

Vintage Car Community Projekt



Inhalt

Projekt Überblick

Pflichtenheft

Softwareentwurf

Testkonzept

Demo



Projekt Überblick

Ziel: Eine Plattform, um Besitzer von Old- und Youngtimern mit Anbietern von Stellplätzen zu verknüpfen

Rollen:

Projektmanager	Georg Kleffmann
Requirements Engineer	Julian Duchscherer
Softwarearchitekt	Thaddäus Friedel
Testmanager	Sascha Lang



Projekt Überblick

Vorgehen: 1-2 Treffen die + extra Treffen vor den Abgaben

Tools:

Draw.io: zum Erstellen der Diagramme

Discord: für Treffen die nicht vor Ort waren

Word-online: erstellen von Dokumenten

Jetbrains Code-with-me: gemeinsames Coding

Umsetzung:

TypeScript: Als Programmiersprache

Node.js: Für den Server

SQLite3: Als Datenbank

DataGrip: Zum Managen der Datenbank

HTML: Für die Darstellung



Pflichtenheft

Systemvision:

Das Ziel der Vintage Car Community ist es, Fahrzeughaltern von Young- und Oldtimern das Finden eines Stellplatzes und die Pflege sowie die Wartung ihres Fahrzeuges zu erleichtern. Die App soll Lagerhaltern, Werkstätten und Ersatzteilanbietern eine Plattform geben, um sich mit potenziellen Kunden und untereinander zu verbinden.



Pflichtenheft

Use Case 3: Stellplatzsuche

Abschnitt	Beschreibung
Name	Stellplatz suchen
Primär Akteur	Einlagerer
Weitere Akteure	Lagerhalter
Stakeholder Ziele	Nach einem geeigneten Stellplatz suchen können, der den Anforderungen entspricht IDs: StR.E.1, StR.E.2
Auslösendes Ereignis	Das Bedürfnis einen Stellplatz zu finden welcher den eigenen Anforderungen entspricht.
Beschreibung	Der Einlagerer sucht anhand von gewissen Fahrzeugeigenschaften mit Hilfe des Systems nach einem geeigneten Stellplatz. Das System liefert ihm anhand der Fahrzeugeigenschaften die passenden freien Stellplätze in Verbindung mit dem dazugehörigen Lagerhalter.
Beziehung zu anderen Use Cases	Angebot anfordern



