

SEP Projektmappe

PROJEKTMAPPE DES PROJEKTES

SEP-Drive

Dokumentation des Projektes

Gruppe M:

Simon Elias Lobo Nabert

Finn Berthold

Dennis Lüdiger

Tim Besau

Björn Adler

Zheyuan Wu

Mohamad Al Atrach

Inhalt

Projektbeschreibung	4
Zyklus I.....	5
Spezifikationsplanung	5
User-Stories	7
Papierprototypen	10
Strukturdiagramme (Komponenten- und Klassendiagramme)	15
Verhaltensdiagramme (Kommunikationsdiagramme)	17
Funktionalitätsplanung	20
Systemtests	24
Zyklus II	31
Spezifikationsplanung	31
User-Stories	32
Papierprototypen	36
Strukturdiagramme (Komponenten- und Klassendiagramme)	41
Verhaltensdiagramme (Kommunikationsdiagramme)	44
Funktionalitätsplanung	52
Modultests	57
Systemtests	58
Zyklus III.....	66
Spezifikationsplanung	66
User-Stories	67
Papierprototypen	69
Strukturdiagramme (Komponenten- und Klassendiagramme)	75
Verhaltensdiagramme (Kommunikationsdiagramme)	78
Funktionalitätsplanung	83
Modultests	86
Systemtests	86
Nutzerhandbuch.....	92
Technische Anforderungen	92
Installationsanleitung	92
Bedienungsanleitung.....	93

Projektbeschreibung

SEP-Drive ist eine Mitfahr-App, bei der Kunden Fahrtanfragen erstellen und Fahrtangebote von Fahrern erhalten. SEP-Drive 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 bewegt. Das Projekt soll als Webanwendung (bestehend aus einem Front- und Backend) umgesetzt werden. Front- und Backend sollen voneinander separat entwickelt werden und jeweils als eigenständiger 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.	Registrierung der Nutzer			
1.1	Nutzer Registrierung	User Story	Zheyuan	Fertig
1.1.1	Registrierungs-Fenster	Papierprototyp	Zheyuan	Fertig
2.	Login eines Benutzers			
2.1	Login Nutzer	User Story	Zheyuan	Fertig
2.1.1	Login-Fenster	Papierprototyp	Zheyuan	Fertig
2.2	Zwei-Faktor-Authentifizierung	User Story	Dennis	Fertig
2.2.1	Zwei-Faktor-Auth-Fenster	Papierprototyp	Dennis	Fertig
3.	Benutzerprofile			
3.1	Nutzerprofil	User Story	Finn	Fertig
3.1.1	Fahrerprofil-Fenster	Papierprototyp	Finn	Fertig
3.1.2	Kundenprofil-Fenster	Papierprototyp	Finn	Fertig
4.	Kartenvisualisierung von Routen			
4.1	Kartenvisualisierung	User Story	Simon	Fertig
4.1.1	Kartenvisualisierung	Papierprototyp	Finn	Fertig
4.2	Zoom/Bewegen der Karte	User Story	Simon	Fertig
5.	Fahranfragen erstellen			
5.1	Start und Ziel festlegen	User Story	Simon	Fertig
5.1.1	Fahranfrage-Fenster	Papierprototyp	Simon	Fertig
5.2	Fahrzeugklasse	User Story	Simon	Fertig
5.3	Aktive Fahranfrage einsehen	User Story	Dennis	Fertig
5.3.1	Aktive Fahranfrage-Fenster	Papierprototyp	Dennis	Fertig
5.4	Fahranfrage löschen	User Story	Dennis	Fertig
6.	Dashboard			
6.1	Suche	User Story	Dennis	Fertig
6.1.1	Suchergebnisse	Papierprototyp	Finn	Fertig

6.2	Dashboard-Fenster	Papierprototyp	Finn	Fertig
7.	Architektur			
7.1	Komponentendiagramm	Diagramm	Tim	Fertig
7.2	Klassendiagramm	Diagramm	Björn	Fertig
7.3	Kommunikationsdiagramm	Diagramm	Mohamad	Fertig

User-Stories

User Story-ID	1.1
User Story-Beschreibung	Als Nutzer möchte ich mich mit meinem Benutzernamen, Vornamen, Nachnamen, E-Mail, Geburtsdatum, Rolle (Fahrer oder Kunde) registrieren, damit ich einen Account erstellen kann.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Zheyuan Wu
Abhängigkeiten zu anderen User Stories	2.1, 2.2

User Story-ID	2.1
User Story-Beschreibung	Als Benutzer möchte ich mich mit meinem Benutzernamen und Passwort einloggen, sodass ich Zugang zum System erhalten kann.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Zheyuan Wu
Abhängigkeiten zu anderen User Stories	1.1, 2.2

User Story-ID	2.2
User Story-Beschreibung	Als Nutzer möchte ich den Login-Vorgang zusätzlich durch eine Zwei-Faktor-Authentifizierung sichern, um mein Konto noch besser zu schützen
Geschätzter Realisierungsaufwand	1 Tag
Priorität	gering
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	2.1

User Story-ID	3.1
User Story-Beschreibung	Als Kunde oder als Fahrer möchte ich mir mein Nutzer-Profil angucken können durch „klicken“ auf Benutzernamen oder durch die Suchfunktion.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	mittel
Autor	Finn Berthold
Abhängigkeiten zu	1.1, 2.1, 6.1

anderen User Stories	
----------------------	--

User Story-ID	4.1
User Story-Beschreibung	Als Nutzer möchte ich die Route auf der Karte angezeigt bekommen, um Start, Ziel und Zwischenstopps der Strecke visuell nachvollziehen zu können.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	4.2, 5.1

User Story-ID	4.2
User Story-Beschreibung	Als Nutzer möchte ich mit der Karte interagieren können, um mir die Route genauer anzuschauen.
Geschätzter Realisierungsaufwand	0,5 Tage
Priorität	gering
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	4.1

User Story-ID	5.1
User Story-Beschreibung	Als Kunde möchte ich eine Fahranfrage stellen können, um eine Fahrt von einem Startpunkt zu meinem gewünschten Ziel zu buchen.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	5.2, 5.3, 5.4

User Story-ID	5.2
User Story-Beschreibung	Als Kunde möchte ich für meine Fahranfrage eine Fahrzeugklasse auswählen können, sodass nur Fahrer infrage kommen die meinen Bedürfnissen entsprechen.
Geschätzter Realisierungsaufwand	0,5 Tage
Priorität	mittel
Autor	Simon Lobo
Abhängigkeiten zu	5.1, 5.3, 5.4

anderen User Stories	
-----------------------------	--

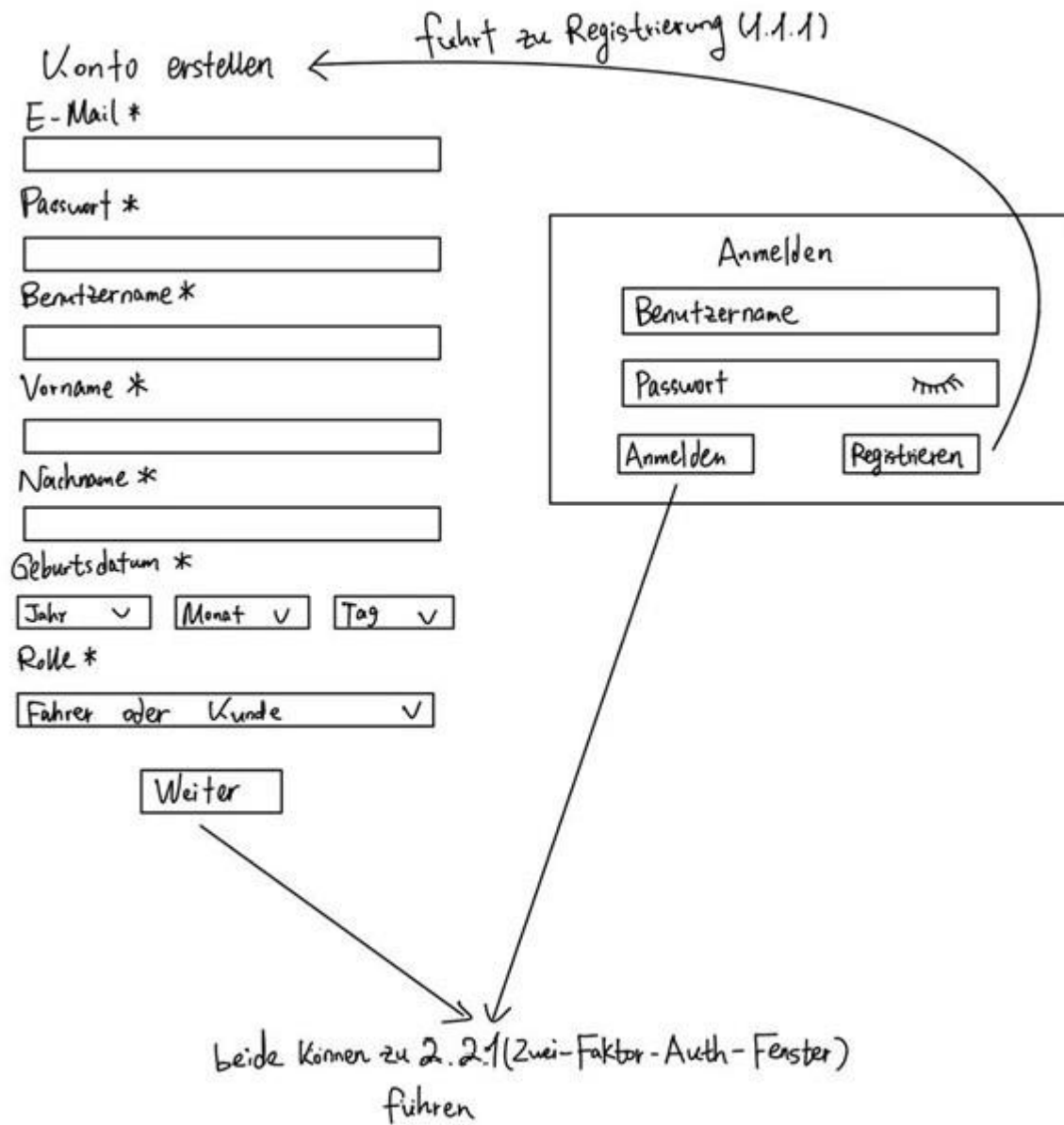
User Story-ID	5.3
User Story-Beschreibung	Als Kunde möchte ich meine aktive Fahranfrage auf einer zusätzlichen Seite sehen können, um einen besseren Überblick zu haben
Geschätzter Realisierungsaufwand	0,5 Tage
Priorität	mittel
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	5.1, 5.2, 5.4

User Story-ID	5.4
User Story-Beschreibung	Als Kunde möchte ich die Möglichkeit haben meine aktive Fahranfrage löschen zu können, um flexibler zu sein
Geschätzter Realisierungsaufwand	0,5 Tage
Priorität	gering
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	5.1, 5.3

User Story-ID	6.1
User Story-Beschreibung	Als Nutzer möchte ich eine Suchfunktion haben, um nach anderen Benutzern suchen zu können
Geschätzter Realisierungsaufwand	2 Tage
Priorität	mittel
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	3.1

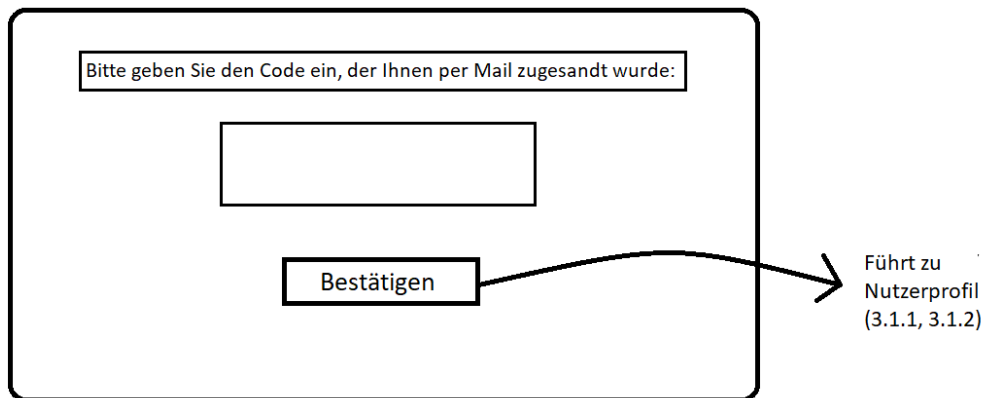
Papierprototypen

1.1.1 Registrierungs-Fenster + 2.1.1 Login-Fenster:



2.2.1 Zwei-Faktor-Authentifizierung:

Zwei-Faktor-Authentifizierung



3.1.1 Fahrerprofil-Fenster:

1.



3.1.2 Kunden-Profil-Fenster:

1.

Profilbild

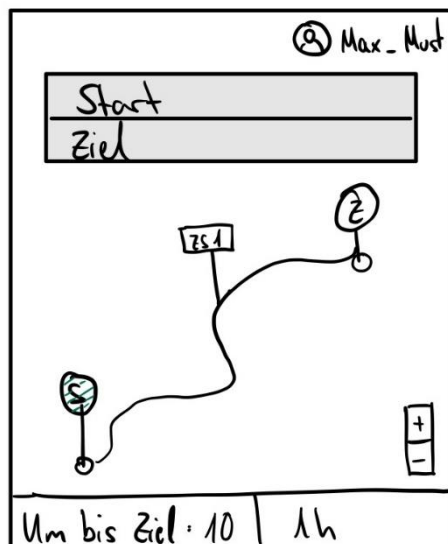
Benutzername: max_muster
Rolle: Kunde
Vorname: Max
Nachname: Muster
Email: max@example
Geburtsdatum:

[Rating: ★★★★★]
[Fahrten: 23]

Profil bearbeiten

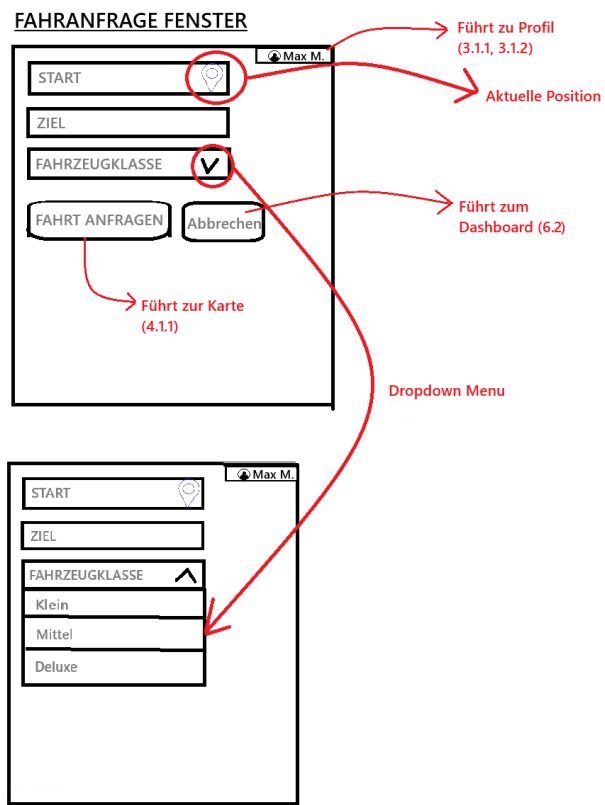
4.1.1 Kartenvisualisierung:

3.



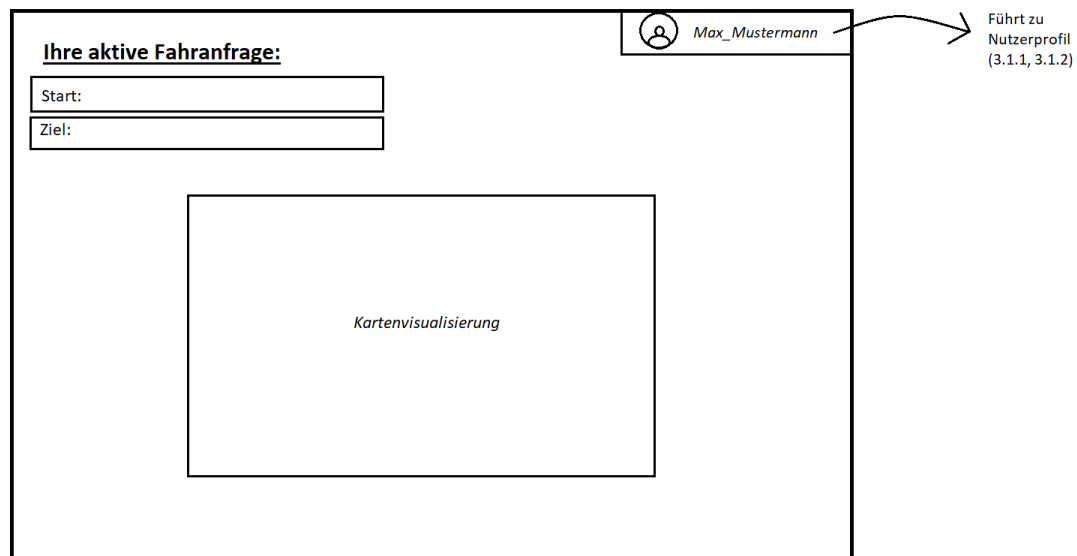
→ führt zu Nutzerprofil

5.1.1 Fahranfrage Fenster:

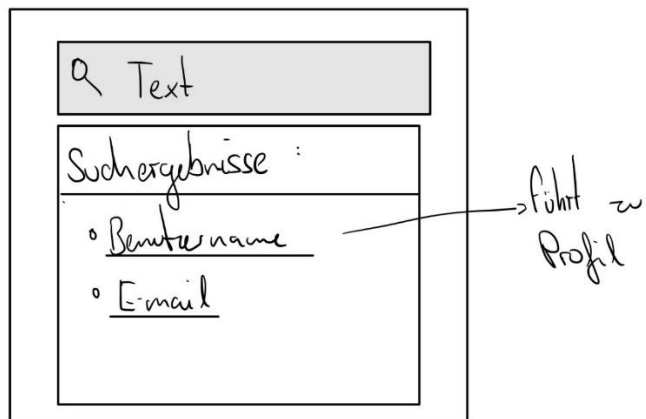


5.3.1 Aktive Fahranfrage Seite:

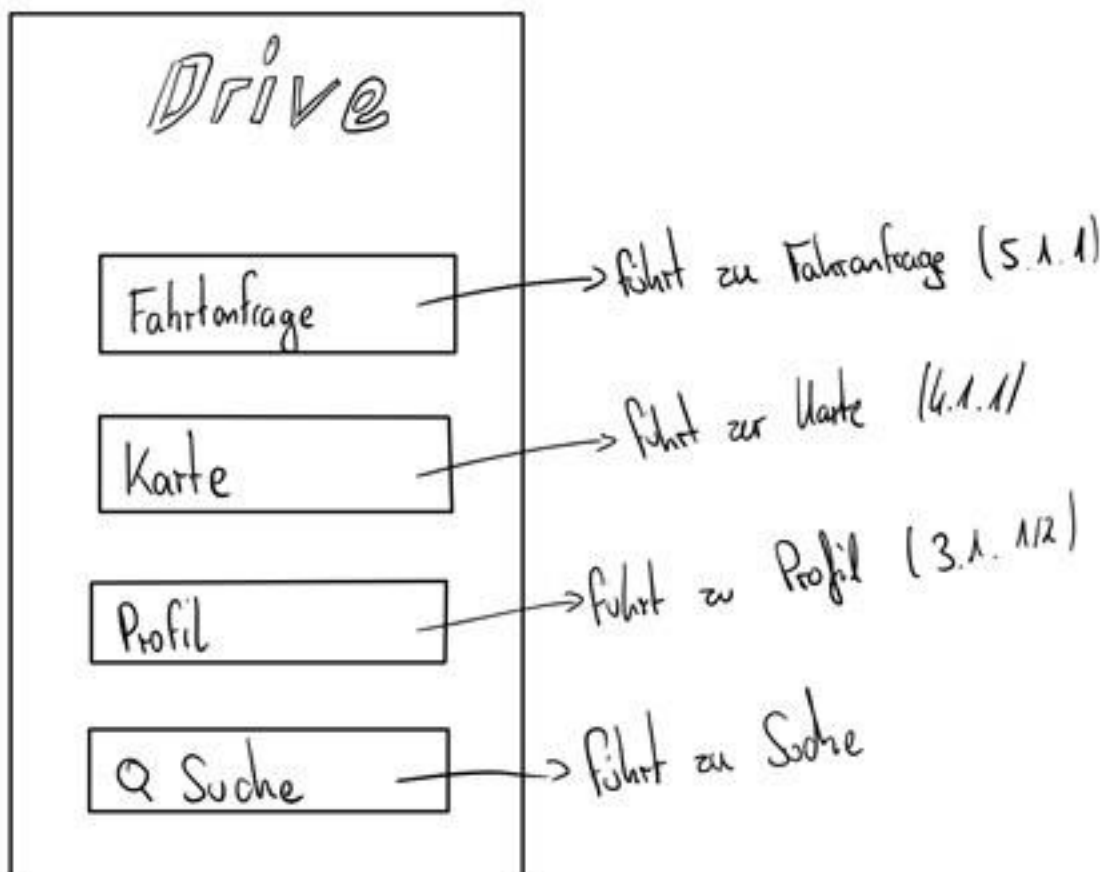
Aktive Fahranfrage Seite



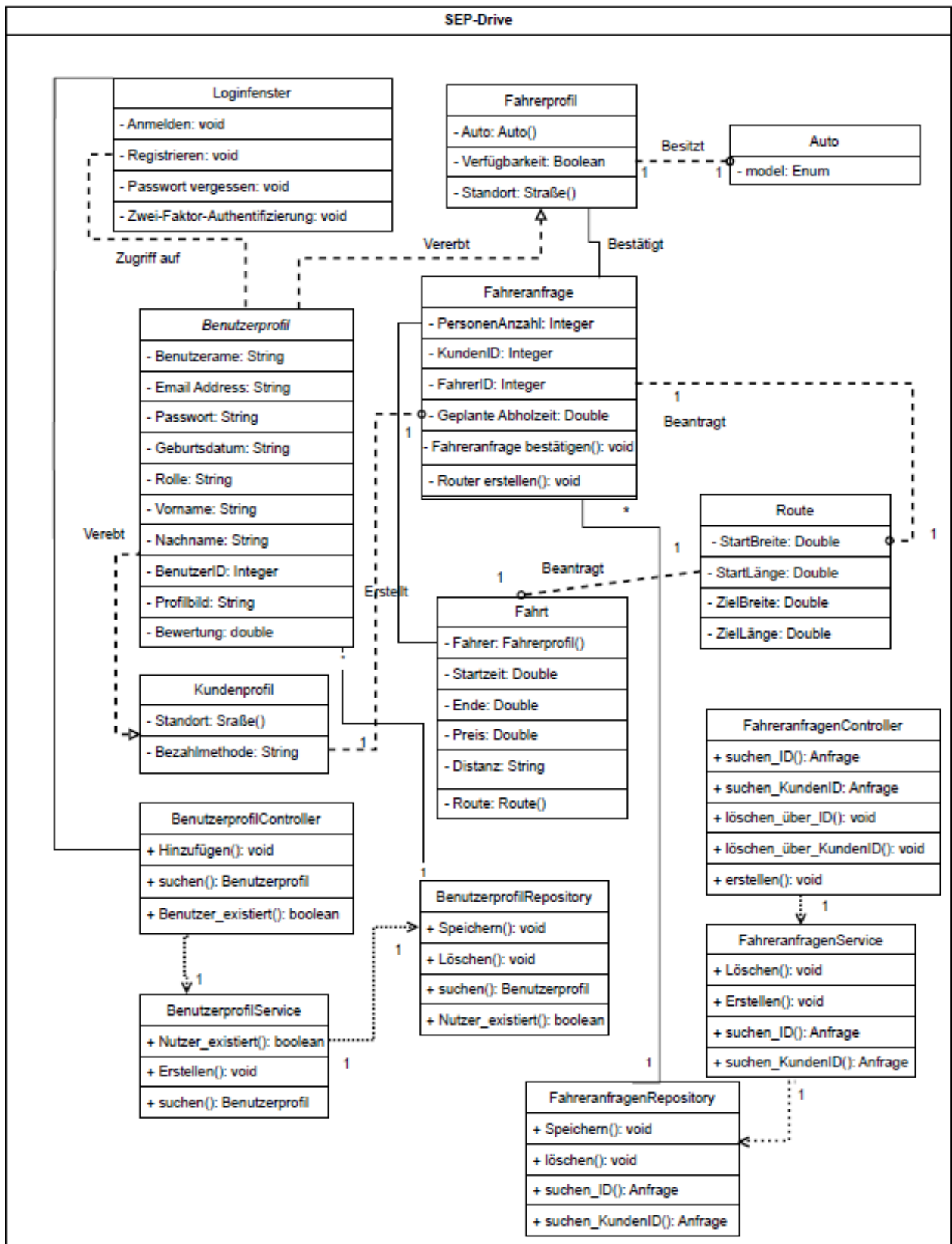
6.1.1 Suchergebnisse:

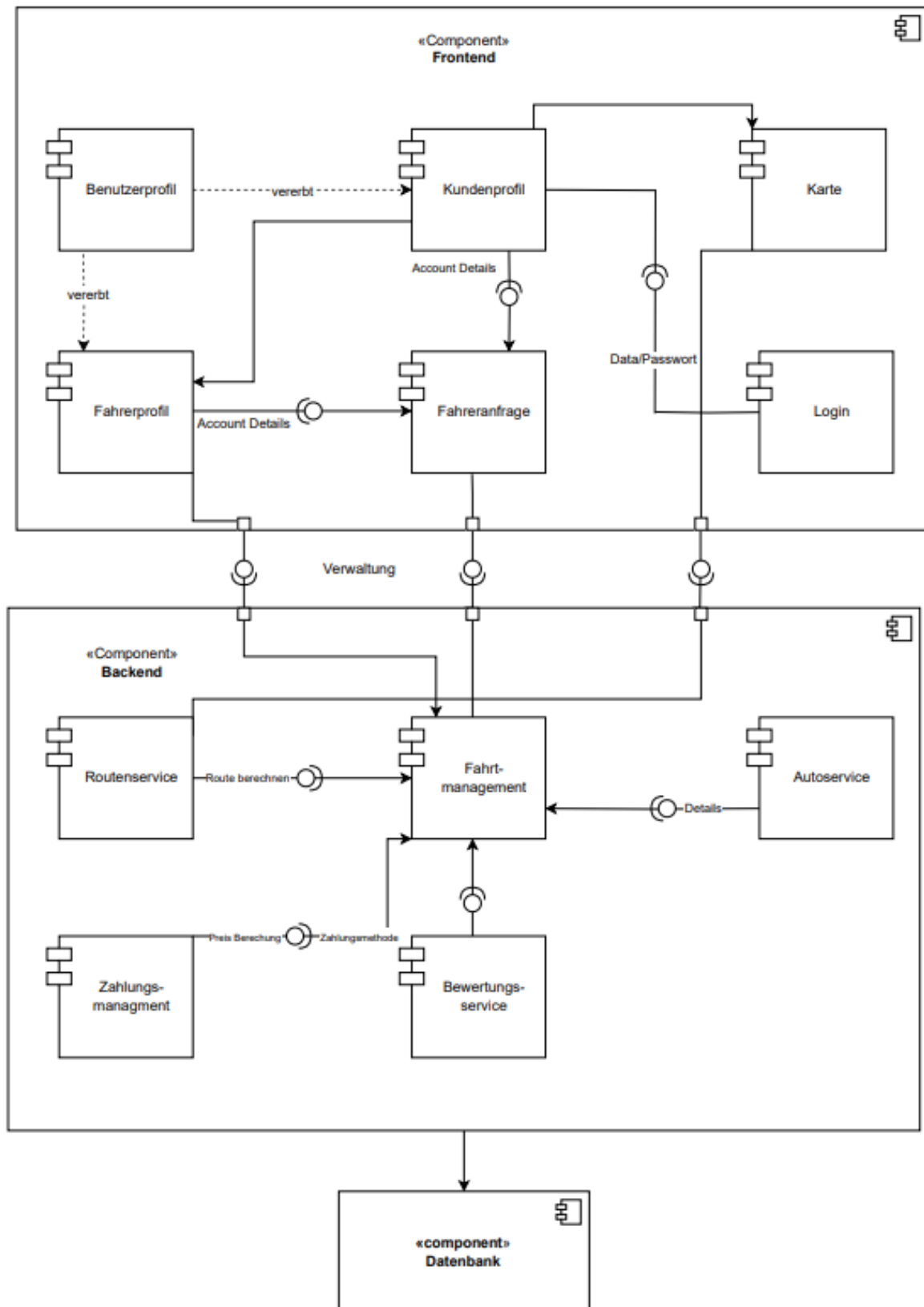


6.2 Dashboard-Fenster:



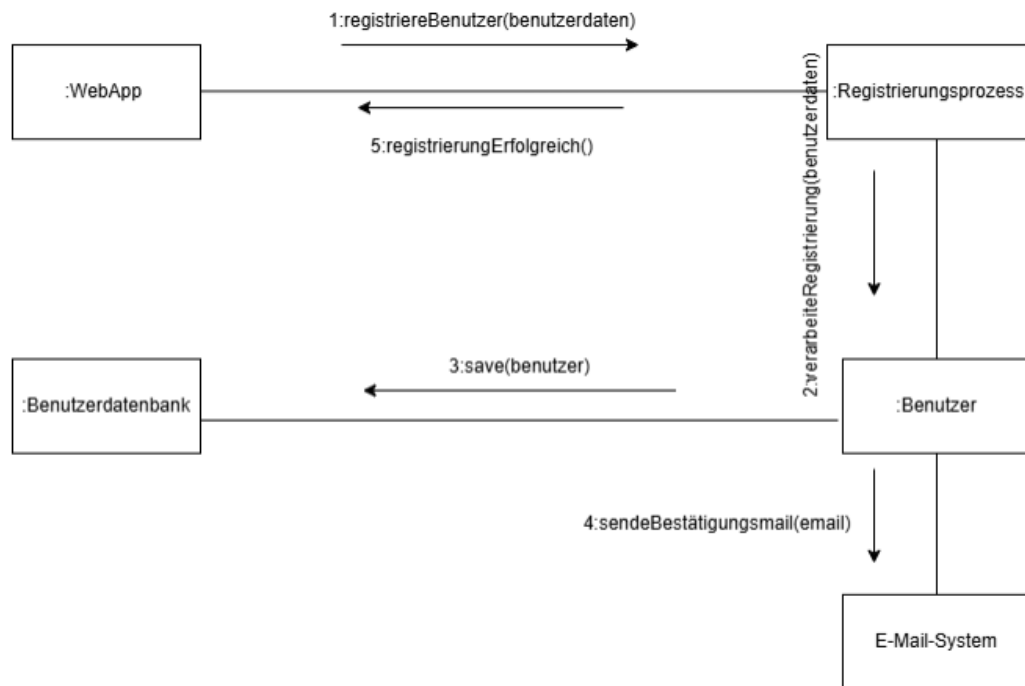
Strukturdiagramme (Komponenten- und Klassendiagramme)



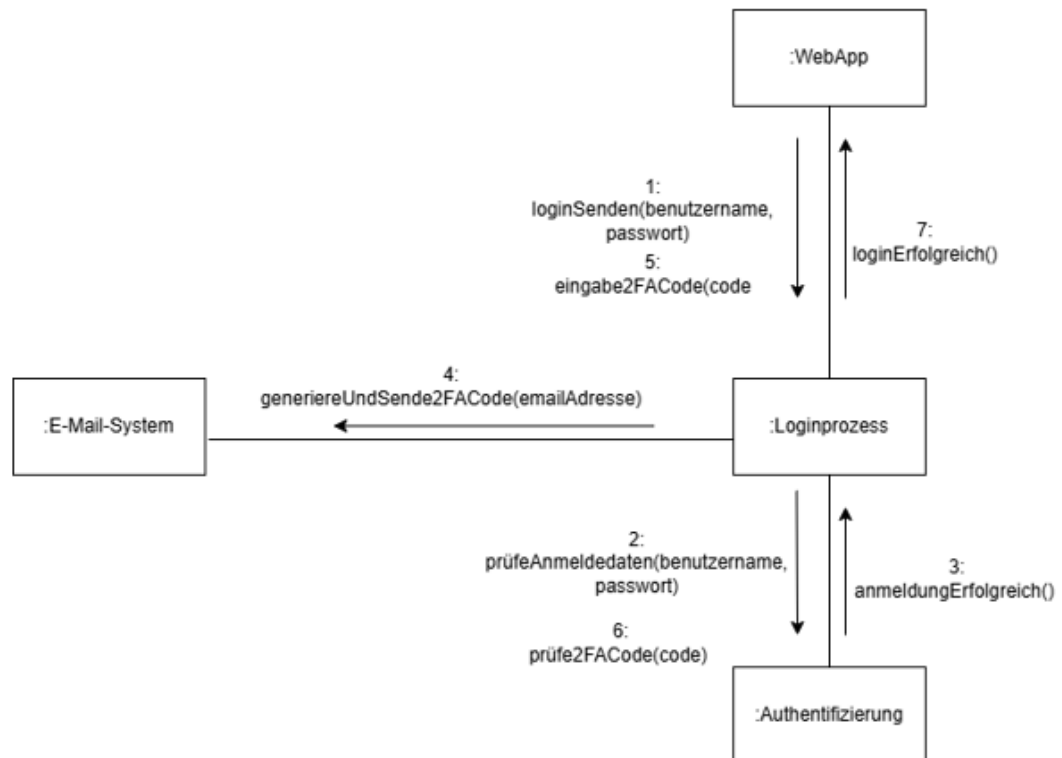


Verhaltensdiagramme (Kommunikationsdiagramme)

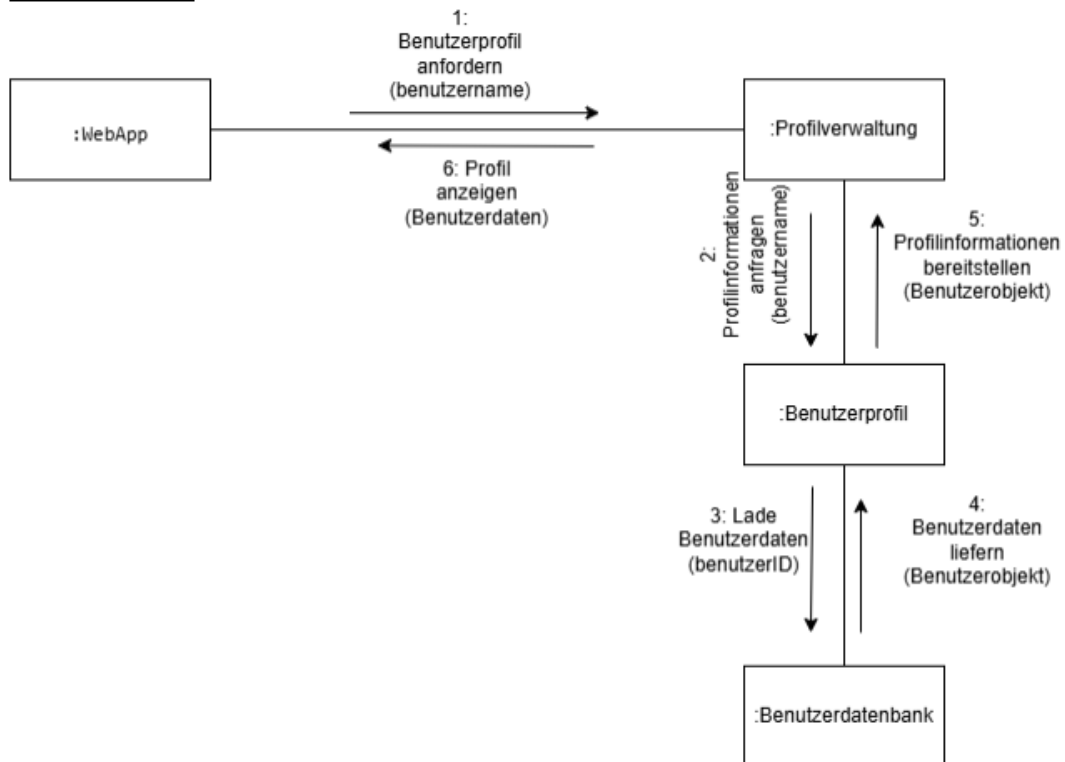
Registrierung eines Benutzers



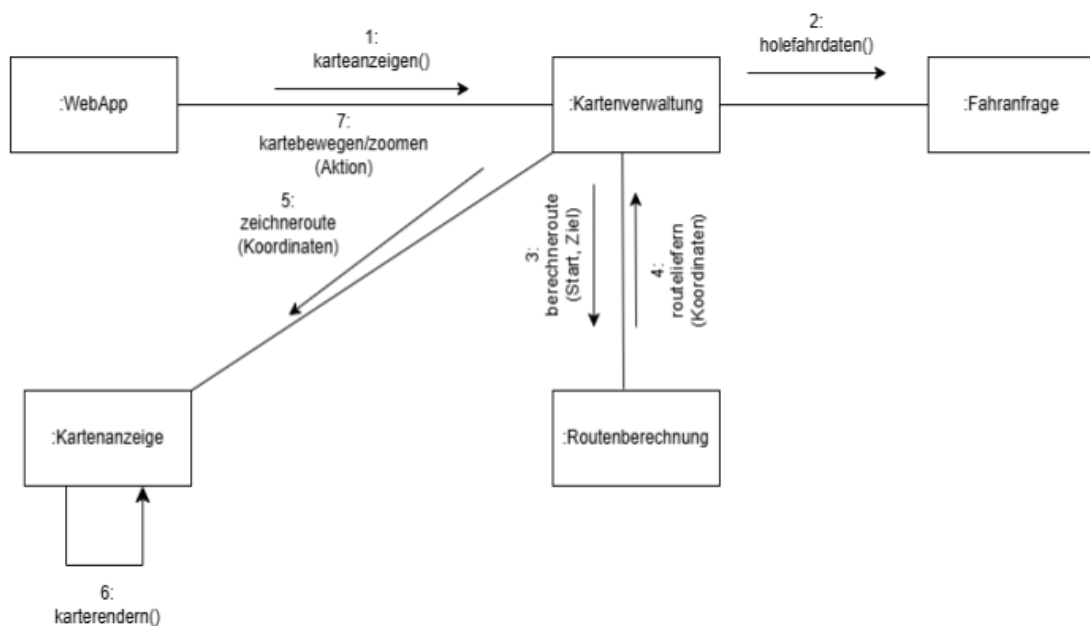
2FA Login



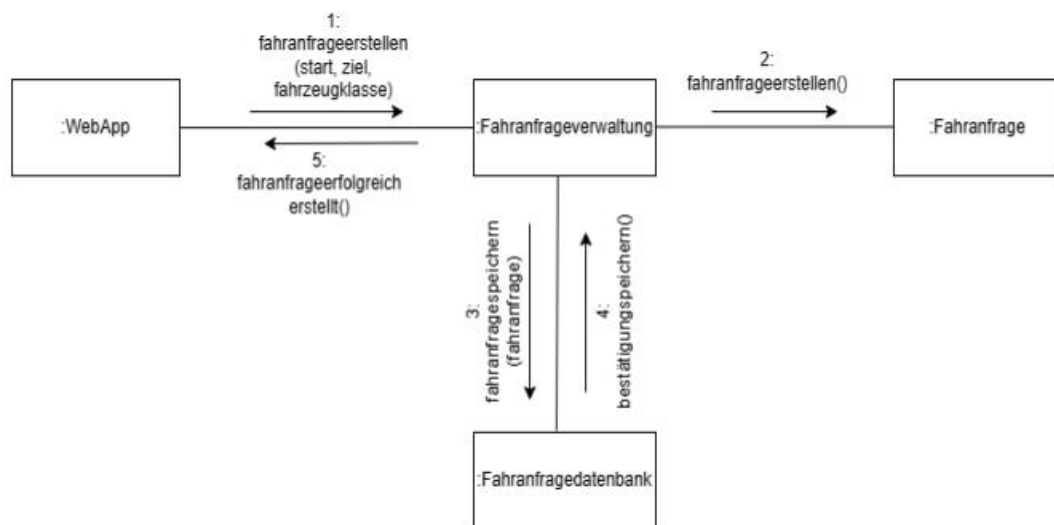
Benutzerprofil



Kartenvisualisierung von Routen



Fahrplan erstellen



Funktionalitätsplanung

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
1.	Nutzer Registrierung					
1.1	Registrierung view im Frontend	Zheyuan	1.2	1.1	Frotend/src/app/pages/auth/register	Fertig
1.2	Registrierung im Backend	Mohamad	1.1	1.1	.config/Security Config .Controller/Auth Controller .dto/RegisterUserRequest, RegisterResponse, verfiyCodeRequest .entity/User .exceptions/DuplicateResourceException, GlobalExceptionHandler .repository/User Repository .service/UserService	Fertig
2.	Nutzer Login					
2.1	Login im Frontend	Zheyuan	2.2	2.1	Frotend/src/app/pages/auth/login	Fertig
2.2	Login im Backend	Mohamad	2.1	2.1	.dto/LoginRequest, LoginResponse, .service/UserService, EmailService .Models.enums/Car Class, Role	Fertig
2.3	Zwei-Faktor-Authentifizierung im Frontend	Zheyuan	2.1	2.2	Frotend/src/app/components/two-factor-auth	Fertig
3.	Nutzerprofile					
3.1	Nutzerprofil Frontend	Simon	3.2	3.1	Frontend/src/app/profile (außer Z. 34 - 44 in	Fertig

					profile.html und Z. 37 – 44 in profile.ts, von Zheyuan aus Zyklus 2)	
3.2	Nutzerprofil Backend	Tim	3.1	3.1	Backend/drive/src/java/com.sep.drive/userprofile	Fertig
4.	Kartenvisualisierung von Routen					
4.1	Visualisierung Frontend	Dennis	4.2	4.1, 4.2	Frontend/src/app/map-visualization	Fertig
4.2	Visualisierung Backend	Tim	4.1	4.1, 4.2	Backend/drive/src/java/com.sep.drive/CarRequest/Route	Fertig
5.	Fahranfrage					
5.1	Fahranfrage erstellen Frontend	Finn	5.2	5.1, 5.2	Frontend/src/app/create-ride-request	Fertig
5.2	Fahranfrage erstellen Backend	Björn	5.1	5.1, 5.2	Backend/drive/src/java/com.sep.drive/CarRequest	Fertig
5.3	Aktive Fahranfrage einsehen Frontend	Dennis	5.4	5.3	Frontend/src/app/active-ride-request-page	Fertig
5.4	Aktive Fahranfrage einsehen Backend	Björn	5.3	5.3	Backend/drive/src/java/com.sep.drive/CarRequest/CarRequestController	Fertig
5.5	Fahranfrage löschen Frontend	Finn	5.6	5.4	Frontend/src/app/active-ride-request-page	Fertig
5.6	Fahranfrage löschen Backend	Björn	5.5	5.4	Backend/drive/src/java/com.sep.drive/CarRequest/CarRequestController	Fertig
6	Dashboard					
6.1	Dashboard Fenster	Simon	1.1, 2.1, 3.1, 4.1, 5.1, 5.2, 5.3	6.1	Frontend/src/app/dashboard	Fertig
6.2	Navigationsleiste	Simon	3.1, 3.2	3.1	Frontend/src/app/navbar	Fertig
7	Frontend Services					

7.1	Auth	Zheyuan			Frontend/src/app/service/auth.service.ts	Fertig
7.2	Kartenvisualisierung	Dennis			Frontend/src/app/service/mapRouteService.service.ts	Fertig
7.3	Fahranfrage	Finn			Frontend/src/app/service/ride-request.service.ts	Fertig
7.3.1	Fahranfrage nach kundenID suchen	Dennis			Frontend/src/app/service/ride-request.service.ts -> getRideRequestById()	Fertig
7.3.2	Fahranfrage basierend auf KundenID löschen	Dennis			Frontend/src/app/service/ride-request.service.ts -> deleteRideRequestById()	Fertig
7.4	User	Simon			Frontend/src/app/service/user.service.ts (außer getUserIdByToken Methode (Zeile 24))	Fertig
8	Datenbank					
8.1	Datenbank einrichten	Mohamad			Backend/src/main/resources/application.properties	Fertig
8.2	Springboot	Mohamad			Backend/src/main/resources/application.properties	Fertig
9	Docker					
9.1	Dockerfile Frontend	Tim			Frontend/Dockerfile	Fertig
9.2	Dockerfile Backend	Björn			Backend/Dockerfile	Fertig
9.3	Dockercompose file	Finn			Gruppe-m/docker-compose.yml	Fertig
9.4	Nginx config	Dennis			Frontend/nginx.conf	Fertig

Systemtests

Datum	20.05.2025		
Tester	Chantal Jedrschyk		
SW-Version	V 1.0		
Vorbedingung(en)	Nutzer hat Verbindung zum System aufgebaut		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer klickt auf den Button „Register now“	Das System leitet den User auf die Seite Nutzer registrieren	✓
2	Der Benutzer gibt seine Daten ein	Die Daten werden in den Kästchen angezeigt	✓
3	Der Benutzer klickt auf „Register“.	Das System zeigt die Meldung „Registered successfully“ auf dem Display an und öffnet ein Eingabefeld für die Zwei-Faktor-Authentifizierung.	✓
4	Der Benutzer klickt auf „Confirm“	Das System leitet den User auf das Log In feld weiter	✓
Nachbedingung(en)	Nutzer kann mit Username und Password in Homepage als Kunde oder Fahrer reingehen		✓
Testurteil	Test bestanden.		

