer auf "Fahrten" klicken. Durch Klicken kann auf die entsprechende Fahrt ein Angebot abgegeben werden, die Fahrt wird grün markiert. Durch erneutes Klicken wird das Angebot zurückgezogen.

## **9. Fahrt Durchführung**

Der Nutzer wechselt auf Home/Map. Durch klicken auf Start wird die Fahrt für Fahrer und Kunde gestartet. Durch Klicken auf Pause wird die Fahrt angehalten. Hier kann die Fahrt wieder gestartet werden. Am Ziel angekommen, wird man auf die Bewertung weitergeleitet.

## **10. Live-Änderungen**

Während eine Fahrt pausiert ist, kann der Kunde Änderungen an der Fahrt vornehmen, indem er den Knopf oben Links auf der Karte anklickt. Dort können, noch nicht erreichte, Zwischenstopps und das Ziel bearbeitet werden. Sofern keine Änderungen gewünscht sind, sollen die Eingabefelder so gelassen werden und dennoch auf "Weiter" geklickt werden.

## **11. Bewertung**

Hier wählt der Nutzer die entsprechende Anzahl an Sternen aus und bestätigt. Danach wird der Nutzer auf die Startseite weitergeleitet. Wenn die Webseite an dieser Stelle geschlossen wird, anstatt eine Bewertung abzugeben, hängt sich das System für diesen Nutzer auf.

## **12. Fahrt Historie**

Als Nutzer auf Statistik klicken. Dort sieht der Nutzer alle abgeschlossenen Fahrten und kann diese sortieren.

## **13. Statistiken**

Als Fahrer auf Statistik klicken. Dort sieht der Fahrer alle persönlichen Statistiken zu seinen Fahrten und kann diese monatlich oder täglich anzeigen lassen und nach einem bestimmten Datum suchen.

## **14. Leaderboard**

Als Nutzer auf Statistik klicken. Dort sieht der Nutzer das Leaderboard und kann die Daten sortieren.