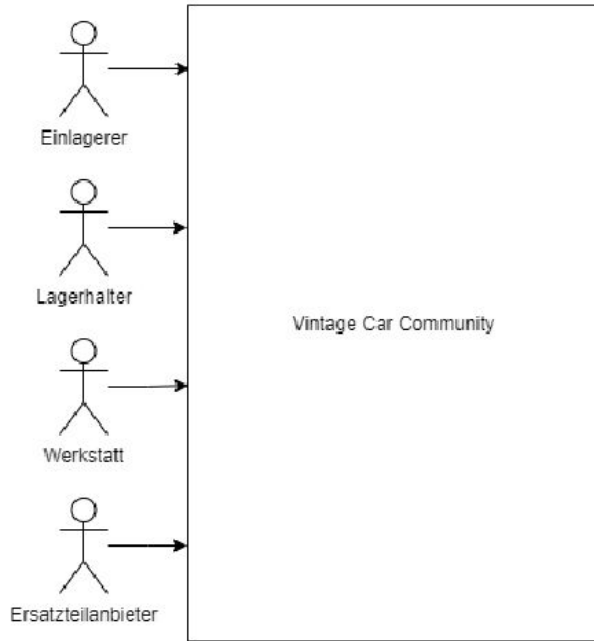
Pflichtenheft

Use Case 3: Stellplatzsuche

Abschnitt	Beschreibung
Vorbedingung	
Nachbedingung	Der Einlagerer hat einen möglichen Lagerplatz gefunden, welcher seinen Ansprüchen entspricht
Hauptszenario	<ol style="list-style-type: none">1. Nutzer startet den Vorgang der Lagerplatzsuche2. System erfragt alle Informationen, welche die Ansprüche des Nutzers an einen Lagerplatz widerspiegeln3. System sucht alle Lagerplätze, welche den Kriterien entsprechen und listet sie auf4. Nutzer fordert ggf. Angebot von möglichen Lagerplätzen an5. Nutzer beendet den Vorgang6. System kehrt zu dem Startbildschirm zurück
Alternativszenario	<p>4a. Der Nutzer findet keinen Lagerplatz, zu dem er ein Angebot anfordern möchte</p> <p>Weiter mit 5. im Hauptszenario</p>
Ausnahmeszenario	<p>3a. Es existieren keine Lagerplätze, welche den Kriterien des Nutzers entsprechen</p> <ol style="list-style-type: none">1. System informiert den Nutzer das keine Lagerplätze gefunden wurden <p>Weiter mit 5. im Hauptszenario</p>
Spezielle Anforderungen	