Datum	20.05.2025		
Tester	Chantal Jedrschyk		
SW-Version	V 1.0		
Vorbedingung(en)	Nutzer „simon“ ist registriert und besitzt gültige Zugangsdaten (Username + Password) und hat eine Verbindung zum System		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer gibt seinen Benutzernamen und Passwort ein.	Das System überprüft die Eingabe. Bei korrekten Daten wird ein Sicherheitscode zur Zwei-Faktor-Authentifizierung erzeugt und per Mail versendet.	✓
2	Der Benutzer gibt den Sicherheitscode ein	Das System überprüft den Code. Ist er korrekt, wird der Benutzer eingeloggt.	✓
3	Der Benutzer klickt auf den Button „Log In“ dann wird eingeloggt.	Das System leitet den Benutzer auf das Dashboard weiter.	✓
Nachbedingung(en)	Der Benutzer ist erfolgreich im System angemeldet und hat Zugriff auf seine Dashboard-Funktionen („Profil ansehen“, „Fahranfrage erstellen“, „Aktive Fahranfrage“)		✓
Testurteil	Test bestanden.		

Datum	20.05.2025		
Tester	Chantal Jedrschyk		
SW-Version	V 1.0		
Vorbedingung(en)	Nutzer „simon“ ist eingeloggt und befindet sich auf der Dashboard Seite. Ein weiterer Nutzer „hansi“ ist ebenfalls registriert im System. Der Nutzer „xxx“ ist nicht im System registriert.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer klickt in das Feld „Benutzername suchen“	Das System ist bereit für die Eingabe	✓
2	Der Benutzer gibt den Benutzernamen „simon“ ein	Der eingegebene Name wird im Suchfeld angezeigt	✓
3	Der Benutzer klickt auf „Suchen“.	Das System leitet den Nutzer auf die <u>eigene</u> Profilseite	✓
(Alternative) 2a	Der Benutzer gibt den Benutzername „hansi“ ein	Der eingegebene Name wird im Suchfeld angezeigt	✓
2a1	Der Benutzer klickt auf „Suchen“.	Das System leitet den Nutzer auf die Profilseite von „hansi“	✓
(Ausnahme) 2b	Der Benutzer gibt den Benutzernamen „xxx“ ein	Der eingegebene Name wird im Suchfeld angezeigt	✓
2b1	Der Benutzer klickt auf „Suchen“	Das System zeigt die Meldung an „Benutzer xxx wurde nicht gefunden.“	✓
Nachbedingung(en)			✓
Testurteil	Test bestanden.		

Datum	20.05.2025		
Tester	Chantal Jedrschyk		
SW-Version	V 1.0		
Vorbedingung(en)	Nutzer ist eingeloggt und befindet sich auf einer beliebigen Seite (außer Registrierung/Login)		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Benutzer klickt auf seinen Benutzernamen	Das System leitet den User auf die eigene Profilseite	✓
Nachbedingung(en)			✓
Testurteil	Test bestanden.		

Datum	20.05.2025		
Tester	Chantal Jedrschyk		
SW-Version	V 1.0		
Vorbedingung	Nutzer "simon" ist registriert, eingeloggt und befindet sich auf dem Dashboard		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X

1	Benutzer klickt auf "Fahranfrage erstellen" auf dem Dashboard	Das System leitet auf das Formular zur Erstellung einer Fahranfrage weiter	✓
2	Benutzer legt Start- und Zielpunkt der Fahrt sowie Fahrzeugklasse fest durch verschiedene Möglichkeiten die alle mit Koordinaten für Start sowie Ziel enden	Die Daten werden in den Kästchen angezeigt	✓
2 a (Alternative)	Benutzer gibt Koordinaten "manuell" ein aufgeteilt in Latitude (lat) und Longitude (lng)	Die Daten werden in den Kästchen angezeigt	✓
2 b (Alternative)	Benutzer nutzt Adresse suchen Funktion durch Eingabe der Adresse und Bestätigung des Buttons "Adresse suchen" darunter	Vorschläge anpassenden Adressvorschlägen passend zur Eingabe des Nutzers	✓
2 c (Alternative)	Benutzer nutzt beim Festlegen des Starts die "Aktuelle Position" durch Auswahl der Option und Bestätigung des Zugriffs auf den Standort durch den Browser.	Koordinaten werden übertragen in die Kästchen dafür	✓
2 d (Alternative)	Benutzer nutzt POI suche zur Festlegung von Start oder Ziel durch Eingabe von einem Ort und einer Kategorie (Theater, Restaurant, ...)	Vorschläge an POIs passend zur Eingabe des Nutzers	✓
3	Der Benutzer klickt auf Fahranfrage erstellen	Das System erstellt die Fahranfrage und leitet bei erfolgreicher Erstellung zum Dashboard zurück.	✓
3a (Ausnahme)	Nutzer legt nur einen Punkt fest, sprich nur Start oder Ziel	Das System fordert bei Betätigung des "Fahranfrage erstellen" Buttons den Nutzer auf gültige Start- und Zielkoordinaten festzulegen	✓

Nachbedingung(en)	Nutzer kann unter „aktive Fahranfrage“ seine erstellte Fahranfrage sehen	✓
Testurteil	Test bestanden.	

Datum	20.05.2025		
Tester	Chantal Jedrschyk		
SW-Version	V 1.0		
Vorbedingung	Nutzer „simon“ ist eingeloggt und hat eine Fahranfrage erstellt		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Benutzer klickt auf „Aktive Fahranfrage“ im Dashboard	Der Nutzer wird auf die Seite der aktiven Fahranfrage weitergeleitet. Ihm werden nun Start, Ziel (inklusive Koordinaten), Status und die gewählte Fahrzeugklasse seiner Fahranfrage angezeigt	✓
2	Benutzer klickt auf „Fahranfrage löschen“	Die Fahranfrage wird aus der Datenbank entfernt und der Benutzer sieht nun ein Fenster mit dem Text „Keine aktive Fahranfrage“	✓
Nachbedingung(en)			
Testurteil	Test bestanden.		

Datum	20.05.2025		
Tester	Chantal Jedrschyk		
SW-Version	V 1.0		
Vorbedingung	Nutzer „simon“ ist eingeloggt und hat noch keine Fahranfrage gestellt oder hat seine aktive Fahranfrage gelöscht		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Benutzer klickt auf „Aktive Fahranfrage“ im	Der Benutzer sieht ein Fenster mit dem Text „Keine aktive Fahranfrage“	✓

	Dashboard		
Nachbe- dingung(en)			
Testurteil	Test bestanden.		

Zyklus II

Spezifikationsplanung

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
1.	Geldkonten			
1.1	Kontoübersicht	User Story	Zheyuan	Fertig
1.1.1	Kundekontoübersicht-Fenster	Papierprototyp	Zheyuan	Fertig
1.1.2	Fahrerkontoübersicht-Fenster	Papierprototyp	Zheyuan	Fertig
1.2	Geld aufladen	User Story	Zheyuan	Fertig
1.2.1	Geldaufladen-Fenster	Papierprototyp	Zheyuan	Fertig
2.	Liste der verfügbaren Fahranfragen			
2.1	Tabelle der verfügbaren Fahranfragen	User Story	Simon	Fertig
2.1.1	Tabelle der verfügbaren Fahranfragen-Fenster	Papierprototyp	Finn	Fertig
2.2	Aktuelle Position	User Story	Dennis	Fertig
3.	Fahrtplanung			
3.1	Fahrtplanung	User Story	Simon	Fertig
3.1.1	Fahrtplanung-Fenster	Papierprototyp	Finn	Fertig
4.	Fahrtangebote			
4.1	Fahrangebot auswählen	User Story	Simon	Fertig
4.2	Kundenbenachrichtigung	User Story	Simon	Fertig
4.2.1	Kundenbenachrichtigung-Fenster	Papierprototyp	Finn	Fertig
4.3	Fahrangebot zurückziehen	User Story	Simon	Fertig
4.4	Fahrtangebote Liste	User Story	Simon	Fertig
5.	Durchführung einer Fahrt			
5.1	Fahrtsimulation (Karte)	User Story	Dennis	Fertig
5.1.1	Fahrtsimulation (Fenster)	Papierprototyp	Finn	Fertig
5.2	Simulation kontrollieren	User Story	Dennis	
5.3	Simulationsanzeige	User Story	Dennis	Fertig
5.4	Ende der Simulation	User Story	Dennis	Fertig
5.4.1	Ende der Simulation (Fenster)	Papierprototyp	Finn	Fertig
6.	Fahrthistorie			
6.1	Fahrthistorie	User Story	Zheyuan	Fertig
6.1.1	Fahrthistorie-Fenster	Papierprototyp	Zheyuan	Fertig
6.2	Suche nach Fahrer oder Kunde	User Story	Zheyuan	Fertig

6.2.1	Suchergebnisse	Papierprototyp	Zheyuan	Fertig
7.	Architektur			
7.1	Komponentendiagramm	Diagramm	Tim	Fertig
7.2	Klassendiagramm	Diagramm	Björn	Fertig
7.3	Kommunikationsdiagramm	Diagramm	Mohamad	Fertig

User-Stories

User Story-ID	1.1
User Story-Beschreibung	Als Nutzer möchte ich eine Übersicht über mein Geldkonto sehen können, damit ich weiß, wie viel Guthaben ich aktuell habe.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Zheyuan Wu
Abhängigkeiten zu anderen User Stories	1.2

User Story-ID	1.2
User Story-Beschreibung	Als Kunde habe ich die Möglichkeit, Geld auf mein Konto zu laden, damit ich Fahrten bezahlen kann.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	mittel
Autor	Zheyuan Wu
Abhängigkeiten zu anderen User Stories	1.1

User Story-ID	2.1
User Story-Beschreibung	Als Fahrer möchte ich die verfügbaren Fahranfragen als Tabelle angezeigt bekommen, die ich nach beliebigen Merkmalen sortieren kann, um mir die beste Fahrt auszusuchen.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	2.2

User Story-ID	2.2
User Story-Beschreibung	Als Fahrer möchte ich meine aktuelle Position eingeben können, indem ich entweder die aktuelle Position verwende, nach einer Adresse suche, nach

	Point of Interests suche, oder die genauen Koordinaten eingebe, um mir die Entfernung zum Startpunkt der Fahrten berechnen zu lassen
Geschätzter Realisierungsaufwand	1 Tag
Priorität	niedrig
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	2.1

User Story-ID	3.1
User Story-Beschreibung	Als Kunde möchte ich die geplante Route und Preis, Distanz und Dauer der Fahrt sowohl bei der Erstellung als auch während der Fahrt angezeigt bekommen, um besser planen zu können.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	mittel
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	5.1

User Story-ID	4.1
User Story-Beschreibung	Als Fahrer möchte ich mir eine Fahrt aus der Liste der verfügbaren Fahranfragen aussuchen, damit der Kunde dann mein Angebot annehmen kann und ich mit der Fahrt beginnen kann.
Geschätzter Realisierungsaufwand	1 Tag
Priorität	mittel
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	2.1, 2.2, 4.2

User Story-ID	4.2
User Story-Beschreibung	Als Kunde möchte ich eine Benachrichtigung bekommen, wenn ein Fahrer meine Fahrt durchführen will, und anhand seiner Bewertung, Anzahl von Fahrten und gefahrene Distanz entscheiden, ob ich das Angebot annehme oder ablehne, um nur von guten Fahrern gefahren werden.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	mittel
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	4.1

User Story-ID	4.3
User Story-Beschreibung	Als Fahrer möchte ich die Möglichkeit haben mein Fahrangebot zurückzuziehen, damit ich flexibel bin, falls eine bessere Fahranfrage gestellt

	wird
Geschätzter Realisierungsaufwand	3 Tage
Priorität	mittel
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	4.1

User Story-ID	4.4
User Story-Beschreibung	Als Kunde möchte ich, dass mir Informationen zu den Angeboten der Fahrer, die meine Fahrt durchführen wollen in einer Tabelle angezeigt werden, um mir den besten Fahrer auszusuchen.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	mittel
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	4.2

User Story-ID	5.1
User Story-Beschreibung	Als Nutzer möchte ich die Möglichkeit haben den Status der derzeitigen Fahrt auf einer Karte zu verfolgen.
Geschätzter Realisierungsaufwand	4 Tage
Priorität	hoch
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	

User Story-ID	5.2
User Story-Beschreibung	Als Nutzer möchte ich die Simulation der Fahrt starten, pausieren und fortsetzen können sowie die Geschwindigkeit der Simulation anpassen können.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	5.1

User Story-ID	5.3
User Story-Beschreibung	Als Nutzer möchte ich während der Simulation einen Marker auf der Karte sehen, der sich auf der Karte vom Start- zum Endpunkt bewegt, um die aktuelle Position zu sehen
Geschätzter Realisierungsaufwand	2 Tage
Priorität	Mittel

Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	5.1, 5.2

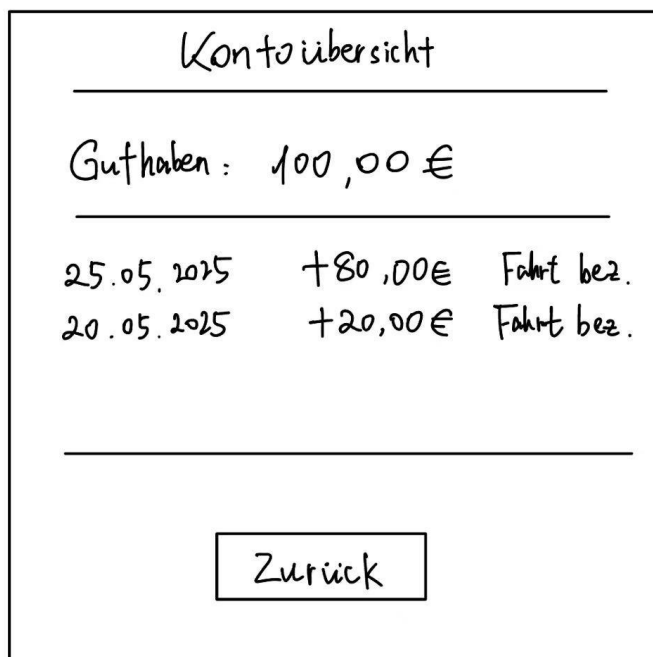
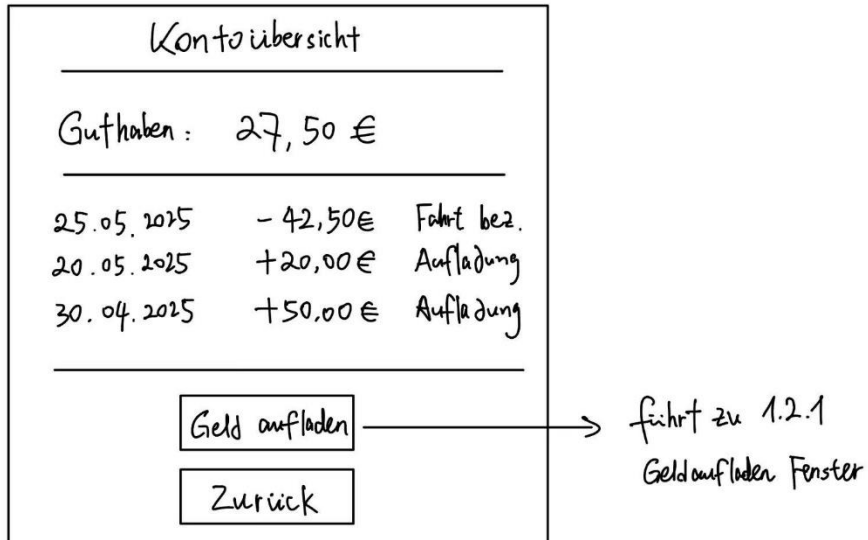
User Story-ID	5.4
User Story-Beschreibung	Als Nutzer möchte ich nach dem Abschluss einer Fahrt den anderen Teilnehmer der Fahrt bewerten können, um meine Zufriedenheit auszudrücken.
Geschätzter Realisierungsaufwand	0,5 Tage
Priorität	niedrig
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	5.1

User Story-ID	6.1
User Story-Beschreibung	Als Nutzer möchte ich meine Fahrthistorie in Tabellenform einsehen können, damit ich einen Überblick über meine vergangenen Fahrten, Zahlungen, Bewertungen usw. habe.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Zheyuan Wu
Abhängigkeiten zu anderen User Stories	2.1, 5.1

User Story-ID	6.2
User Story-Beschreibung	Als Nutzer möchte ich in meiner Fahrthistorie gezielt nach dem Namen eines Fahrers oder Kunden suchen können, damit ich schnell die Fahrten mit einer bestimmten Person einsehen kann.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	mittel
Autor	Zheyuan Wu
Abhängigkeiten zu anderen User Stories	5.1, 6.1

Papierprototypen

1.1.1 Kundekontoübersicht-Fenster + 1.1.2 Fahrerkontoübersicht-Fenster:



1.2.1 Geldaufladen-Fenster:

Geld aufladen

Betrag eingeben:

€

Aufladen

Zurück

2.1.1 Tabelle verfügbarer Fahrfragen Fenster:

Verfügbare Fahrfragen:

ID	Datum / Uhrzeit	Entfernung zu Kunde	Kundenname	Kundenbewertung	Fahrzeugklasse
9	02.03.25 13:06	76km	Bijan B.	★★★★★	Deluxe
11	02.03.25 13:07	44km	Peter R.	★★★★★	Medium
31	02.03.25 13:04	20km	Strobel S.	★★★★★	klein
1	02.03.25 13:06	1km	Ecker E.	★★★★★	Deluxe

Deine aktuelle Position:

Lat:
 Lng:

Position aktualisieren

Position aktualisieren

aktuelle Position

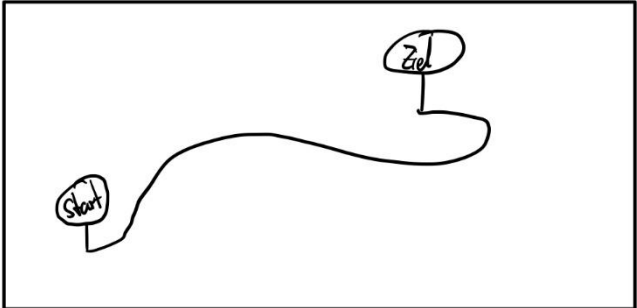
Adresse Suchen

POI Suche

Koordinaten

3.1.1 Fahrtplanung Fenster (Nur die Erweiterung zu vorher):

Routenplanung:



Gesamtdistanz: 71 km
Gesamtdauer: 60 min
Preis: 60 €

Anfrage erstellen

4.2.1 Kundenbenachrichtigung-Fenster

Angebote:

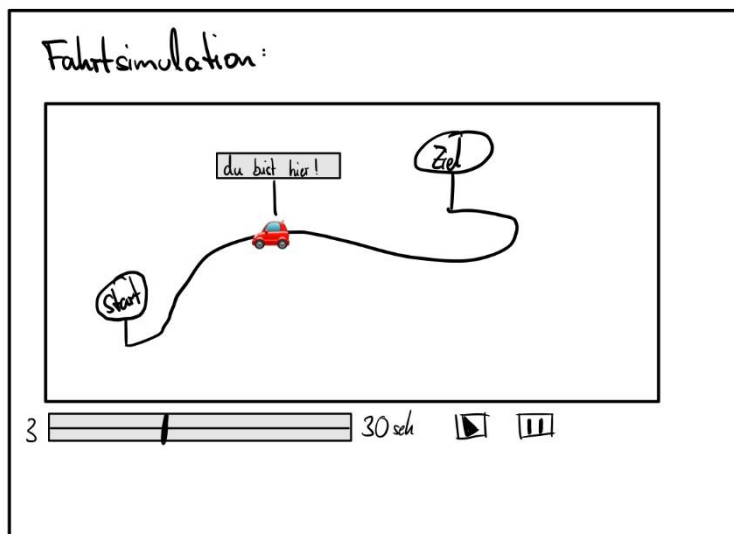
Fahrer	Bewertung	Fahrten	Distanz ges.	Sortieren
Biyan. B	★★★★★	1	2 km	Annehmen
Peter. S	★★★★☆	85	600 km	Annehmen

Gesamtstrecke

Aufsteigend

Absteigend

5.1.1 Simulierte Fahrt-Fenster



5.4.1 Ende der Simulation-Fenster:



6.1.1 Fahrthistorie-Fenster (Kunde):

Fahrthistorie

Fahrt-ID	Datum/ Uhrzeit	Distanz	Dauer	Betrag	Bew. Kunde	Bew. Fahrer	Kunde (Name)	Kunde (User name)	Fahrer (Name)	Fahrer (User Name)
101	12.05.2015 10:00	12 km	25 min	35,00€	Jan Renner	Jan Renner	Jan Renner	Viper	Shi Liu	Alex
102	10.05.2015 18:30	3 km	10 min	15,00€	Jan Renner	Jan Renner	Jan Renner	Viper	Ding Wang	Top
105	15.05.2015 09:45	1 km	2 min	5,00€	Jan Renner	Jan Renner	Jan Renner	Viper	Ding Wang	Top
111	18.05.2015 15:15	22 km	40 min	80,00€	Jan Renner	Jan Renner	Jan Renner	Viper	ale AAA	Nils
127	20.05.2015 22:00	2 km	6 min	7,00€	Jan Renner	Jan Renner	Jan Renner	Viper	bb BBB	Georg

Seite : 1 2 3 ...

