

# **Software Engineering**

Prof. Dr. Claus Pahl

## **Projekt 1**

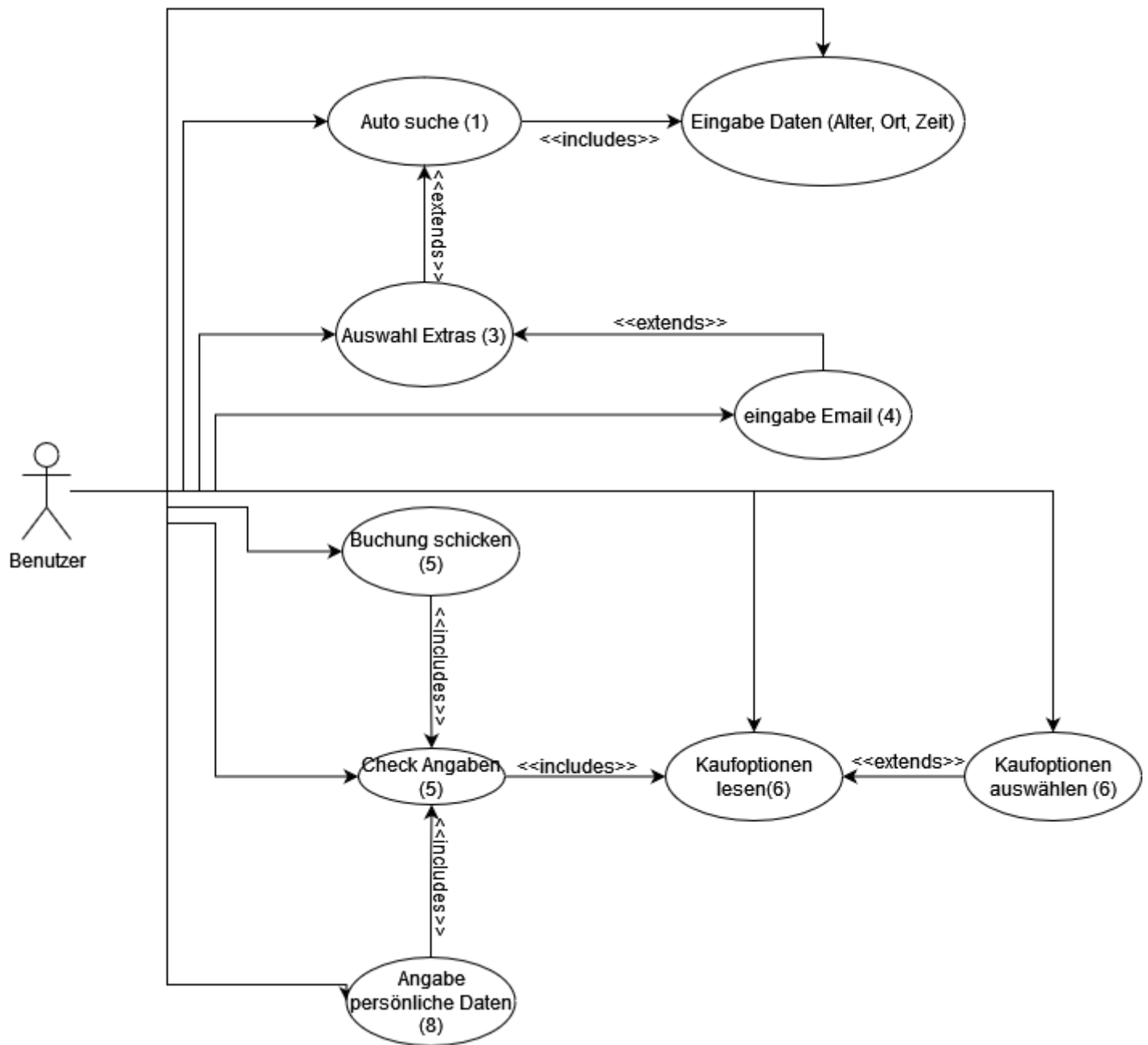
Irene Avezzù  
Edoardo Comizzoli  
Felix Demetz  
Ivana Nworah Bortot

11.04.2023

## Inhaltsübersicht

1. Abstraktes Anwendungsfallmodell .....	3
2. Vereinfachtes abstraktes Anwendungsmodell .....	4
3. Abstraktes Klassenmodell .....	5
4. Abstrakten Entitätsklassendiagramm .....	6
5. Sequenzdiagramm .....	7
6. Klassendiagramm 'Anfang der Buchung' .....	8
7. Klassendiagramm MVC.....	9
8. Zustandsdiagramm EReservation .....	10
9. Komponentendiagramm .....	11
10. Bereitsstellungsdiagramm .....	12

## 1. Abstraktes Anwendungsfallmodell



### A1 Beschreibung Designentscheidung

Entwicklung abstraktes Anwendungsmodell mit ca. 10 wichtigste Anwendungsfälle, die für den Kunden von Interesse sind.