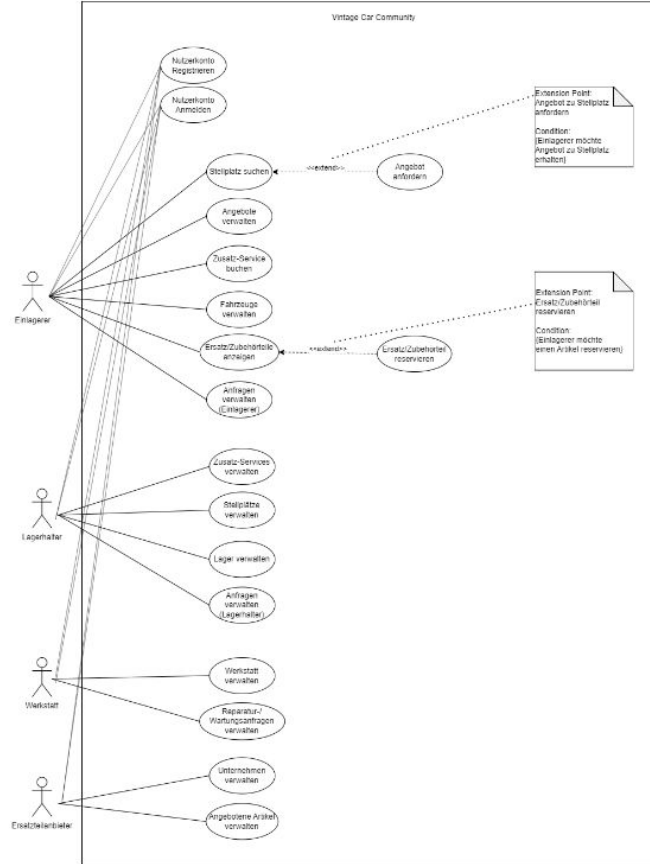
Pflichtenheft

Kontextdiagramm

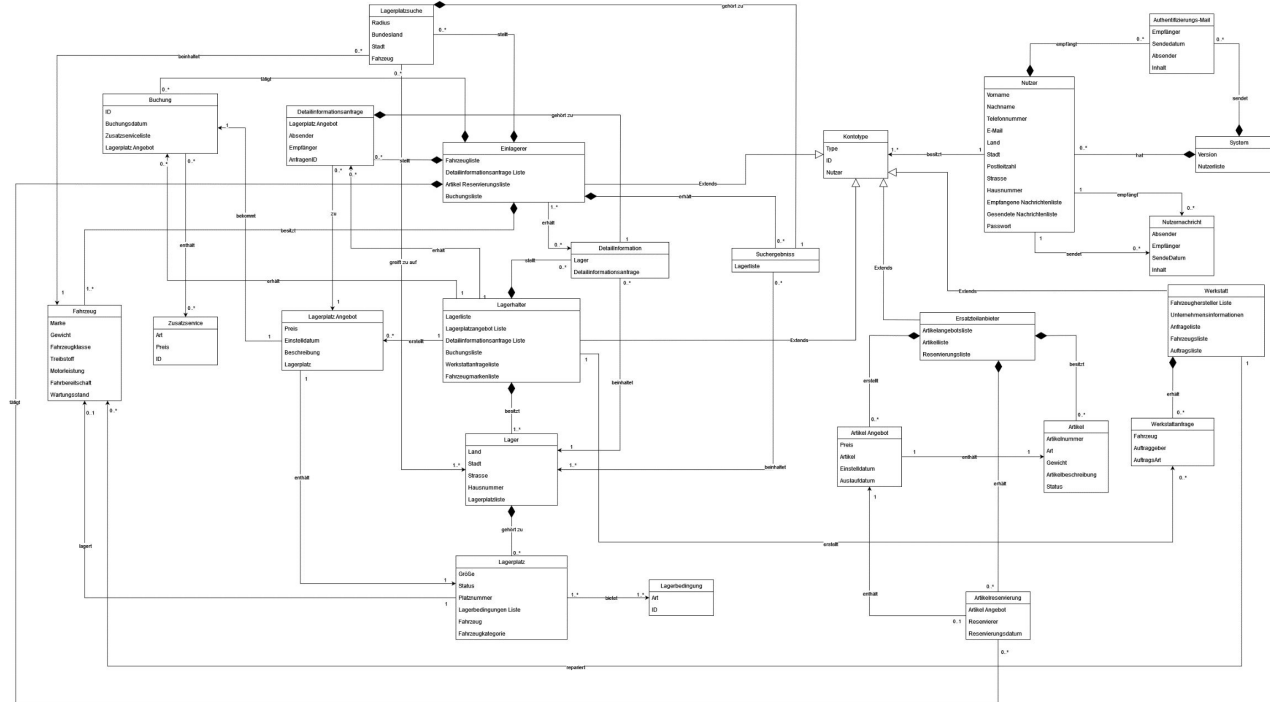


Pflichtenheft

Use Case Modell

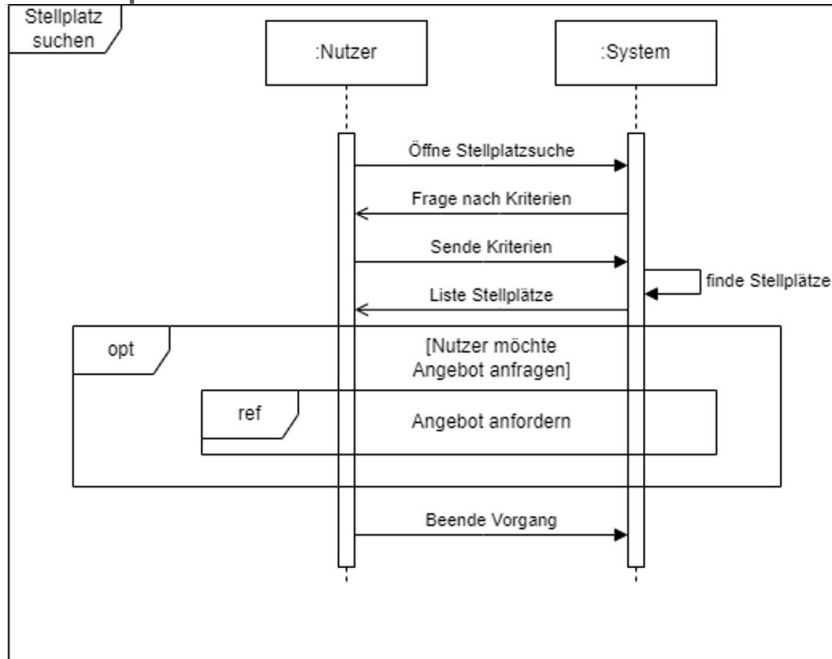


Pflichtenheft



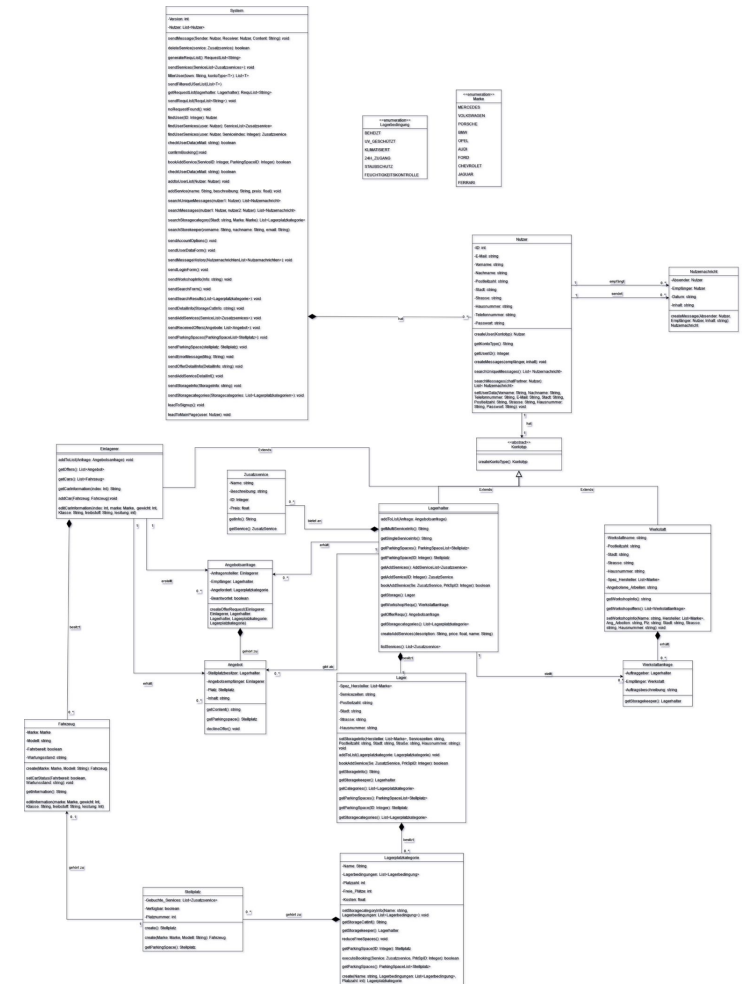
Pflichtenheft

Use Case 3: Stellplatzsuche



Softwareentwurf - Überblick

- Contracts