6.2.1 Suchergebnisse:

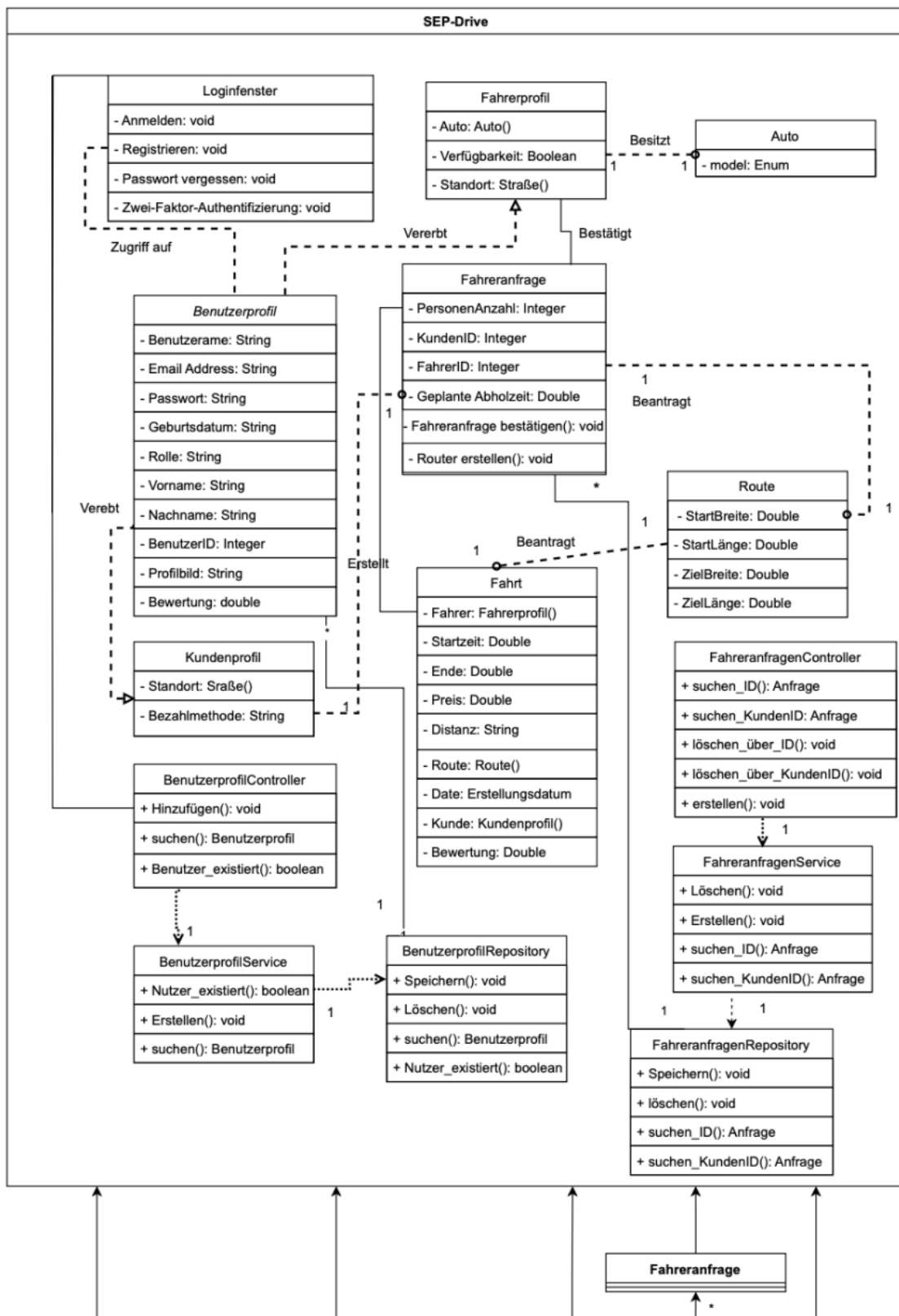
Fahrthistorie

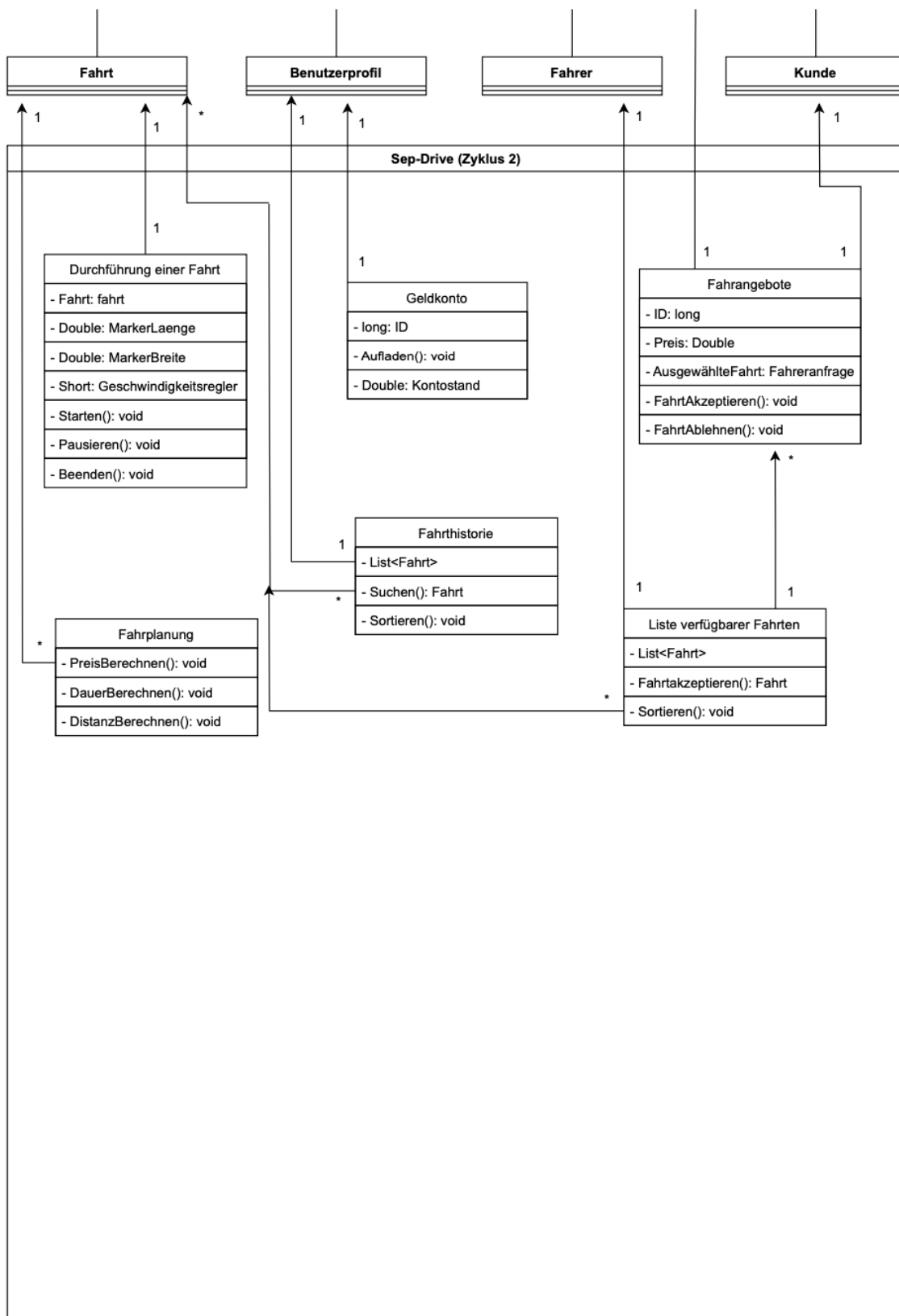
Klicken

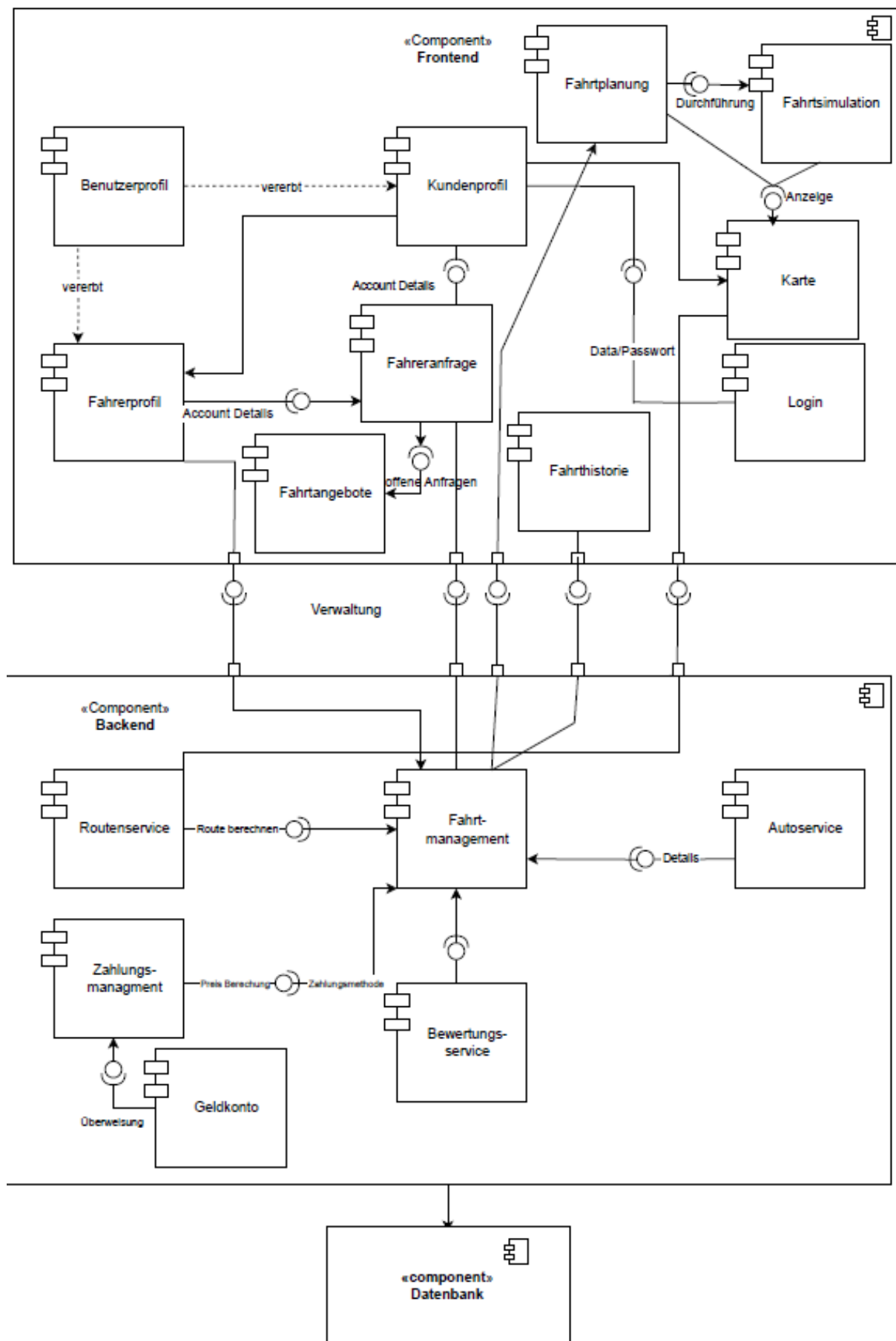
Fahrt-ID	Datum/ Uhrzeit	Distanz	Dauer	Betrag	Bew. Kunde	Bew. Fahrer	Kunde (Name)	Kunde (User name)	Fahrer (Name)	Fahrer (User Name)
102	10.05.2015 18:30	3 km	10 min	15,00€	Jan Renner	Jan Renner	Jan Renner	Viper	Ding Wang	Top
105	15.05.2015 09:45	1 km	2 min	5,00€	Jan Renner	Jan Renner	Jan Renner	Viper	Ding Wang	Top

Seite : 1 2 3 ...

Strukturdiagramme (Komponenten- und Klassendiagramme)

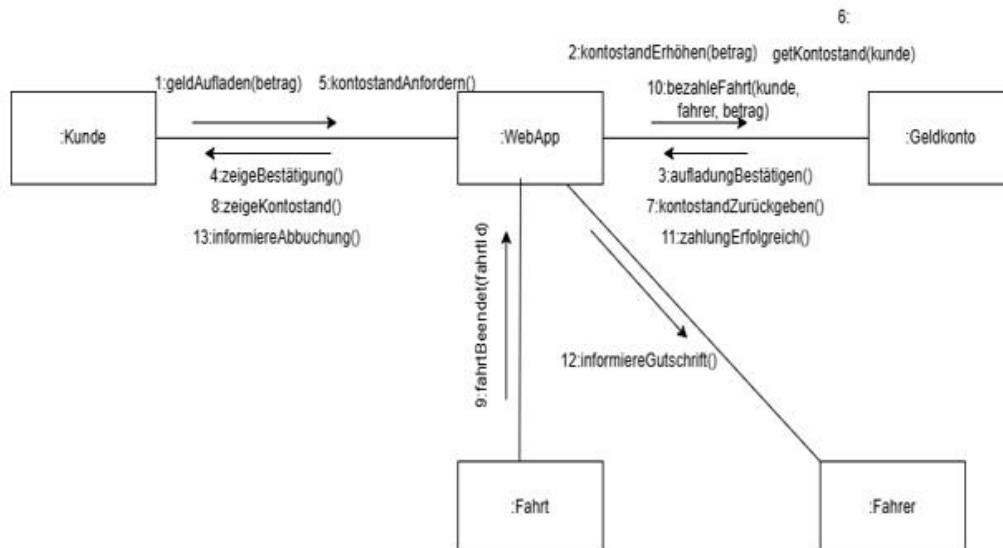




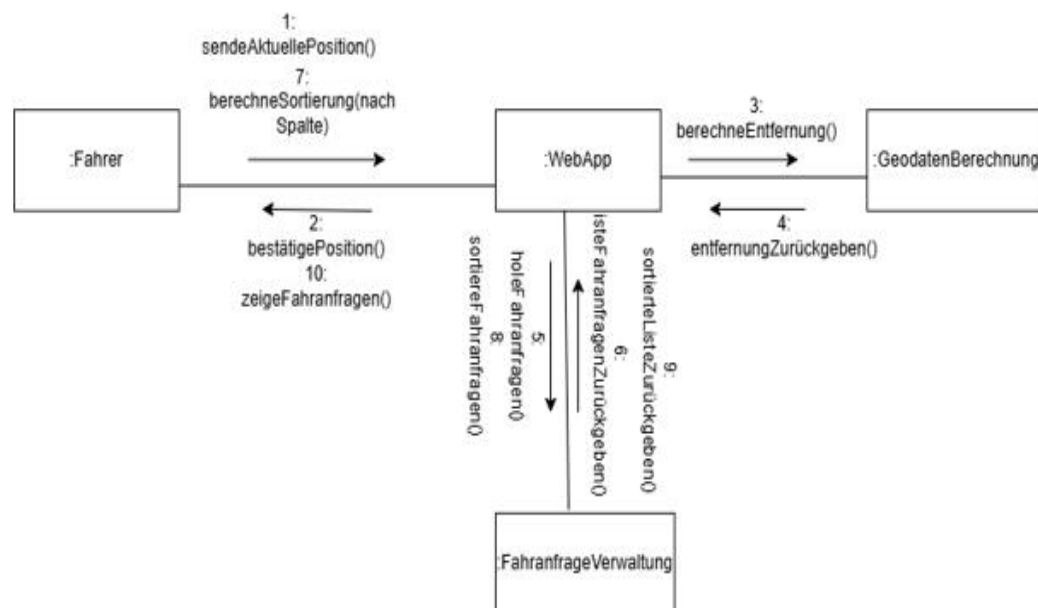


Verhaltensdiagramme (Kommunikationsdiagramme)

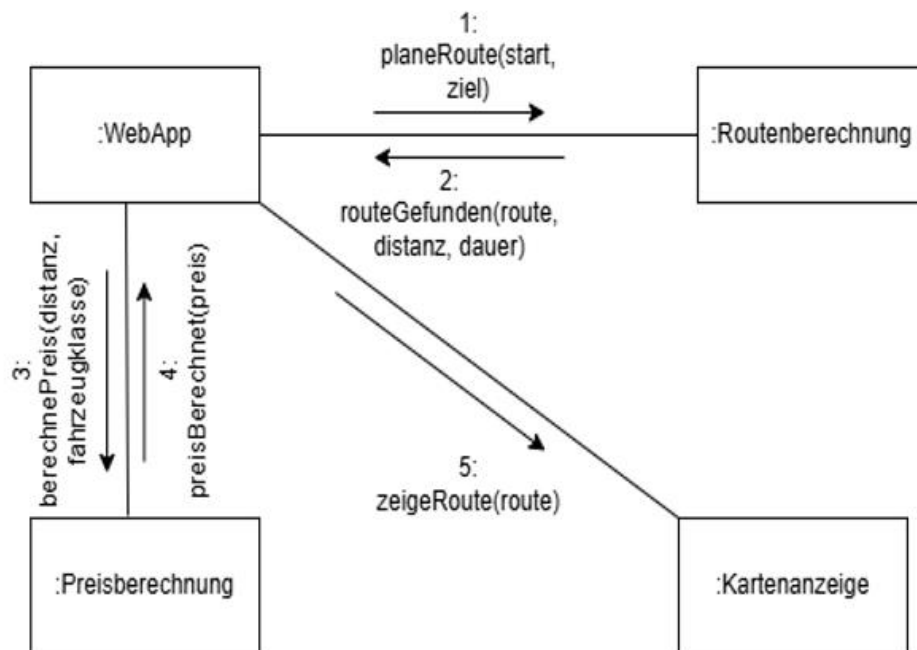
Verwaltung der Geldkonten:



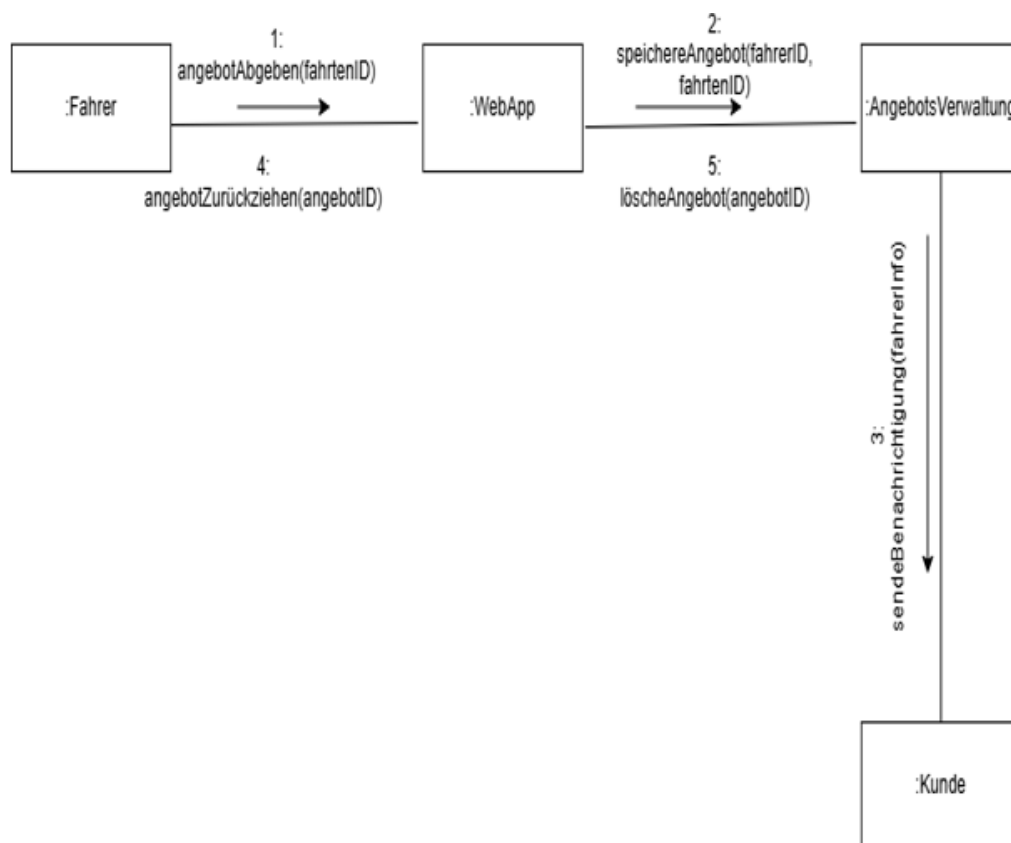
Liste der verfügbaren Fahranfragen:



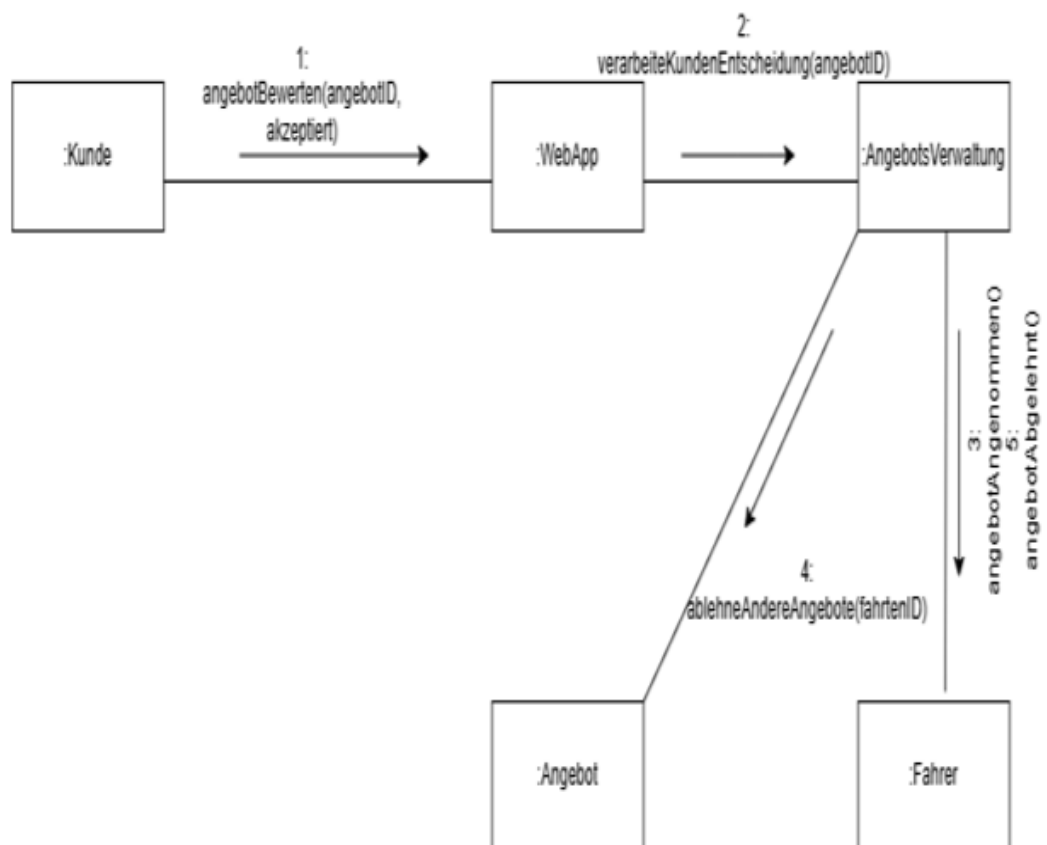
Fahrtplanung mit Routen:



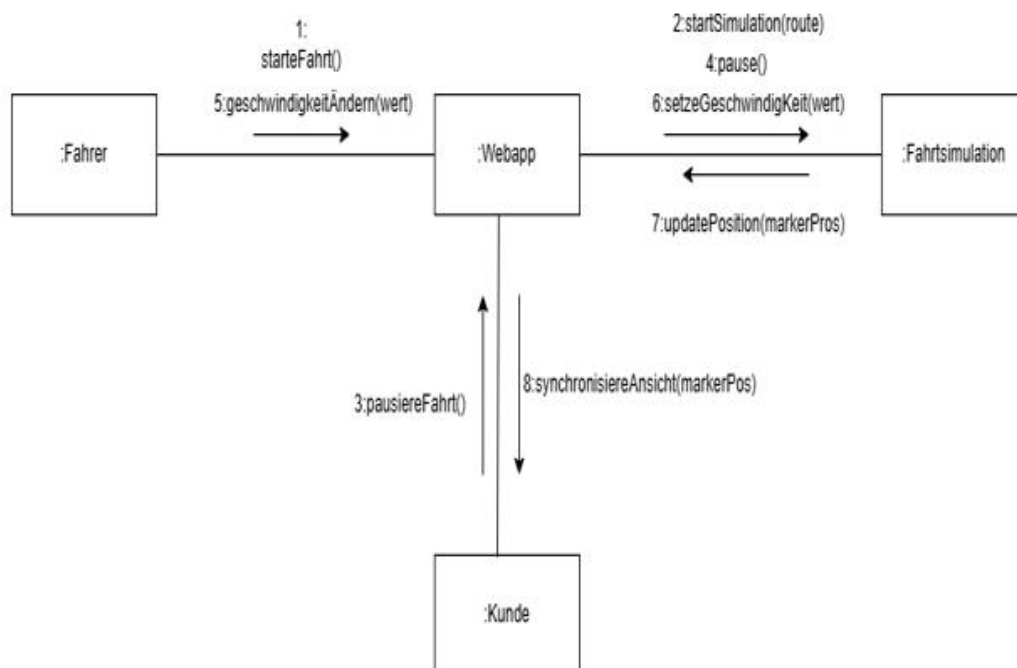
Fahrer erstellt Fahrtangebot:



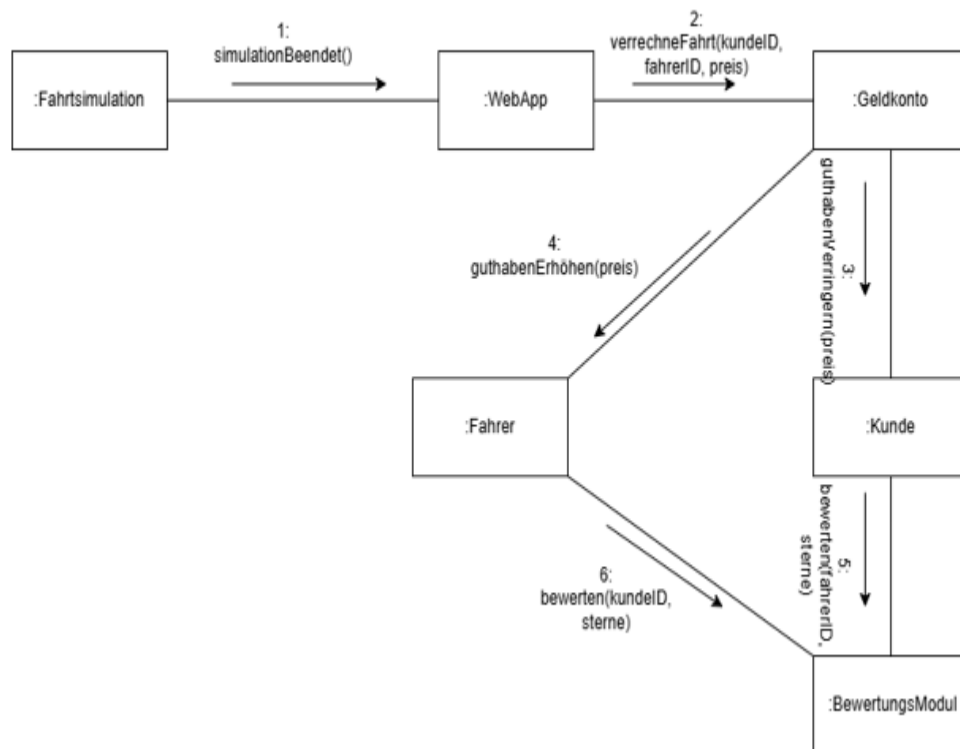
Kunde entscheidet über Fahrerangebote:



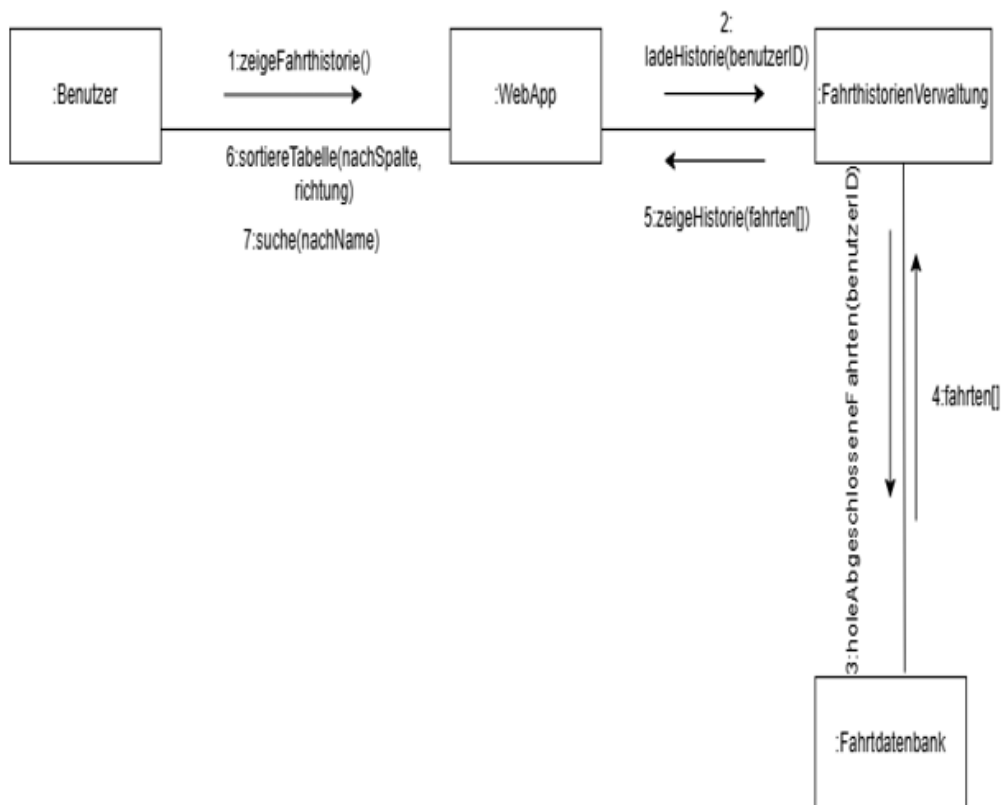
Synchrone Fahrtsimulation und Steuerung:



Abschluss der Fahrt mit Zahlung und Bewertung:



Anzeige und Verwaltung der Fahrthistorie:



Funktionalitätsplanung

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
1.	Geldkonten					
1.1	Geldkonten Frontend	Zheyuan	1.2	1.1, 1.2	Frotend/src/app/components/user-balance	Fertig
1.2	Geldkonten Backend	Mohamad	1.1	1.1, 1.2	Backend/src/main/java/com.sep.drive/controller/WalletController Backend/src/main/java/com.sep.drive/dto/BalanceResponse Backend/src/main/java/com.sep.drive/dto/TopUpRequest Backend/src/main/java/com.sep.drive/CompleteRideRequest Backend/src/main/java/com.sep.drive/service/RideService	Fertig
2.	Verfügbare Fahranfragen					
2.1	Liste Frontend	Simon	2.2	2.1, 2.2, 4.3	Frontend/src/app/ride-request-table (außer Methoden in ride-request-table.component.ts Z. 174 – 230 und Z. 17 – 116 in ride-request-table.component.html, von Finn aus create-ride-request.compon	Fertig

					ent.ts übernommen für Startpunktauswahl)	
2.2	Liste Backend	Björn	2.1	2.1, 2.2, 4.3	Backend/src/main/java/com.sep.drive/CarRequest/CarRequestRepository(Ab Z.24)+CarRequestService(Ab Z.42)+CarRequestController(Ab Z.86)+CarRequestAvailable+CarRequest(Zusätzliche Attribute)	Fertig
3.	Fahrtplanung					
3.1	Fahrtplanung Frontend	Finn	3.2	3.1	Fontend/src/app/create-ride.request	Fertig
3.2	Fahrtplanung Backend	Tim	3.1	3.1	Backend/src/main/java/com.sep.drive/Fahrtplanung/RideRequestCal (wird doch nicht benutzt)	Fertig
4.	Fahrangebote					
4.1	Kundenbenachrichtigung Frontend	Finn	4.2, 4.3, 4.4	4.2	Fontend/src/app/ride-offer-page	Fertig
4.2	Kundenbenachrichtigung Backend	Mohamad	4.1, 4.3, 4.4	4.2	Backend/src/main/java/com.sep.drive/controller/OfferController Backend/src/main/java/com.sep.drive/dto/CreateOfferRequest Backend/src/main/java/com.sep.drive/dto/OfferResponse Backend/src/main/java/com.sep.drive/service/OfferService	Fertig

					Backend/src/main/java/com.sep.drive/repository/OfferRepository	
4.3	Angebotsliste Frontend	Finn	4.4	4.1, 4.4	Frontend/src/app/ride-offer-page	Fertig
d						
4.4	Angebotsliste Backend	Tim	4.3	4.1, 4.4	Backend/src/main/java/com.sep.drive/RideOffer (Kompletter Ordner)	Fertig
5.	Durchführung einer Fahrt					
5.1	Fahrtdurchführung Frontend	Dennis	5.2, 5.3, 5.4	5.1, 5.2, 5.3	Frontend/src/app/simulation	Fertig
5.2	Fahrtdurchführung Backend	Dennis	5.1, 5.3, 5.4	5.1, 5.2, 5.3	Backend/drive/src/main/java/com.sep.drive/controller/SimulationController + Backend/drive/src/main/java/com.sep.drive/service/SimulationService	Fertig
5.3	Fahrtende Frontend	Dennis	5.1, 5.2, 5.4	5.4	Frontend/src/app/simulation + Frontend/src/app/rating-popup	Fertig
5.4	Fahrtende Backend	Dennis	5.1, 5.2, 5.3	5.4	Backend/drive/src/main/java/com.sep.drive/controller/SimulationController + Backend/drive/src/main/java/com.sep.drive/ser	Fertig

					vice/Simulation Service	
6.	Fahrthistorie					
6.1	Fahrthistorie Frontend	Zheyuan	6.2	6.1, 6.2	Frontend/src/app/pages/ride-history	Fertig
6.2	Fahrthistorie Backend	Mohamad	6.1	6.1, 6.2	Backend/src/main/java/com.sep.drive/controller/RideHistoryController Backend/src/main/java/com.sep.drive/dto/PagedRideHistoryResponse Backend/src/main/java/com.sep.drive/dto/RideHistoryEntry Backend/src/main/java/com.sep.drive/service/OfferService	Fertig
7.	Frontend Services					
7.1	Fahrerfragen Service	Simon			Frontend/src/app/service/driver-ride-request.service.ts	Fertig
7.2	Fahrerfragen Data Source	Simon			Frontend/src/app/service/ride-request.dataSource.ts	Fertig
7.3	Geldkonten	Zheyuan			Frontend/src/app/service/balance.service.ts	Fertig
7.4	Fahrthistorie	Zheyuan			Frontend/src/app/service/ride-history.service.ts	Fertig

7.5	Wallet	Dennis			Frontend/src/app/service/wallet.service.ts	Fertig
7.6	WebSocket	Dennis			Frontend/src/app/service/websocket.service.ts	Fertig
8.0	Sonstiges					
8.1	Zusätzliche Suchmethoden für User	Björn			Backend/drive/src/main/java/com.sep.drive/userprofile/UserProfileController(Ab Z.52)+UserProfileService(Ab Z.43)+UserProfileRepository(Ab Z.17)+MergeName	Fertig
8.2	Alternativer Speicher für abgeschlossene Anfragen	Björn			Backend/drive/src/main/java/com.sep.drive/CompletedRequest	Fertig

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 für SMALL Fahrzeugklasse (calculateRidePrice)	sepm/Backend/drive/src/main/test/com/sep/drive/service/RideServiceTest.java	Fertig
2	Preisberechnung für MEDIUM Fahrzeugklasse (calculateRidePrice)	sepm/Backend/drive/src/main/test/com/sep/drive/service/RideServiceTest.java	Fertig
3	Preisberechnung für DELUXE Fahrzeugklasse (calculateRidePrice)	sepm/Backend/drive/src/main/test/com/sep/drive/service/RideServiceTest.java	Fertig
4	Fehlerbehandlung bei null-Fahrzeugklasse (calculateRidePrice)	sepm/Backend/drive/src/main/test/com/sep/drive/service/RideServiceTest.java	Fertig
5	Komplette Fahrt mit SMALL und ausreichendem Guthaben (completeRide)	sepm/Backend/drive/src/main/test/com/sep/drive/service/RideServiceTest.java	Fertig
6	Komplette Fahrt mit MEDIUM und ausreichendem Guthaben (completeRide)	sepm/Backend/drive/src/test/java/com/sep/drive/service/RideServiceTest.java	Fertig
7	Komplette Fahrt mit DELUXE und ausreichendem Guthaben (completeRide)	sepm/Backend/drive/src/main/test/com/sep/drive/service/RideServiceTest.java	Fertig
8	Fehlerfall bei unzureichendem Guthaben beim Kunden (completeRide)	sepm/Backend/drive/src/main/test/com/sep/drive/service/RideServiceTest.java	Fertig

Systemtests

Datum	17.06.25		
Tester	Chantal Jedrschyk		
SW-Version	2.0		
Vorbedingung(en)	Der Nutzer ist als Fahrer eingeloggt und befindet sich auf der „Kunden Fahranfragen“ Seite. Es gibt mindestens Zwei Fahranfragen von Kunden.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Nutzer klickt auf ein beliebiges Element der Kopfzeile der Tabelle	Die Tabelleneinträge werden dementsprechend sortiert angezeigt	✓
2	Nutzer legt Standort/Startpunkt durch verschiedene Auswahlmöglichkeiten fest	Die jeweilige Entfernung zum Start der Fahrt wird in der entsprechenden Spalte angezeigt	✓
2 a (Alternative)	Nutzer gibt Koordinaten “manuell” ein, aufgeteilt in Breite (Lat) und Länge (Lng)	Die Daten werden in den Kästchen angezeigt	✓
2 b (Alternative)	Nutzer wählt Adresse suchen Funktion durch Eingabe der Adresse und Bestätigung des Buttons “Adresse suchen”	Adressvorschlägen passend zur Eingabe des Nutzers	✓
2 c (Alternative)	Benutzer nutzt die “Aktuelle Position” durch Auswahl der Option und Bestätigung des Zugriffs auf den Standort durch den Browser.	Der aktuelle Standort wird in Koordinatenform	✓
2 d (Alternative)	Benutzer nutzt POI-Suche durch Eingabe von einem Ort und einer Kategorie (Theater, Restaurant, ...)	Vorschläge an POIs passend zur Eingabe des Nutzers	✓
3	Der Nutzer klickt auf	Der Text des Buttons ändert sich zu „Angebot zurückziehen“, alle anderen Buttons sind	✓

	„Angebot abgeben“	ausgegraut	
4 (Alternative)	Der Nutzer klickt auf „Angebot zurückziehen“	Alle Buttons sind wieder klickbar und zeigen „Angebot abgeben“ an	✓
Nachbedingung(en)	Der Kunde sieht das Angebot des Fahrers in „Fahrangebote“		✓
Testurteil	Test bestanden		

Datum	17.06.2025		
Tester	Chantal Jedrschyk		
SW-Version	2.0		
Vorbedingung(en)	Der Nutzer ist als Kunde oder Fahrer eingeloggt und befindet sich auf der “Profil ansehen” Seite.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Nutzer (Kunde oder Fahrer) klickt auf den Button “Geldkonto”	Das Guthaben des aktuellen Nutzers wird in einem Popup-Fenster angezeigt.	✓
2	Kunde hat einen zusätzlichen Button “aufladen”, dann klickt er darauf	Ein neues Popup-Fenster zur Eingabe des Betrags wird angezeigt.	✓
2.1	Kunde gibt einen Betrag im Bereich (0, 10,000] ein und klickt auf “bestätigen”	Das Aufladen-Fenster schließt sich sofort, es wird zum Guthaben-Fenster zurückgekehrt, eine Erfolgsmeldung wird angezeigt und das aktualisierte Guthaben wird dargestellt.	✓
2.2	Kunde gibt keinen Betrag ein und klickt auf “zurück”	Das Aufladen-Fenster schließt sich sofort und es wird zum Guthaben-Fenster zurückgekehrt.	✓
3	Der Nutzer (Kunde oder Fahrer) klickt auf “zurück”	Das Guthaben-Fenster schließt sich sofort und Profil Seite bleibt.	✓
4	Der Nutzer (Kunde oder Fahrer) klickt auf “aktualisieren” (nicht unbedingt notwendig)	Das Guthaben des aktuellen Nutzers wird aktualisiert und angezeigt.	✓

Nachbedingung(en)	Nach dem Aufladen verfügt der Kunde über genügend Guthaben, um die zukünftigen Fahrtkosten zu bezahlen.	✓
Testurteil	Test bestanden	

Datum	17.06.2025		
Tester	Chantal Jedrschyk		
SW-Version	2.0		
Vorbedingung(en)	Der Nutzer ist als Kunde oder Fahrer eingeloggt und befindet sich auf der "Profil ansehen" Seite.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Nutzer (Kunde oder Fahrer) klickt auf den Button "Fahrthistorie"	Alle Fahrten, die mit dem Namen dieses Kunden übereinstimmen, werden auf der Tabelle angezeigt.	✓
2	Der Nutzer gibt den Namen eines anderen Nutzers ein, der nicht seiner eigenen Rolle entspricht, und klickt auf "suchen"	In der Tabelle bleiben nur die Einträge, die beide Namen enthalten, übrig.	✓
3	Der Nutzer klickt auf "zurücksetzen"	Die vorherige Suche wird zurückgesetzt und die Tabelle zeigt erneut alle Fahrten an.	✓
Nachbedingung(en)			
Testurteil	Test bestanden		

Datum	17.06.2025
Tester	Chantal Jedrschyk
SW-Version	2.0
Vorbedin-	Der Nutzer ist als Kunde im System eingeloggt und registriert, und befindet sich

gung(en)	auf der "Fahranfrage erstellen" Seite.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde legt Start sowie Zielkoordinaten durch verschiedene Methoden fest.	Koordinaten werden übernommen und Start sowie Ziel festgelegt	✓
2	Durch einen Klick auf den Button "Route Planen" neben dem "Fahranfrage erstellen" Button soll ein Preview der Route angezeigt werden	Es wird nun der erwartete Preis, die erwartete Fahrzeit bis zum Ziel, die Strecke bis zum Ziel als auch eine Karte mit der Route unten eingeblendet.	✓
2 a	Kunde kann durch Zoomen etc. Auf der Karte navigieren	Karte zoomt raus/rein	✓
3	Kunde drückt auf "Fahranfrage erstellen"	Die Fahranfrage wird erstellt und der Kunde wird weitergeleitet auf die Aktive-Fahranfrage-Seite wo er auch nochmal Strecke, Preis und alle wichtigen Informationen zu seiner Anfrage erhält	✓
Nachbedingung(en)			
Testurteil	Test bestanden		

Datum	17.06.2025		
Tester	Chantal Jedrschyk		
SW-Version	2.0		
Vorbedingung(en)	Nutzer ist als Kunde im System eingeloggt, registriert und hat bereits eine Fahranfrage erstellt.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde klickt im	Kunde wird zur Tabelle mit den	✓

	Dashboard auf "Fahrangebote"	Fahrtangeboten weitergeleitet.	
1 a	Kunde klickt auf den Pfeil neben dem Titel der entsprechenden Spalte	Falls bereits mehrere Angebote zur Fahranfrage vorliegen werden diese nach der entsprechenden Spalte sortiert. Aufsteigend oder absteigend.	✓
2	Fahrer erstellt auf der Fahranfragen Liste ein Angebot zu der passenden Fahranfrage des Kunden	Kunde bekommt ein Pop-up dass es ein neues Angebot zu seiner Fahranfrage gibt.	✓
2 a	Kunde klickt in der Tabelle unter "Aktionen" auf Annehmen Fahrangebots	Das ausgewählte Angebot wird auf aktiv gestellt und zeitgleich werden alle anderen noch offenen Angebote der Fahranfrage abgelehnt.	✓
2 b	Kunde klickt in der Tabelle unter "Aktionen" auf Ablehnen Fahrangebots	Das ausgewählte Angebot wird abgelehnt und aus der Tabelle gelöscht. Der Fahrer kann nun ein neues Angebot erstellen.	✓
Nachbedingung(en)			
Testurteil	Test bestandne		

Datum	17.06.2025		
Tester	Chantal Jedrschyk		
SW-Version	2.0		
Vorbedingung(en)	Ein Kunde und ein Fahrer sind gleichzeitig in unterschiedlichen Fenstern angemeldet. Der Kunde hat eine Fahranfrage erstellt, die vom Fahrer angenommen wurde. Beide befinden sich auf der Fahrtsimulations-seite. Der Kunde hat genug Guthaben aufgeladen		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Fahrer oder Kunde drücken auf Start	Ein Marker erscheint, der die Route entlangfährt	✓
2	Fahrer oder Kunde drückt erneut auf Start	Der Marker bewegt sich weiter, ohne das etwas passiert	✓

3	Kunde drückt auf den "Pause" button	Der Marker bleibt sowohl beim Kunden als auch beim Fahrer stehen. Der "Pause" Button zeigt nun sowohl beim Kunden als auch beim Fahrer "continue" an.	✓
4	Kunde drückt auf "continue"	Der Marker fährt ab dort, wo pausiert wurde, die Route weiter entlang	✓
5	Fahrer drückt auf den "Pause" button	Der Marker bleibt sowohl beim Kunden als auch beim Fahrer stehen. Der "Pause" Button zeigt nun sowohl beim Kunden als auch beim Fahrer "continue" an.	✓
6	Fahrer drückt auf "continue"	Der Marker fährt ab dort, wo pausiert wurde, die Route weiter entlang	✓
7	Der Kunde ändert die Simulationsgeschwindigkeit per Schieberegler	Der Marker bewegt sich nun schneller bzw. langsamer, je nachdem, ob die Simulationsgeschwindigkeit verringert oder erhöht wurde. Beim Fahrer wird die neue Simulationsgeschwindigkeit ebenfalls angezeigt	✓
8	Der Fahrer ändert die Simulationsgeschwindigkeit per Schieberegler	Der Marker bewegt sich nun schneller bzw. langsamer, je nachdem, ob die Simulationsgeschwindigkeit verringert oder erhöht wurde. Beim Kunde wird die neue Simulationsgeschwindigkeit ebenfalls angezeigt	✓
Nachbedingung(en)			
Testurteil	Test bestanden		

Datum	17.06.2025		
Tester	Chantal Jedrschyk		
SW-Version	2.0		
Vorbedingung(en)	Ein Kunde und ein Fahrer sind gleichzeitig in unterschiedlichen Fenstern angemeldet. Der Kunde hat eine Fahranfrage erstellt, die vom Fahrer angenommen wurde. Beide befinden sich im Simulationsfenster. Der Marker hat das Ziel erreicht. Ein Bewertungsfenster ist erschienen. Beide Profile besitzen noch keine Bewertungen		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X

1	Der Kunde verändert den Schieberegler und drückt auf "bewerten"	Der übergebene Wert wird nun auf dem Profil des Fahrers unter "Bewertung" angezeigt	✓
2	Der Fahrer verändert den Schieberegler und drückt auf "bewerten"	Der übergebene Wert wird nun auf dem Profil des Kunden angezeigt	✓
3	Der Kunde drückt auf "Abbrechen"	Es wird keine Bewertung an den Fahrer übergeben. Unter "Bewertung" auf dem Profil des Fahrers steht weiterhin 0.	✓
4	Der Fahrer drückt auf "Abbrechen"	Es wird keine Bewertung an den Kunden übergeben. Unter "Bewertung" auf dem Profil des Kunden steht weiterhin 0.	✓
Nachbedingung(en)	Kunde und Fahrer sehen auf der Seite nur noch ein Fester mit dem Text "keine aktive Fahranfrage"		✓
Testurteil	Test bestanden		