- Interaktionsdiagramme
- Entwurfsklassendiagramm
- GUI Skizze
- Design Decisions



Contracts UC3

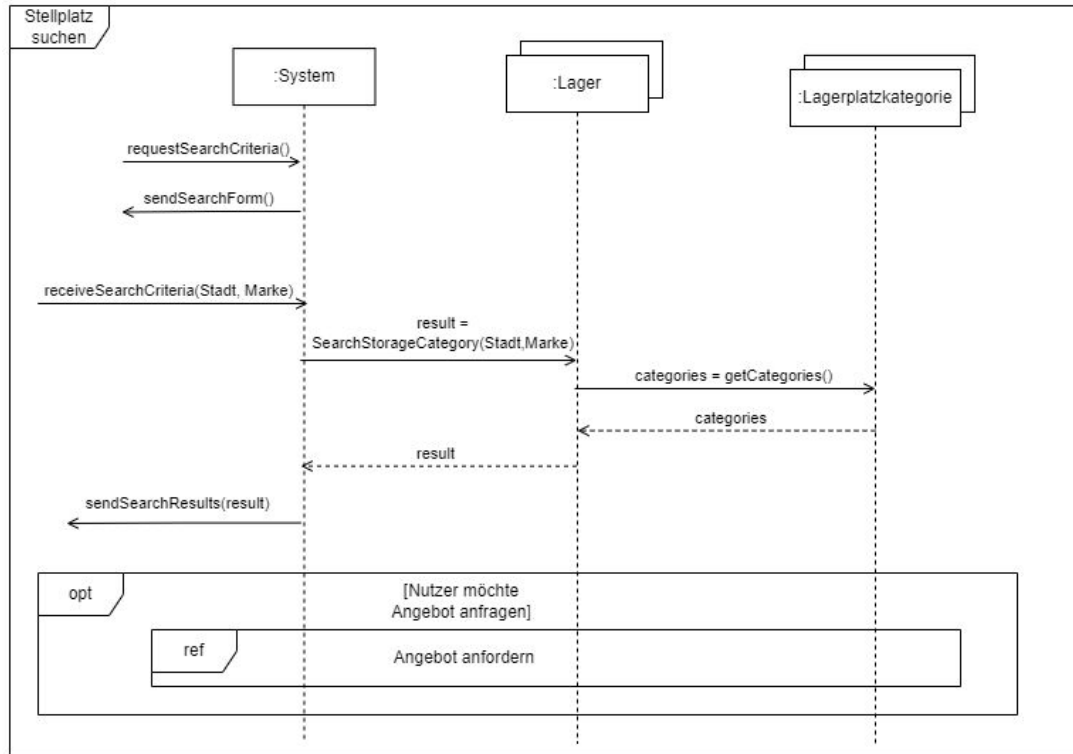
Contract UC3.CO1: requestSearchCriteria

Operation: requestSearchCriteria(void)
Querverweise: Stellplatz suchen
Vorbedingungen: Stellplatzsuche wurde geöffnet
Nachbedingungen: •Lagerplatzsuchformular wurde angezeigt