Datum	17.06.25		
Tester	Chantal Jedrschyk		
SW-Version	2.0		
Vorbedingung(en)	Ein Kunde und ein Fahrer sind gleichzeitig in unterschiedlichen Fenstern angemeldet. Der Kunde hat eine Fahranfrage erstellt, die vom Fahrer angenommen wurde. Die gewünschte Fahrzeugklasse stimmt mit der des Fahrers überein. Beide befinden sich auf der Fahrtsimulations-seite. Der Kunde hat nicht genug Guthaben aufgeladen		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Kunde drückt auf "start"	Der Kunde erhält eine Benachrichtigung, die ihn darüber aufklärt, dass sein Guthaben nicht ausreicht. Die Simulation startet nicht	✓
2	Der Fahrer drückt auf "start"	Der Fahrer erhält eine Benachrichtigung, die ihn darüber aufklärt, dass das Guthaben des Kunden nicht ausreicht. Die Simulation startet nicht	✓
3	Der Kunde geht auf sein Profil und lädt das nötige Guthaben auf	Er sieht seinen neuen Kontostand	✓

4	Der Kunde geht zurück auf die Simulationsseite und drückt auf "Start"	Die Simulation startet	✓
Nachbedingung(en)			
Testurteil	Test bestanden		

Zyklus III

Spezifikationsplanung

ID	Artefakt	Art des Artefakts	Verantwortlicher	Status
1.	Chat			
1.1	Chat	User Story	Zheyuan	Fertig
1.1.1	Chat Fenster	Papierprototyp	Finn	Fertig
1.2	Chatnachrichten bearbeiten/löschen	User Story	Simon	Fertig
2.	Zwischenstopps bei Fahrterstellung			
2.1	Zwischenstopp hinzufügen	User Story	Dennis	Fertig
2.1.1	Zwischenstopp hinzufügen Ansicht	Papierprototyp	Finn	Fertig
3.	Grafische Statistiken			
3.1	Grafische Statistiken Data	User Story	Zheyuan	Fertig
3.1.1	Grafische Statistiken Fenster (monatlich)	Papierprototyp	Zheyuan	Fertig
3.1.2	Grafische Statistiken Fenster (täglich)	Papierprototyp	Zheyuan	Fertig
3.2	Grafische Statistiken monatlich täglich	User Story	Simon	Fertig
4.	Live-Änderungen während der Fahrt			
4.1	Live-Änderungen	User Story	Dennis	Fertig
4.1.1	Live-Änderungen Ansicht	Papierprototyp	Finn	Fertig
5.	Fahrer-Leaderboard			
5.1	Fahrer-Leaderboard	User Story	Simon	Fertig
5.1.1	Fahrer-Leaderboard Ansicht	Papierprototyp	Simon	Fertig
6.	Architektur			
6.1	Klassendiagramm	Diagramm	Tim	Fertig
6.2	Komponentendiagramm	Diagramm	Björn	Fertig
6.3	Kommunikationsdiagramm	Diagramm	Mohamad	Fertig

User-Stories

User Story-ID	1.1
User Story-Beschreibung	Als Nutzer möchte ich nach der Angebotsannahme einer Fahrt über eine Chat-Oberfläche mit dem potenziellen Fahrer bzw. Kunden kommunizieren können, sodass ich in Echtzeit Nachrichten senden und empfangen kann.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Zheyuan
Abhängigkeiten zu anderen User Stories	1.2

User Story-ID	1.2
User Story-Beschreibung	Als Nutzer möchte ich Chatnachrichten bearbeiten oder löschen können, solange sie vom Empfänger noch nicht gelesen wurden, um Missverständnisse zu vermeiden und um wichtige Informationen rechtzeitig auszutauschen.
Geschätzter Realisierungsaufwand	1 Tag
Priorität	mittel
Autor	Simon
Abhängigkeiten zu anderen User Stories	1.1

User Story-ID	2.1
User Story-Beschreibung	Als Kunde möchte ich beim Erstellen einer Fahranfrage Zwischenstopps zwischen Start und Ziel hinzufügen können, um meine Fahrt flexibel nach meinen Bedürfnissen zu planen
Geschätzter Realisierungsaufwand	3 Tage
Priorität	Hoch
Autor	Dennis
Abhängigkeiten zu anderen User Stories	keine

User Story-ID	3.1
User Story-Beschreibung	Als Fahrer möchte ich meine bisherigen Leistungen anhand von Diagrammen analysieren können, damit ich meine Einnahmen, Fahrzeiten, gefahrene Distanzen und durchschnittliche Bewertungen visuell auswerten kann.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Zheyuan
Abhängigkeiten zu anderen User Stories	3.2

User Story-ID	3.2
User Story-Beschreibung	Als Fahrer möchte ich bei den Diagrammen zwischen einer monatlichen und einer täglichen Ansicht wählen können, um entweder das ganze Jahr oder einzelne Monate im Detail zu betrachten.
Geschätzter Realisierungsaufwand	2 Tage
Priorität	mittel
Autor	Zheyuan
Abhängigkeiten zu anderen User Stories	3.1

User Story-ID	4.1
User Story-Beschreibung	Als Nutzer möchte ich während der Fahrt Zwischenstopps hinzufügen, um während der Fahrt flexibel zu bleiben und sie während der Durchführung anzupassen
Geschätzter Realisierungsaufwand	5 Tage
Priorität	Mittel
Autor	Dennis Lüdiger
Abhängigkeiten zu anderen User Stories	2.1

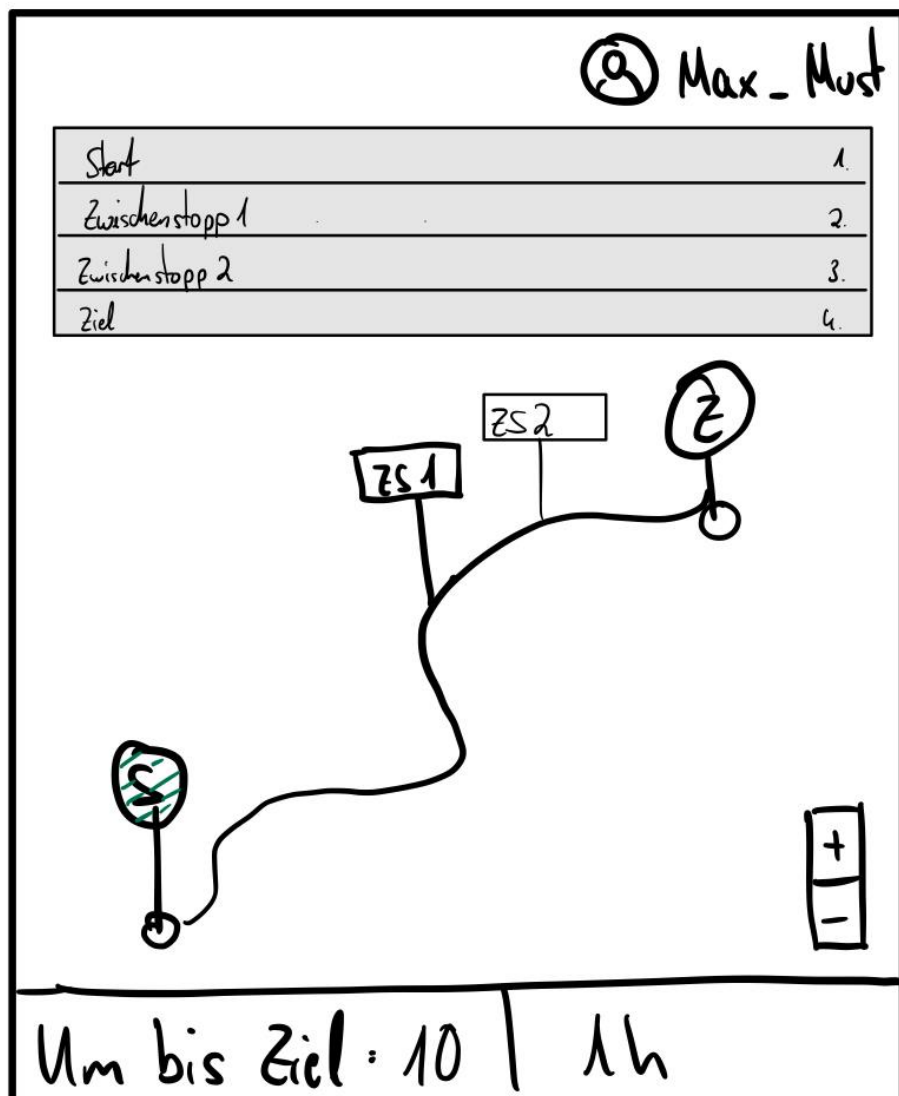
User Story-ID	5.1
User Story-Beschreibung	Als Nutzer möchte ein Fahrer-Leaderboard mit allen relevanten Informationen angezeigt bekommen, dass ich beliebig sortieren und filtern kann, um immer zu wissen, wer die besten Fahrer sind.
Geschätzter Realisierungsaufwand	3 Tage
Priorität	hoch
Autor	Simon Lobo
Abhängigkeiten zu anderen User Stories	keine

Papierprototypen

1.1.1 Chat Fahrer-Kunde

FahrerOl	
<p>Hi, ich brauche 10 min länger!</p> <p>passt das?</p>	
ja, das passt	<div><div></div><div></div></div> <div>nicht gehen</div>
I	
Tasterbus	

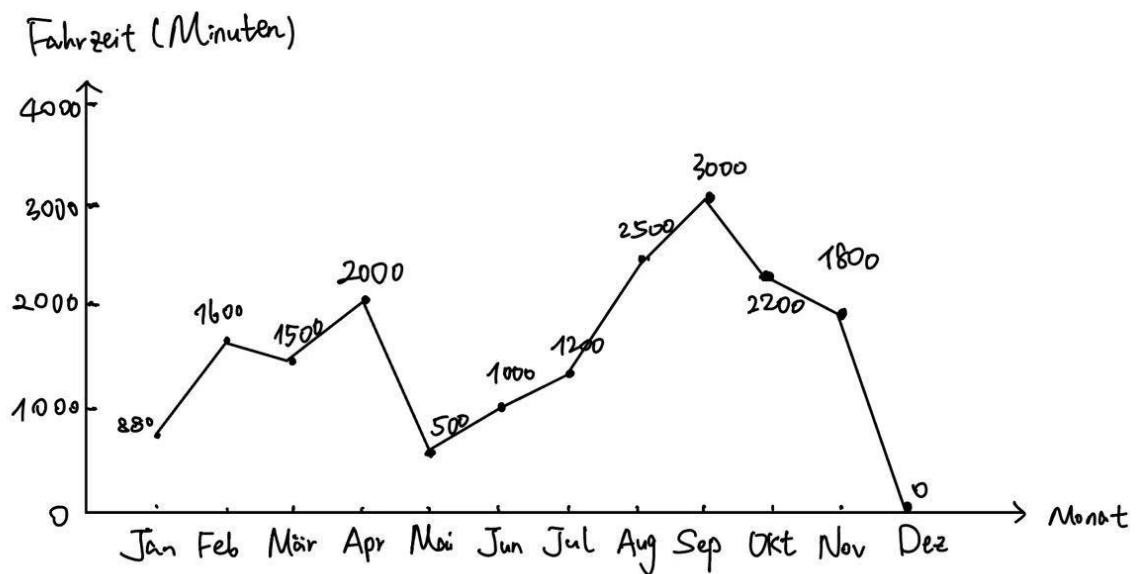
2.1.1 Zwischenstopps Fahrtplanung



3.1.1 Liniendiagramme für Fahrzeit monatlich:

nur bei "Ansicht = Täglich" aktiv
↓
Ansicht: [Monatlich ▼] Jahr: [2025 ▼] Monat: [Dez ▼] Diagrammtyp: [Einnahmen ▼]
Täglich 2024 Nov Distanz
2022 Okt Fahrzeit
Feb Bewertung
Jan

Ansicht: [Monatlich]
Jahr: [2025]
Diagrammtyp: [Fahrzeit]



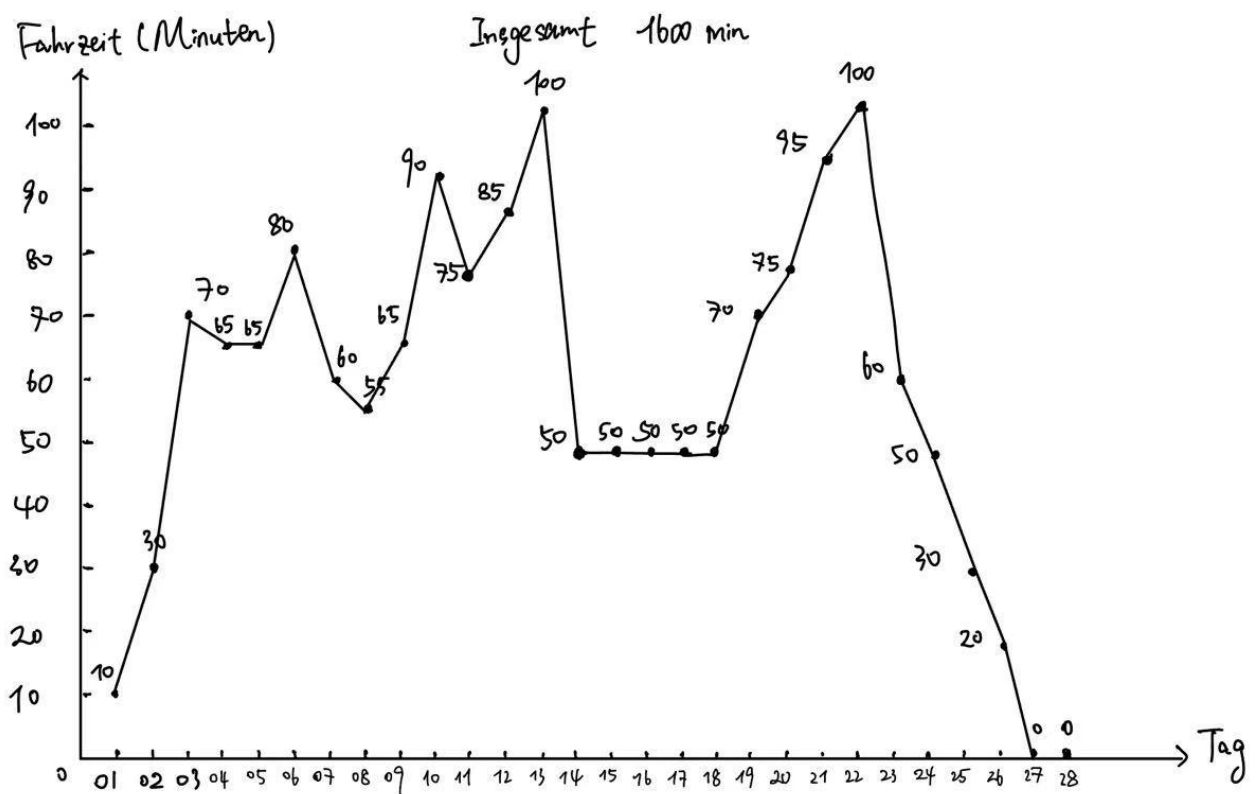
3.1.2 Liniendiagramme für Fahrzeit täglich:

Ansicht: [Täglich]

Jahr: [2025]

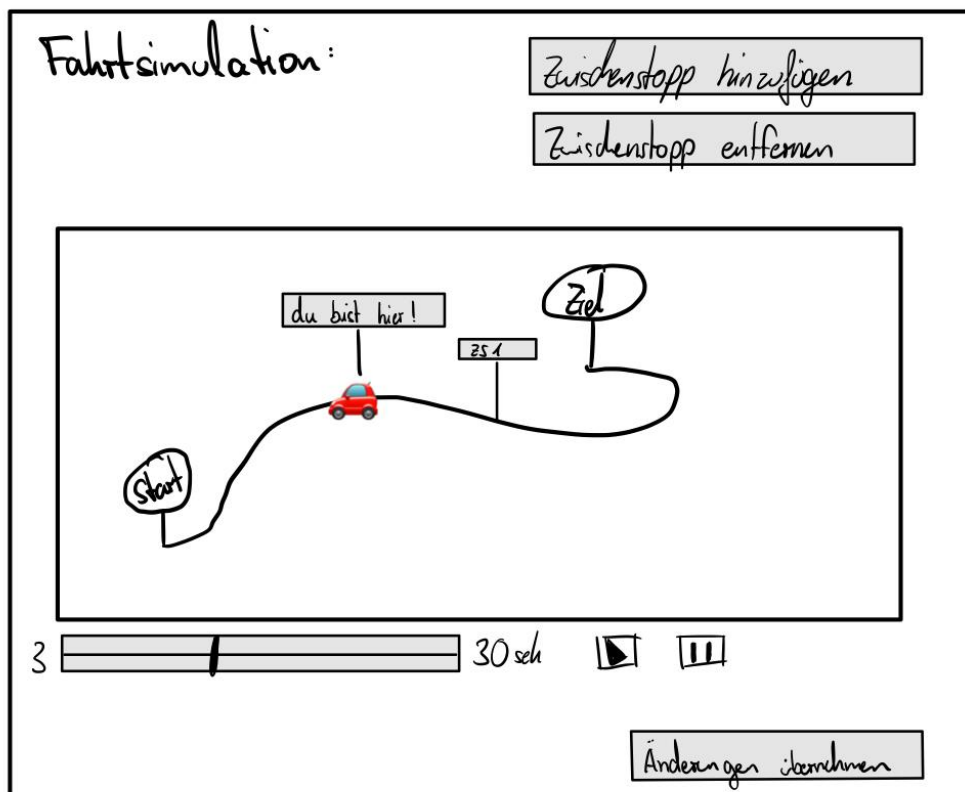
Monat: [Feb]

Diagrammtyp: [Fahrzeit]



Für Einnahmen, gefahrene Distanz und Bewertung muss nach dem korrekten Abrufen der Daten nur der Diagrammtitel sowie die Y-Achse (Werte und Beschriftung) angepasst werden.

4.1.1 Zwischenstopps Simulation



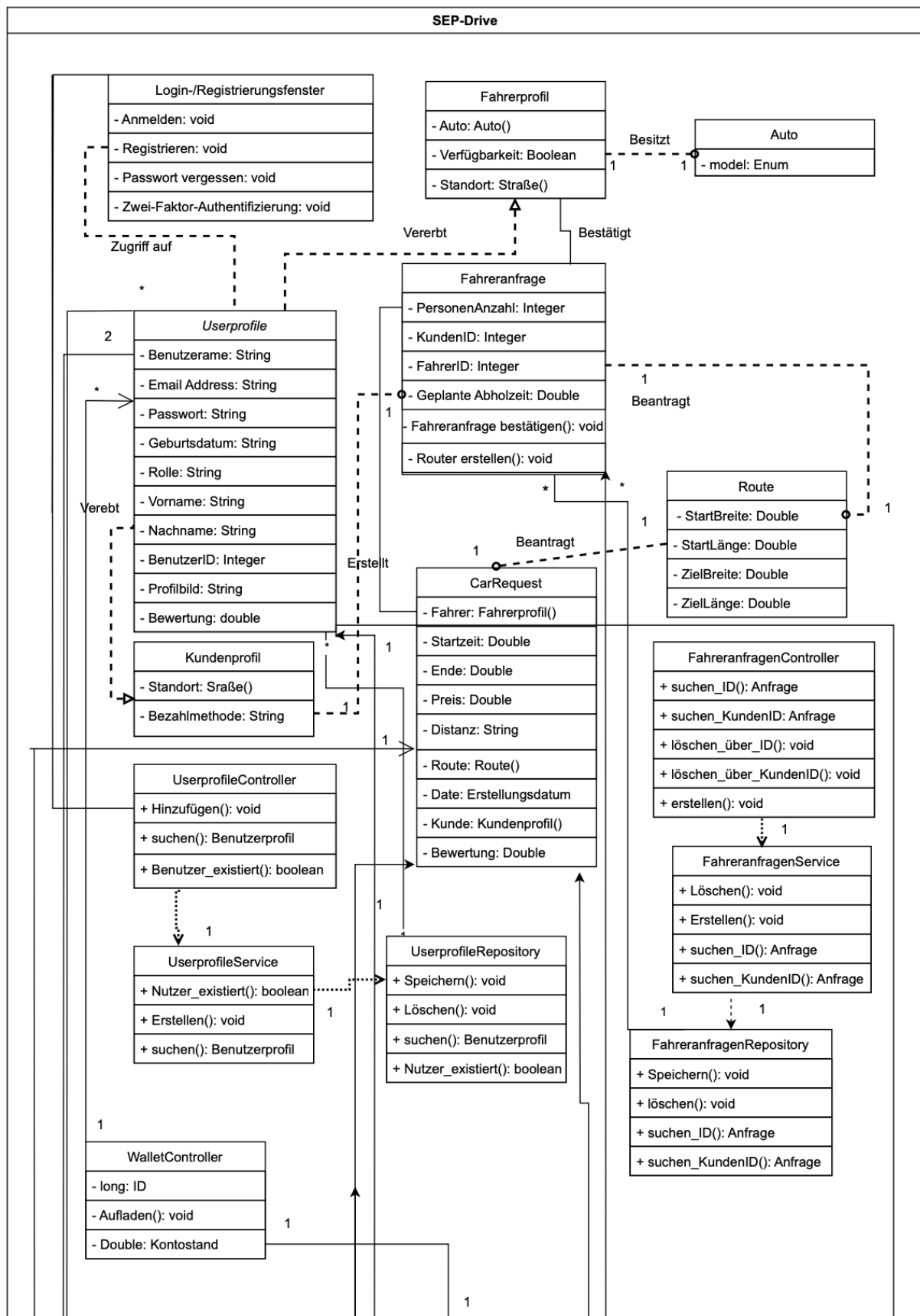
>

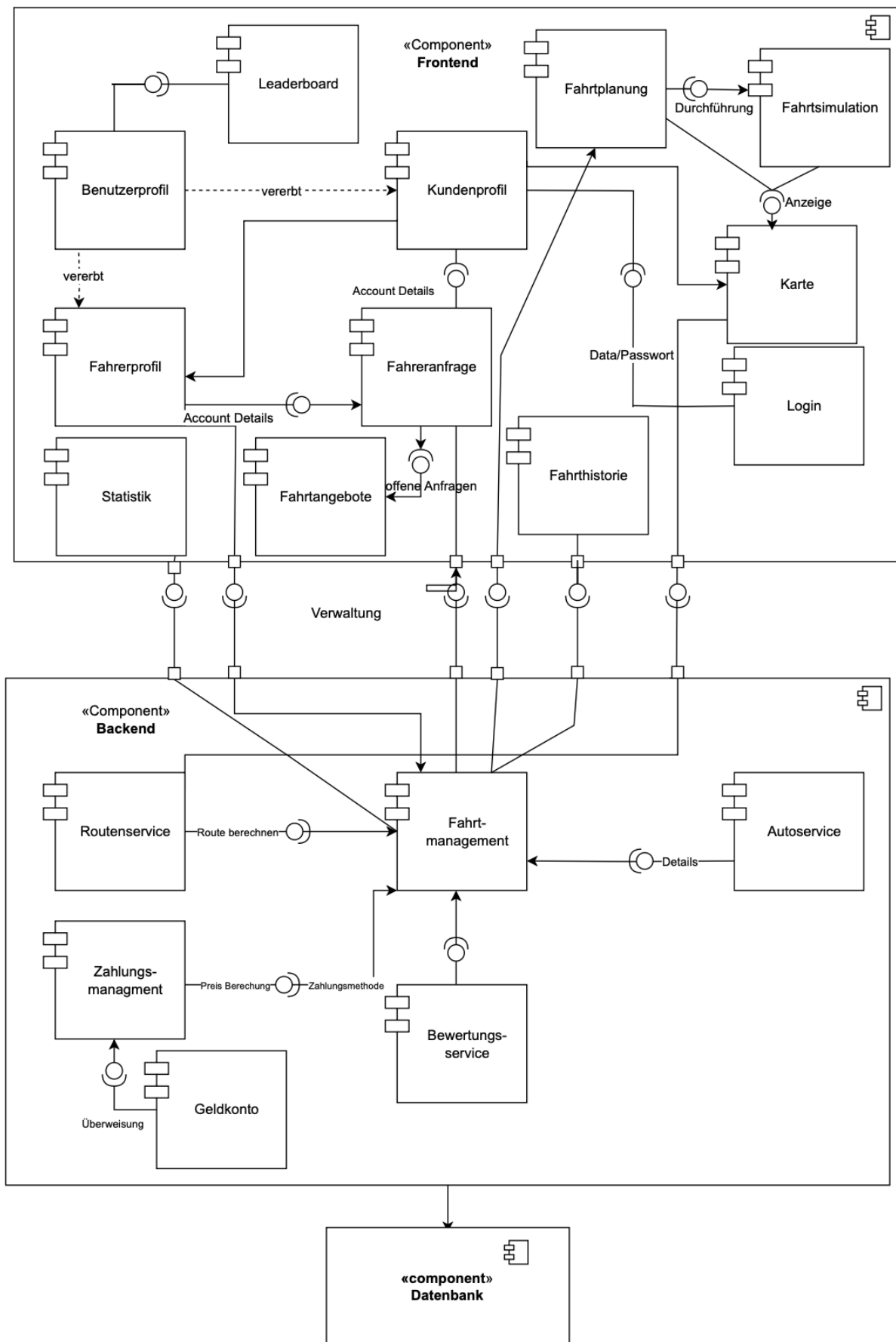
5.1.1 Fahrer-Leaderboard

Fahrer-Leaderboard

Benutzername ↕	Name ↕	ges. Distanz ↕	Bewertung ↕	ges. Fahrtzeit	Fahrten ↕	verdientes Geld ↕
hansi	Hans Speck	899 km	4.3 Sterne	512 min	3	1240 €
fahro	Fahro Gordo	213 km	2.5	103 min	2	345 €

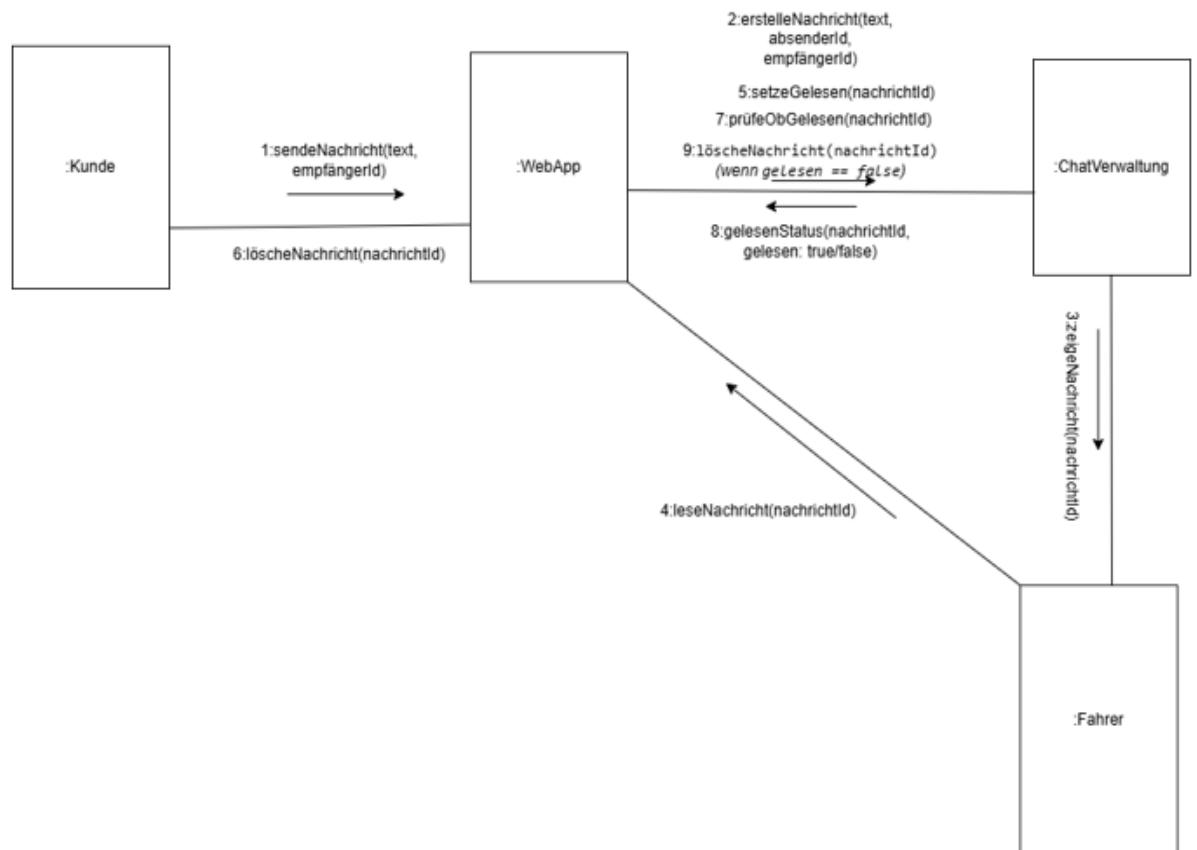
Strukturdiagramme (Komponenten- und Klassendiagramme)



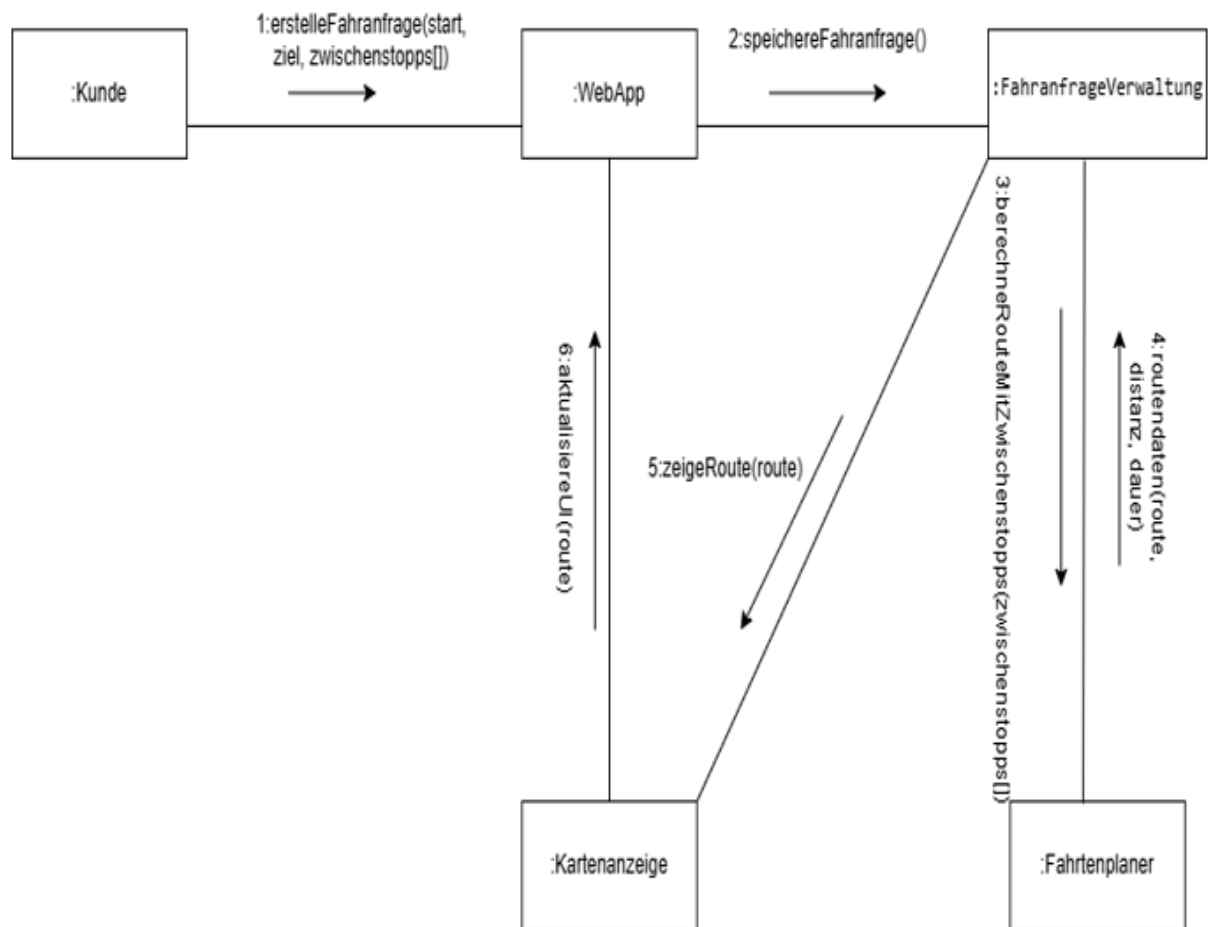


Verhaltensdiagramme (Kommunikationsdiagramme)

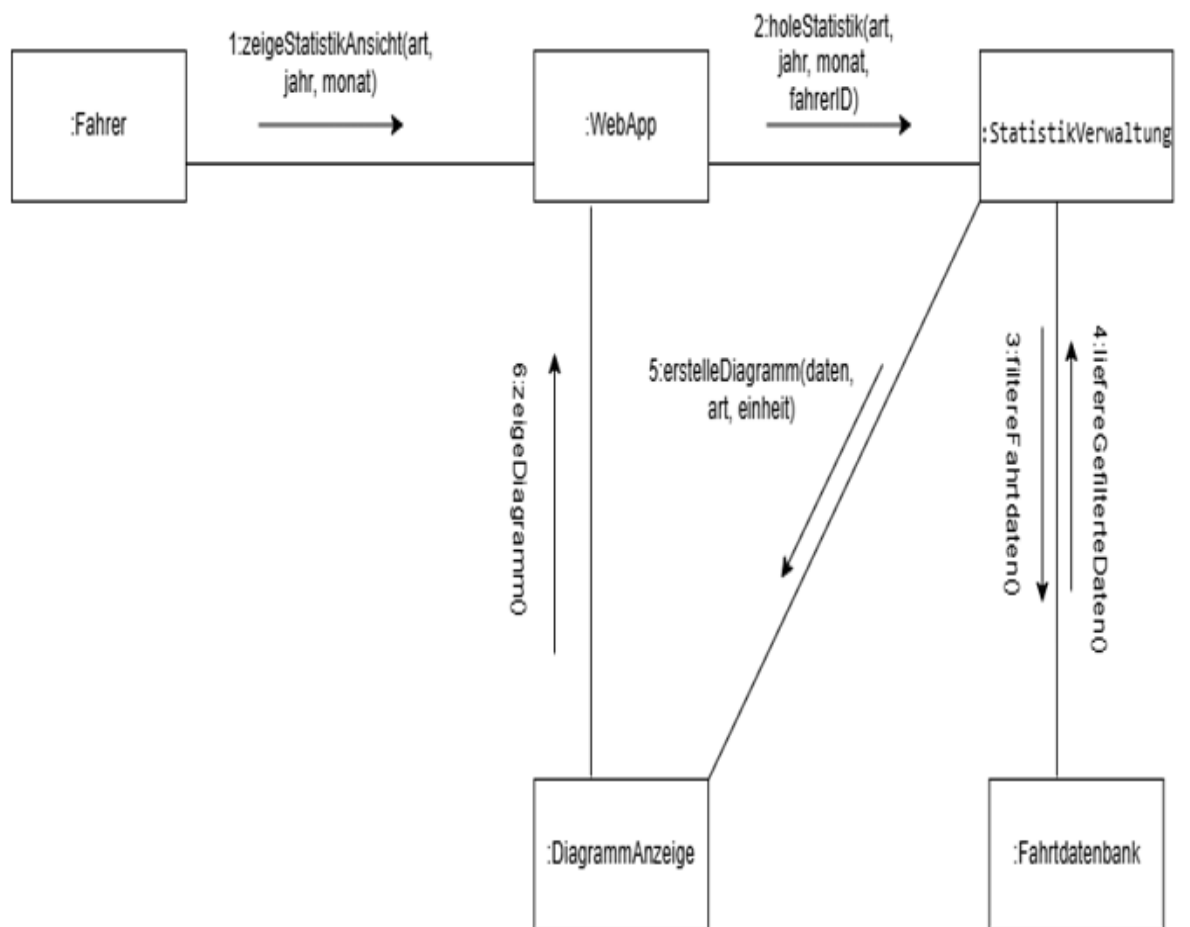
Chat:



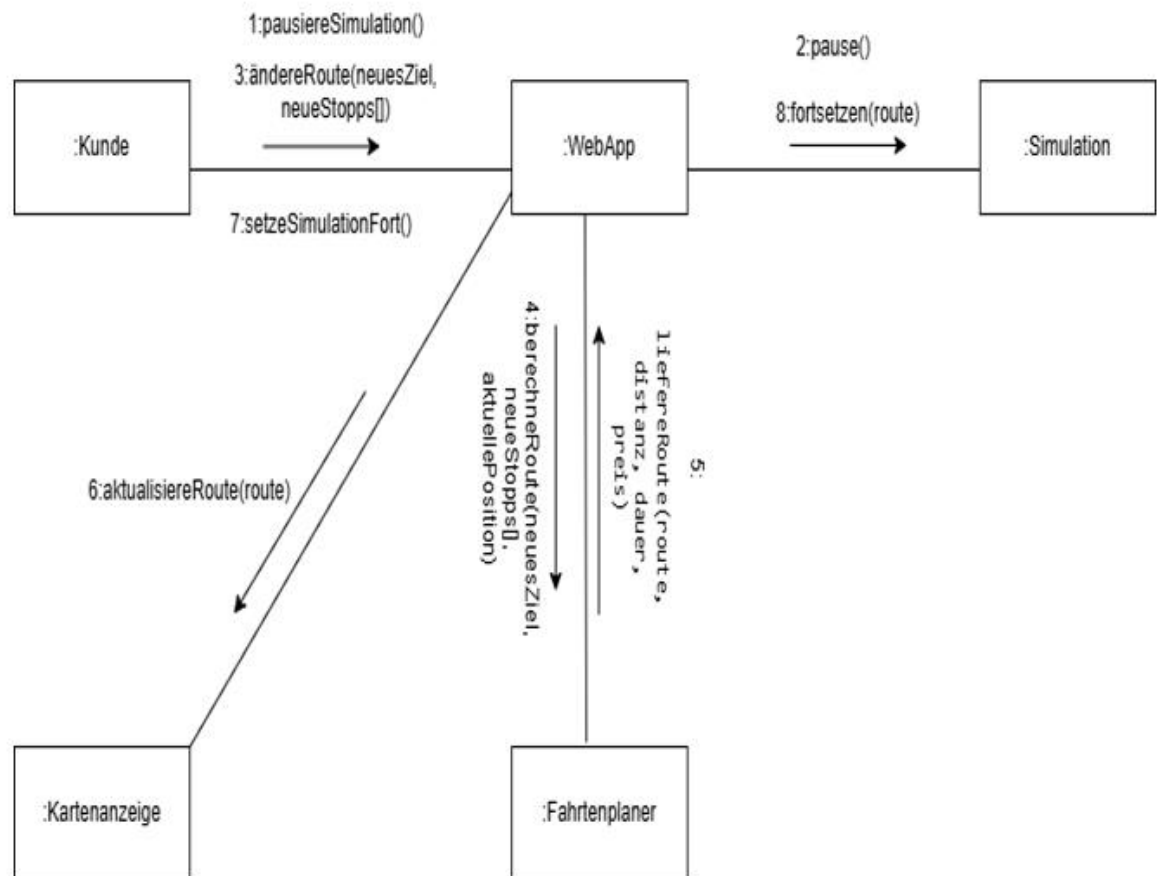
Fahrten mit mehreren Zwischenstopps:



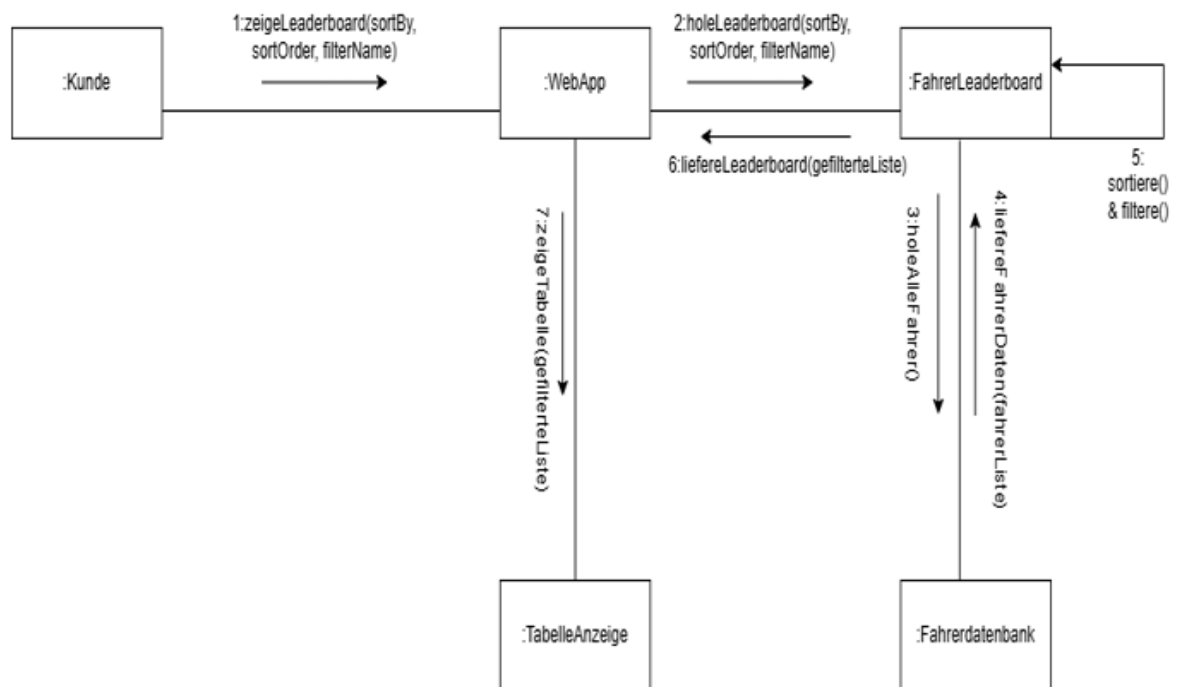
Grafische Statistiken:



Liveänderungen während der Fahrt:



Fahrer Leaderboard:



Funktionalitätsplanung

ID	Funktionalität	Verantwortlicher	Abhängige Funktionalitäten	Verknüpfte User-Stories	Quellcode-referenz	Status
1.	Chat					
1.1	Chat Frontend	Simon	1.2	1.1	Frontend/src/app/chat/chat.component.ts Frontend/src/app/chat/chat.component.html	Fertig
1.2	Chat Backend	Tim	1.1	1.1	Backend/src/main/java/com.sep.drive/Chat	Fertig
2.	Zwischenstopps bei Fahrterstellung					
2.1	Zwischenstopps hinzufügen Frontend	Finn	2.2	2.1	Frontend/src/app/create-ride-request.component Frontend/src/app/waypoint-input (ergänzungen in) Frontend/src/app/map-visualisation	Fertig
2.2	Zwischenstopps hinzufügen Backend	Björn	2.1	2.2	Backend/src/main/java/com.sep.drive/CarRequestController(Z.27-38), Coordinate	Fertig
3.	Grafische Statistiken					
3.1	Grafische Statistiken Frontend	Zheyuan	3.2	3.1	Frontend/src/app/pages/trip-statistics	Fertig
3.2	Grafische Statistiken Backend	Mohamad	3.1	3.1	Backend/src/main/java/com.sep.drive/controller/DriverStatsController	Fertig

					Backend/src/main/java/com.sep.drive/dto/DriverStatResponse Backend/src/main/java/com.sep.drive/service/DriverStatsService	
4.	Live-Änderungen während Fahrt					
4.1	Zwischenstopps bearbeiten (Simulation angepasst) Frontend + kleiner Teil Backend	Dennis	4.2, 2.2	3.1	Frontend/src/app/simulation + Backend/drive/src/main/java/com.sep.drive/service/SimulationService	Fertig
4.2	Zwischenstopps bearbeiten Backend	Björn	4.1, 2.2	3.1	Backend/drive/src/main/java/com.sep.drive/CarRequest/CarRequestController(Z.128-156), CoordinateChangeDTO	Fertig
5.	Fahrer-Leaderboard					
5.1	Leadboard Frontend	Simon	5.2	5.1	Frontend/src/app/leaderboard	Fertig
5.2	Leaderboard Backend	Tim	5.1	5.1	Backend/drive/src/main/java/com.sep.drive/UserProfileRepository(Z.33-37) &UserController (Z.68-71) &UserProfileService (Z.66-71) & UserDTO	Fertig
6.	Frontend Services					
6.1	Chat/WebSocket	Finn	1.1, 1.2	1.1	(Methoden für	Fertig

	t Service				den chat) Frontend/src/ap p/service/webs ocketservice (Z: 22,23 ZZ: 36-39 ZZ: 72-89)	
6.2	Leaderboard Service	Simon	5.1, 5.2	5.1	Frontend/src/ap p/service/leader board.service.ts	Fertig
6.3	Websocket service	Dennis	4.2,2.2	3.1	Frontend/src/ap p/service/webs ocket.service.ts (Zeile 1-20, 25- 29, 54-70, 91- 170)	Fertig
7.	Sonstiges					
7.1	Exceptions hinzugefügt	Björn			Backend/src/ma in/java/com.sep .drive/CarReque st/CarRequestC ontroller + Backend/src/ma in/java/com.sep .drive/UserProfil e/UserControlle r	Fertig

Modultests

ID	Getestete Funktionalität	Quellcode Referenz	Status
1	findAllSort(String sortBy, String order, String filter) – sortiert, filtert und behandelt leere Ergebnislisten korrekt	Sepm/Backend/drive/src/main/test/com/sep/drive/userprofile/UserProfileServiceTest.java	Fertig