Contract UC3.CO2: receiveSearchCriteria

Operation: receiveSearchCriteria(strCity: String, strMarke: String)
Querverweise: Stellplatz suchen
Vorbedingungen: Die Suchparameter wurden angezeigt (Contract UC3.CO1 ist erfüllt)
Nachbedingungen: •Suche wurde ausgeführt •Suchergebnis wurde zurückgeliefert und dem Nutzer angezeigt

Interaktionsdiagramm: UC5 Stellplatzsuche



Softwareentwurf - GUI Skizze

Vintage Car Community

Hallo <Vorname>, was möchtest du tun?

Stellplatz suchen

Angebote verwalten

Zusatz-Service buchen

Fahrzeuge verwalten

Anfragen verwalten

 **Stellplatzsuche**

Stadt 

Marke 

Zurück

Suche

 **Stellplatzsuche**
Stellplätze in <Stadt>

Standort: 12345 Musterstadt Musterstraße 1
Lagerbedingungen: Beheizt, Klimatisiert
Lager spezialisiert auf Hersteller: Mercedes

Standort: 12345 Musterstadt Mustergasse 4
Lagerbedingungen: Beheizt

Standort: 12345 Musterstadt Musterstraße 24a
Lagerbedingungen: Beheizt, 24h Zugang
Lager spezialisiert auf Hersteller: Volkswagen, Audi

4. Design-Decisions

1. Use-Cases im Zusammenhang mit dem Kauf/Verkauf von Ersatz-/Zubehörteilen wurden aus dem ersten Release herausgenommen
2. UC1
 - 2.1. Endet bereits nach Weiterleitung zur Anmeldung, da der Rest bereits in anderen Use-Cases beschrieben
 - 2.2. Das Senden einer Authentifikations-Mail wurde aus dem ersten Release herausgenommen
 - 2.3. Die E-Mail soll überprüft werden, damit sie nicht doppelt genutzt wird
 - 2.4. Dem Sequenzdiagramm fehlte die beginnende Interaktion des Users
3. UC3
 - 3.1. Im ersten Release erstmal Suche nur anhand von Parameter Stadt und Marke
 - 3.2. Dem Domänenmodell fehlte die Klasse "Lagerplatzkategorie" diese wurde hinzugefügt
4. UC4
 - 4.1. Detailinformationsanfragen wurden komplett rausgenommen bzw. durch Angebotsanfragen ersetzt, da nicht nötig
5. UC5
 - 5.1. "Buchung"-Klasse war nicht nötig, daher rausgenommen
6. UC6
 - 6.1. Alt-Block nicht nötig, da nur Stellplätze mit Zusatzservice auswählbar
7. UC12
 - 7.1. "Frage nach Stellplatz Informationen" und "Frage nach Kategorien" erwarten die gleiche Antwort, daher einer überflüssig
 - 7.2. Das Senden von Daten, die erst später abgespeichert werden, ist nicht sinnvoll. Daher werden die Daten erst gesendet, wenn sie auch gespeichert werden sollen.
 - 7.3. Es muss sichergestellt werden, dass man keine Stellplätze löschen kann, auf denen ein Fahrzeug steht
8. UC13
 - 8.1. Siehe 7.2
9. UC15
 - 9.1. Siehe 7.2

Testkonzept - Überblick

- Aufbau des Testkonzeptes:
 - Auflistung der Vorbedingungen, Eingaben/Handlungen des Testers, Erwartete Ausgaben/Reaktionen des Systems, Erwartete Nachbedingungen nach dem Test sowie Prüfanweisung die zu befolgen sind
 - Pro Use-Case gibt es ca. 2-3 Testfälle