Systemtests

Datum	15.07.25		
Tester	Chantal Jedrschyk		
SW-Version	3.0		
Vorbedingung(en)	Es sind zwei Nutzer registriert, eingeloggt und befinden sich auf der Chat-Seite.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Nutzer 1 gibt eine Nachricht in das Textfeld ein und schickt diese über "Senden" ab	Die Nachricht erscheint im Chatverlauf für beide Nutzer (ggf. Runterscrollen nötig)	✓
2	Nutzer 2 schließt die Chat-Seite und Nutzer 1 gibt eine Nachricht in das Textfeld ein und schickt diese über "Senden" ab.	Die Nachricht erscheint für Nutzer 1 im Chatverlauf mit einem Stift- und Mülleimersymbol	✓
2a (Alternative)	Nutzer 1 klickt auf das Mülleimersymbol und bestätigt die Löschung	Die Nachricht wird im Chatverlauf als [Nachricht gelöscht] angezeigt	✓
2b (Alternative)	Nutzer 1 klickt auf das Stiftsymbol	Die Nachricht lässt sich bearbeiten und die Buttons "Speichern" und "Abbrechen" erscheinen	✓
2b1 (Alternative)	Nutzer 1 klickt auf "Abbrechen"	Der Bearbeitungsmodus wird verlassen und die Nachricht wurde nicht verändert	✓

2b2 (Alternative)	Nutzer 1 verändert die Nachricht und bestätigt mit "Speichern"	Der Bearbeitungsmodus wird verlassen und die Nachricht wurde verändert	✓
3	Nutzer 2 kehrt zum Chat mit Nutzer 1 zurück	Für Nutzer 1 verschwinden die Symbole zum Bearbeiten/Löschen. Nutzer 2 sieht eine Nachricht als [Nachricht gelöscht] (Alternative 2a) und eine Nachricht als (bearbeitet) markiert (Alternative 2b2)	✓
Nachbedingung(en)			
Testurteil	Test bestanden		

Datum	15.07.25		
Tester	Chantal Jedrschyk		
SW-Version	3.0		
Vorbedingung(en)	Der Nutzer ist registriert, eingeloggt und befindet sich auf der Leaderboard-Seite. Es sind mindestens 2 Fahrer im System registriert (einer davon mit dem Benutzernamen "hansi")		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Der Nutzer klickt den Kopf einer beliebigen Spalte der Tabelle	Die Tabelle wird nach der entsprechenden Spalte aufsteigend sortiert	✓
2	Der Nutzer klickt erneut den Kopf der Spalte	Die Tabelle wird nach der entsprechenden Spalte nun absteigend sortiert	✓
3	Der Nutzer klickt in die Suchleiste zur Fahrersuche und gibt den Benutzernamen "hansi" ein	In der Suchleiste steht "hansi". Die Tabelle aktualisiert sich automatisch und besteht nur aus einer Zeile mit den Daten zu dem Fahrer "hansi"	✓
4	Der Nutzer gibt in der Suchleiste "x" ein	In der Suchleiste steht "x". Die Tabelle aktualisiert sich automatisch, enthält aber keine Einträge	✓
5	Der Nutzer klickt auf "zurücksetzen"	Die Tabelle aktualisiert sich und alle Fahrer erscheinen wieder	✓

Nachbedingung(en)		
Testurteil	Test bestanden	

Datum	15.07.25		
Tester	Chantal Jedrschyk		
SW-Version	3.0		
Vorbedingung(en)	Der Nutzer ist als Fahrer registriert, eingeloggt und befindet sich auf der Statistiken-Seite. Er hat bereits mindestens eine Fahrt absolviert.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	v / x
1	Der Nutzer hat als Ansicht "monatlich" und als Diagrammtyp "Liniendiagramme" ausgewählt und anschließend auf den Button "suchen" geklickt	Vier Liniendiagramme, die die Einnahmen, Distanz, Fahrzeit und Bewertung des Nutzers für die Monate Januar bis Dezember 2025 darstellen, werden geladen und angezeigt.	v
2	Der Nutzer hat die Ansicht auf "täglich" geändert, den Monat Juli ausgewählt und den Diagrammtyp auf "Balkendiagramme" umgestellt, und anschließend auf den Button "suchen" geklickt	Vier Balkendiagramme, die die Einnahmen, Distanz, Fahrzeit und Bewertung des Nutzers vom 1. bis zum 31. Juli 2025 darstellen, werden geladen und angezeigt.	v
3	Der Benutzer klickt direkt auf den Button "zurücksetzen"	Die vier Diagramme werden auf den Standardzustand zurückgesetzt.	v
Nachbedingung(en)			
Testurteil	Test bestanden		

Datum	15.07.25		
Tester	Chantal Jedrschyk		
SW-Version	3.0		
Vorbedingung(en)	Der Kunde hat eine Fahranfrage erstellt und ein Fahrer hat diese angenommen. Beide Nutzer befinden sich in der Simulation und diese wurde bereits durch einen der beiden Nutzer gestartet.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / ✗
1	Kunde drückt auf Pause	Es erscheinen Knöpfe, um Zwischenstopps hinzuzufügen, das Ziel zu ändern und um Zwischenstopps zu löschen (sofern welche vorhanden sind). Beim Fahrer erscheinen diese Felder nicht	✓
2	Kunde drückt auf den Knopf "Zwischenstopp hinzufügen"	Es erscheint ein Eingabefeld, in dem eine Adresse eingegeben werden kann und es erscheint ein neuer Knopf mit der Beschriftung "Neue Zwischenstopps übernehmen"	✓
3	Kunde drückt auf "Neue Zwischenstopps übernehmen" ohne Koordinaten eingegeben zu haben	Es erscheint eine Warnung mit dem Text "Bitte geben sie gültige Koordinaten ein". Die Route ändert sich nicht	✓
4	Kunde gibt eine gültige Adresse ein, drückt auf Suchen und wählt die Adresse aus	Die Koordinaten dieser Adresse werden nun angezeigt	✓
5	Kunde drückt auf den Knopf "Neue Zwischenstopps übernehmen"	Die Route wird neu berechnet und der neue Zwischenstopp wird nun angezeigt. Der bisherige Fortschritt wird korrekt angezeigt. Tatsächlicher Preis und Distanz ändern sich gegebenenfalls.	✓
6	Kunde drückt auf "continue"	Die Simulation wird an der Stelle, an der pausiert wurde, fortgesetzt	✓
7	Kunde drückt erneut Pause, bevor der nächste Zwischenstopp erreicht wurde	Die Simulation hält an	✓

8	Kunde scrollt runter und drückt unter "Ihre Zwischenstopps" auf den Entfernen Button des nächsten Zwischenstopps	Die Route wird neugeladen und der bisherige Zwischenstopp wird entfernt. Der bisherige Fortschritt bleibt erhalten. Tatsächlicher Preis und Distanz ändern sich gegebenenfalls.	✓
9	Kunde drückt auf "continue"	Die Simulation wird an der Stelle, an der pausiert wurde, fortgesetzt	✓
10	Kunde drückt auf "pause"	Die Simulation hält an	✓
11	Kunde gibt unter "Ziel ändern" neue Koordinaten bzw. Eine neue Adresse ein und drückt auf "Ziel aktualisieren"	Die Route wird neu berechnet und führt nun zum neuen Ziel. Der bisherige Fortschritt bleibt erhalten und wird korrekt angezeigt. Tatsächlicher Preis und Distanz ändern sich.	✓
12	Kunde drückt auf "continue"	Simulation wird fortgesetzt und der Marker folgt nun der neuen Route bis zum Ziel	✓
Nachbedingung(en)	Der Marker ist am Ziel angekommen. Bei Kunde und Fahrer öffnet sich nach kurzer Zeit ein Bewertungsfenster, in dem Kunde und Fahrer die Möglichkeit haben sich gegenseitig zu bewerten. Unter "Fahrthistorie" auf dem Profil des Nutzers wird die Fahrt nun korrekt mit dem tatsächlichen Preis, der tatsächlichen Distanz, usw. angezeigt		✓
Testurteil	Test bestanden		

Datum	15.07.25		
Tester	Chantal Jedrschyk		
SW-Version	3.0		
Vorbedingung(en)	Der Kunde ist als Nutzer registriert und eingeloggt und befindet sich auf der Seite "Fahranfrage erstellen"		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde wählt start-Koordinaten durch das drop down Menu auf	Koordinaten werden übernommen und angezeigt.	✓

	verschiedene Arten aus.		
2	Kunde drückt auf "Wegpunkt hinzufügen" / auf das Lösch Symbol, um beliebig viele Wegpunkte der Route hinzuzufügen oder zu entfernen.	Route wird um Zwischenstopps erweitert	✓
2a	Kunde wählt Koordinaten für die Wegpunkte aus, ebenfalls durch das dropdown Menu wie beim Start.	Koordinaten werden bei den entsprechenden Wegpunkten angezeigt	✓
3	Kunde wählt ziel-Koordinaten durch das drop down Menu auf verschiedene Arten aus.	Koordinaten werden übernommen und angezeigt.	✓
4a	Kunde klickt auf "Route planen"	Der Kunde bekommt die finale route mit Preis, Karte sowie Dauer angezeigt als Vorschau.	✓
4b	Kunde klickt auf "Fahranfrage erstellen"	Die Fahranfrage wird direkt erstellt und der Kunde wird auf die Seite der aktiven Fahranfrage weitergeleitet und bekommt sie hier nochmal detailliert angezeigt.	✓
Nachbedingung(en)	Die Fahranfrage ist im System gespeichert		✓
Testurteil	Test bestanden		

Nutzerhandbuch

Technische Anforderungen

Einen PC, der Windows 10 (oder höher) als Betriebssystem verwendet, Docker Version 27 (oder höher), mindestens 8 GB RAM, einen Webbrowser (z.B. Google Chrome), Internetverbindung (für Routenberechnung, 2FA-Codes per E-Mail, etc.)

Installationsanleitung

1. Images herunterladen und den Speicherort merken
2. Docker Desktop herunterladen und installieren
3. Docker Account erstellen
4. E-Mail verifizieren
5. Docker Desktop starten
6. Terminal (Eingabeaufforderung) öffnen und zum Speicherort der Images navigieren
7. Folgende Befehle ausführen:
 - a. `docker load -i backend.tar`
 - b. `docker load -i frontend.tar`
 - c. `docker load -i database.tar`
8. Nun die folgenden Befehle ausführen. Falls beim Kopieren das Terminal jede Zeile einzeln ausführt: alle Zeilen einzeln ins Terminal kopieren und nach jedem Kopieren ein Leerzeichen setzen (außer bei Bindestrichen am Ende einer Zeile) oder in die Browser-Adresszeile einfügen und von dort rauskopieren. Falls weitere Probleme beim Kopieren der Befehle auftreten, müssen die Befehle abgetippt werden. Insbesondere kann es durch die Konvertierung in eine PDF-Datei vorkommen, dass Bindestriche am Ende einer Zeile beim Kopieren nicht mitkopiert werden.

Aus diesem Grund empfehlen wir das Kopieren der Befehle aus der „docker-befehlte.txt“ Datei, da dort alle Bindestriche richtig dargestellt werden!

- a. `docker volume create postgres-data`
 - b. `docker network create sep-network`
 - c. `docker run -d --name postgres --network sep-network -e POSTGRES_DB=sepDrive -e POSTGRES_USER=postgres -e POSTGRES_PASSWORD=postgres -p 5432:5432 -v postgres-data:/var/lib/postgresql/data postgres:16`
 - d. `docker run -d --name backendcontainer --network sep-network -e SPRING_DATASOURCE_URL=jdbc:postgresql://postgres:5432/sepDrive -e SPRING_DATASOURCE_USERNAME=postgres -e SPRING_DATASOURCE_PASSWORD=postgres -p 8080:8080 gruppe-m-backend`
 - e. `docker run -d --name frontendcontainer -p 4200:80 --network sep-network gruppe-m-frontend`
9. Die URL: <http://localhost:4200> im Browser aufrufen

Bedienungsanleitung

Nach dem Aufruf von <http://localhost:4200> kommt man nun auf die Login Seite, falls man noch keinen Account hat, klickt man auf "Register Now" und wird zum Registrierungsformular geführt. Dort muss man seine E-Mail, Benutzername, Vor- und Nachname, Geburtsdatum eingeben (optional kann man auch ein Profilbild hochladen indem man auf „Choose Profile Picture“ klickt). Außerdem sucht man sich seine Rolle aus (Fahrer oder Kunde), falls man Fahrer auswählt, muss man zudem noch die Fahrzeugklasse auswählen mit der man Kunden befördern möchte. Nach korrekter Eingabe seiner Daten wird man sowohl beim Registrieren als auch beim Login nach einem Zwei-Faktor-Authentifizierungscode gefragt, den man per E-Mail bekommt (alternativ lässt sich hier auch der Supercode „999999“ eingeben). Nach erfolgreicher Eingabe des 2FA-Codes ist man eingeloggt, bzw. nach der Registrierung muss man sich noch einmal einloggen und wird dann auf die Dashboard Seite geleitet. Von hier aus kommt man über den Button „Profil ansehen“ auf sein Profil, kann über den Button „Fahranfrage erstellen“ (nur für Kunden sichtbar) eine Fahranfrage erstellen und über den Button „aktive Fahranfrage“ (nur für Kunden sichtbar) einsehen ob und welche Fahranfrage zurzeit aktiv ist. Über den Button „Fahrangebote“ (nur für Kunden sichtbar) gelangt man zur Seite der Angebote der Fahrer. Über den Button „Leaderboard“ gelangt man zur Leaderboardseite. Außerdem gibt es ein Feld zur Benutzersuche. Dort wird man nach Eingabe eines existierenden Benutzers auf dessen Profil geleitet (fall es das Profil nicht gibt erscheint ein Hinweis).

Im oberen Bereich der Seite ist zu jedem Zeitpunkt eine Navigationsleiste zu sehen (außer beim Login/Registrieren). Über den Button „Start“ gelangt man zum Dashboard, über den Button „Neue Fahranfrage“ (nur für Kunden sichtbar) hat man die Möglichkeit eine Fahranfrage zu erstellen, über den Button „Fahrt-Simulation“ gelangt man zur Fahrt-Simulation, über den Button „Kunden Fahranfragen“ (nur für Fahrer sichtbar) gelangt man zur Seite zum Einsehen der Fahranfragen der Kunden. Wenn man rechts auf seinen Benutzernamen klickt gelangt man auf das eigene Profil (Unterschied zu fremden Profilen ist, dass man mit „Hallo...“ begrüßt wird).

Auf der Seite zum **Erstellen einer Fahranfrage** muss man ein Start, Ziel, sowie die gewünschte Fahrzeugklasse (klein, medium, deluxe) auswählen (es sind nur Fahrten innerhalb Deutschlands möglich). Beim Start hat man die Möglichkeit den aktuellen Standort zu nehmen (muss vom Browser genehmigt werden), Koordinaten auszuwählen und diese dementsprechend einzugeben, POI (Point of Interest) auszuwählen und dann erneut zwischen Restaurant, Theater, Museum auszuwählen und dann in der Suchleiste Suchbegriffe einzugeben und indem man auf "POI Suchen" klickt, sich Ergebnisse anzeigen zu lassen, von denen man durch Anklicken eins auswählen kann. Außerdem besteht die Möglichkeit als Start eine Adresse anzugeben. Nach dem Klicken von „Adresse suchen“ besteht wie beim POI die Möglichkeit, aus den Suchergebnissen, den gewünschten Start durch Anklicken auszuwählen. Beim Festlegen des Ziels gibt es die gleichen Möglichkeiten (außer der Möglichkeit den aktuellen Standort als Ziel anzugeben) wie bei der Startauswahl. Über den Button "Zwischenstopp hinzufügen" kann man, auf gleiche Weise wie bei der Zielauswahl, mehrere **Zwischenstopps** hinzufügen und diese über den Button "Entfernen" wieder löschen. Vor dem abschicken der Fahranfrage kann man, indem man auf „route planen“ klickt, sich Information zur Route anzeigen lassen. Wurden alle Felder richtig ausgefüllt kann man die Fahranfrage abschicken und wird zu der Seite der aktiven Fahranfrage geleitet (falls es bereits eine aktive Fahranfrage gibt, wird man ebenfalls zu der aktiven Fahranfrage geleitet, die neue Fahranfrage wird aber nicht übernommen). Zudem bekommt man einen Hinweis, wenn die Koordinaten auf welche Weise auch immer ausgewählt, ungültig sind.

Auf der Seite „**Aktive Fahranfrage**“ sieht man, seine aktive Fahranfrage auf einer interaktiven Karte visuell dargestellt und kann diese über den Button „Fahranfrage löschen“ löschen. Falls es zurzeit keine aktive Fahranfrage gibt, wird dies als Text erscheinen. Das Ausloggen ist zurzeit noch nicht

möglich (außer indem man das Token im Browser aus der Local Storage löscht. Man kann jedoch im Inkognito- bzw. Private Tab einen neuen Account anlegen oder sich erneut einloggen).

Falls man auf dem eigenen **Profil** ist, kann man über den grünen Button „**Geldkonto**“, den Kontostand einsehen und Geld aufladen. Über den blauen Button „**Fahrthistorie**“ gelangt man zur Seite aller abgeschlossener Fahrten. In der Suchleiste kann man einen Namen (Fahrer oder Kunde) eingeben und bekommt, nachdem man auf „suchen“ geklickt hat, alle Fahrten angezeigt, bei denen die gesuchte Person involviert war. Über den Button „zurücksetzen“ kann man die Suche zurücksetzen und bekommt wieder alle Fahrten angezeigt. Über den Button „Statistiken“ (auf der Profilseite und nur für Fahrer sichtbar) gelangt der Fahrer auf die Seite, wo er seine **Statistiken** einsehen kann. Hier hat man die Möglichkeit zwischen monatlicher und täglicher Ansicht zu wechseln, den gewünschten Zeitraum auszuwählen und den Diagrammtyp auszuwählen. Durch „Suchen“ übernimmt man seine Änderungen und durch „zurücksetzen“ wechselt man zurück zur Standardansicht.

Auf der Seite „**Kunden Fahranfragen**“ können Fahrer die Anfragen der Kunden in einer Tabelle sehen und jede Spalte auf- bzw. absteigend sortieren, indem sie auf ein beliebiges Element der Kopfzeile klicken. Um die Entfernung zum Start der Fahrt zu sehen, muss der Fahrer zuerst den aktuellen Standort angeben (gleiche Vorgehensweise wie beim Erstellen einer Fahranfrage, nur für Orte innerhalb Deutschlands). In der letzten Spalte „Aktionen“ hat der Fahrer die Möglichkeit sich auf eine Fahrt zu bewerben bzw. sein Angebot zurückzuziehen (es liegt in der Verantwortung des Fahrers nur Anfragen anzunehmen, die auch seiner Fahrzeugklasse entsprechen oder den Kunden zu informieren, da es zu Preisänderungen kommen kann). Nachdem der Fahrer sich für eine Fahrt beworben hat, erscheint in derselben Spalte der **Chat-Button**, der den Fahrer zum Chat mit dem entsprechenden Kunden führt. Um zu verhindern, dass versucht wird sich auf gelöschte Fahranfragen zu bewerben oder bereits angenommene Angebote zurückzuziehen, empfehlen wir das Neu laden der Seite oder das Klicken des „aktualisieren“ Buttons. Falls der Kunde das Angebot angenommen hat, sieht der Fahrer beim Aktualisieren anstatt der Tabelle einen entsprechenden Hinweis.

Wenn der Kunde sich auf der Seite „**Fahrangebote**“ befindet und ein Fahrer sich auf die Fahrt bewirbt, erscheint eine entsprechende Benachrichtigung und der Kunde kann weitere Details der Tabelle entnehmen. Die Tabelle kann, wie die Tabelle der Fahranfragen, entsprechend sortiert werden. Der Kunde kann durch die Buttons „annehmen“ bzw. „ablehnen“ in der Aktionen Spalte aussuchen, ob er das Angebot annimmt oder ablehnt und mithilfe der **Chat-Spalte**, den Chat mit dem entsprechenden Fahrer öffnen.

Nach dem Annehmen können sowohl Kunde als auch Fahrer die Fahrt auf der Seite „**Fahrt-Simulation**“ verfolgen bzw. steuern. Über die entsprechenden Buttons kann die Fahrt gestartet, pausiert sowie fortgesetzt werden. Die Fahrt kann nur gestartet werden, wenn der Kunde ausreichendes Guthaben hat, sonst erscheint eine Meldung. Mithilfe des Reglers kann die Simulationsgeschwindigkeit angepasst werden. Der Kunde hat, nach dem Pausieren der Simulation, die Möglichkeit beliebig **Zwischenstopps** hinzuzufügen bzw. zu entfernen und das Ziel zu ändern. Das Entfernen von Zwischenstopps ist unterhalb der Karte möglich indem auf „Entfernen“ des jeweiligen Zwischenstopps klickt. Das Erstellen neuer Zwischenstopps funktioniert wie beim Erstellen der Fahranfrage, muss aber durch „Zwischenstopps übernehmen“ bestätigt werden. Das Ziel kann auf gleiche Weise aktualisiert werden und muss durch „Ziel aktualisieren“ bestätigt werden. Beim Abschluss der Fahrt erscheint für beide ein Fenster zum gegenseitigen Bewerten (durch Abbrechen wird keine Bewertung vergeben).

Auf der Seite **Leaderboard**, sieht man alle registrierten Fahrer standardmäßig nach der Fahrtenanzahl absteigend sortiert. Durch Klicken des Tabellenkopfes, kann man die Sortierung nach dem

gewünschten Merkmal aufsteigend oder absteigend sortieren. In die Suchleiste kann man, über den Benutzernamen, nach einem bestimmten Fahrer suchen. Falls der eingegebene Benutzername existiert, erscheint der entsprechende Fahrer automatisch. Damit wieder alle Fahrer erscheinen kann man entweder die Suchleiste leeren oder auf den Button "zurücksetzen" klicken.

Auf der **Chatseite** erscheint die Chathistorie, falls es schon einen Chat zwischen den zwei Personen gab (sonst erscheint der Text "Beginne die Unterhaltung"). In der Texteingabe kann man seine Nachricht eingeben und über den Button "Senden" abschicken. Falls die andere Person den Chat nicht offen hat und somit die Nachricht ungelesen ist, besteht die Möglichkeit, durch das Klicken der entsprechenden Symbole die Nachricht zu bearbeiten bzw. zu löschen. Das Bearbeiten einer Nachricht kann mit "speichern" gespeichert und mit "abbrechen" abgebrochen werden. Nachdem man den Mülleimer geklickt hat und bestätigt, hat das man die Nachricht löschen will, wird die entsprechende Nachricht für beide als gelöscht dargestellt. Sobald der Chatpartner die Nachrichte gelesen hat, verschwinden die Symbole zum Löschen und Bearbeiten. Bei vielen Nachrichten ist ggf. Runterscrollen nötig, um die neusten Nachrichten zu sehen.

Bei Fehlern, die die Routenberechnung oder ähnliches betreffen, bitte sicherstellen das die verwendete API OpenStreetMap verfügbar ist.