2.1.2. Testfall UC01-02

Vorbedingungen:

- Es besteht bereits ein Konto mit der gewählten Test-E-Mail
- Tester befindet sich auf der Registrierungsseite

Eingaben/Handlungen:

1. Erstellen eines Kontos mit der E-Mail eines bereits bestehenden Kontos
2. Alle weiteren Daten sind frei wählbar

Erwartete Ausgaben/Reaktionen:

- System zeigt Fehlermeldung
- System leitet erneut auf Registrierungsseite weiter

Erwartete Nachbedingungen:

- Kein Konto wird erstellt
- Alle Daten außer der E-Mail-Adresse bleiben in den Eingabefeldern erhalten
- Erneute Weiterleitung zur Registrierungsseite erfolgt

Prüfanweisungen:

- Eingaben sind über das Registrieren Fenster zu übergeben
- Nach der Registrierung muss geprüft werden,
 - dass das bereits bestehende Konto nicht beeinflusst wurde
 - dass die übernommenen Daten korrekt sind

Testkonzept - Überblick

2.8.1. Testfall UC08-01

Vorbedingungen:

- Tester hat ein Dummy Einlagerer und ein Dummy Lagerhalter Konto erstellt
- Tester hat über das Dummy Einlagerer Konto dem Dummy Lagerhalter Konto eine Anfrage gestellt
- Tester befindet sich auf der Homepage

Eingaben/Handlungen:

1. Tester begibt sich zur Anfragen Verwalten Seite

Erwartete Ausgaben/Reaktionen:

- System sucht gestellte Anfragen des Nutzers
- System listet gestellte Anfragen auf

Erwartete Nachbedingungen:

- Tester sieht Liste der gestellten Anfragen
- Angezeigte Anfragen stimmen mit gestellten Anfragen des Kontos überein

Prüfanweisungen:

- Sicherstellen das die Liste der gestellten Anfragen mit den gestellten Anfragen des Kontos übereinstimmen
- Sicherstellen das die gestellten Anfragen korrekt angezeigt werden

2.6.4. Testfall UC06-03

Vorbedingungen:

- Tester hat bereits mindestens einen Dummy Lagerplatz mit Zusatzservices durch einen Dummy Einlagerer gebucht
- Tester befindet sich auf der Zusatz-Service buchen Seite
- Tester hat einen Dummy Lagerplatz ausgewählt

Eingaben/Handlungen:

1. Tester wählt einen der angezeigten Zusatzservices aus
2. Tester bricht Buchungsvorgang ab

Erwartete Ausgaben/Reaktionen:

- System leitet Tester zurück zur Liste der Lagerplätze mit Zusatzservices

Erwartete Nachbedingungen:

- Tester befindet sich auf der Zusatz-Service buchen Seite

Prüfanweisungen:

- Sicherstellen das die Weiterleitung korrekt abgelaufen ist



Testkonzept - UseCase Stellplatz suchen

2.3.1. Testfall UC03-01

Vorbedingungen:

- Tester hat ein Testkonto erstellt
- Tester befindet sich auf der Homepage

Eingaben/Handlungen:

1. Tester startet Suchvorgang mit beliebigen Daten

Erwartete Ausgaben/Reaktionen:

- System sucht Lagerplätze mit gewählten Daten
- System listet gefundene Lagerplätze auf

Erwartete Nachbedingungen:

- Tester wird zu Liste der gesuchten Lagerplätze weitergeleitet

Prüfanweisungen:

- Daten der aufgelisteten Lagerplätze sind zu vergleichen mit den gewählten Daten



Testkonzept - UseCase Stellplatz suchen

2.3.2. Testfall UC03-02

Vorbedingungen:

- Tester hat ein Testkonto erstellt
- Tester befindet sich auf der Homepage

Eingaben/Handlungen:

1. Tester startet Suchvorgang mit ungültigen Daten

Erwartete Ausgaben/Reaktionen:

- System findet keinen Lagerplatz

Erwartete Nachbedingungen:

- System informiert über fehlende Lagerplätze

Prüfanweisungen:

- Es ist sicher zu stellen das kein Lagerplatz angezeigt wird

Live Demo