Nachdem der Kunde Daten für die Suchmaschine angegeben hat, gibt die Datenbank eine List mit den dazu passenden Autos an.

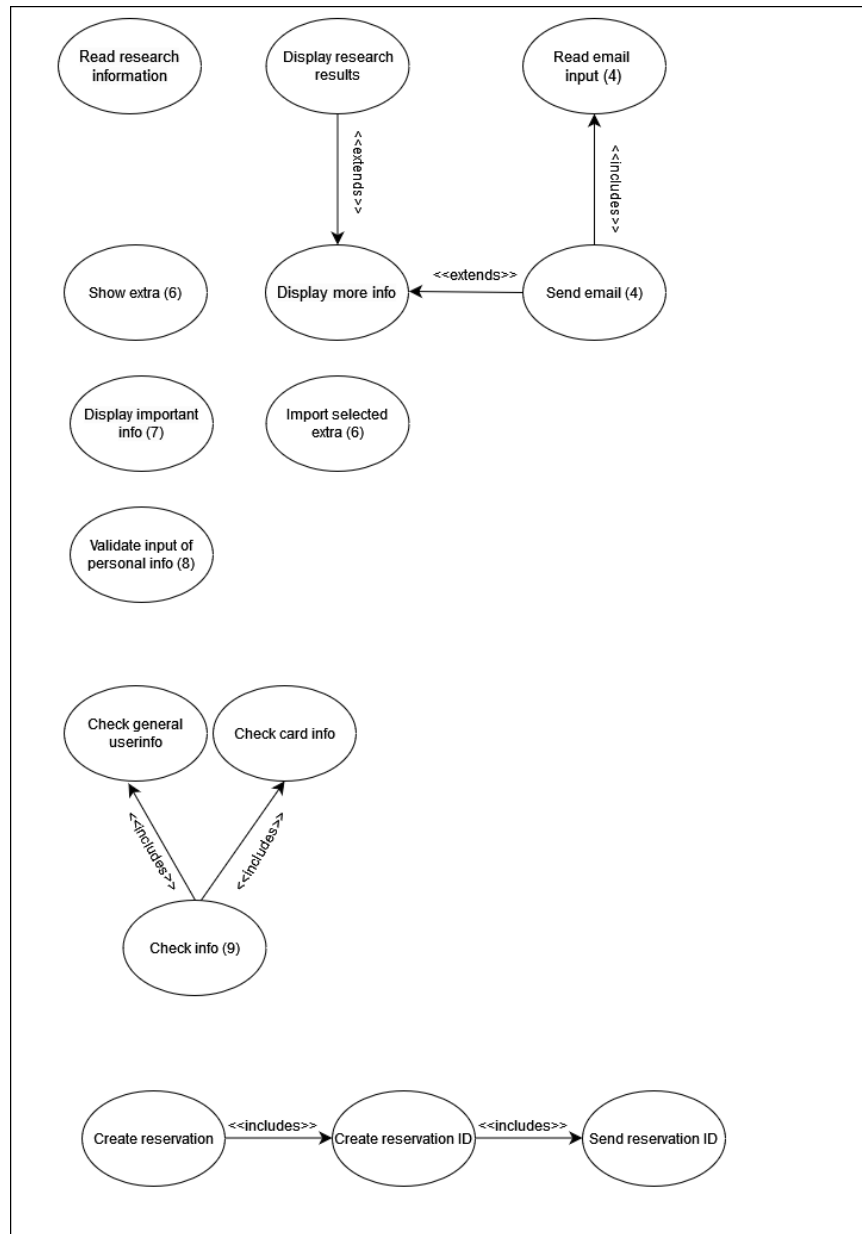
Der Kunde kann sich ein Auto auswählen und mehrere Infos ansehen.

Mit der Angabe der persönlichen Email bekommt er eine schriftliche Zusammenfassung via Email.

Die Buchung des Autos Umfasst persönliche Daten und die Auswahl zusätzlicher Extras.

Vor der Buchung kontrolliert der User die personenspezifischen Daten.

## 2. Vereinfachtes abstraktes Anwendungsmodell



### A2 Beschreibung Designentscheidung

Entwicklung vereinfachtes abstraktes Anwendungsmodell mit ca. 10-15 Anwendungsfälle, die für das System von Interesse sind.

Das Modell beinhaltet nur ein "glückliches Szenario" (d.h. keine Fehler, Probleme, ...)

Aus der Sicht des Systems wird zuerst eine Abfrage der Suchmaschinen-Daten ausgeführt.

Es werden dementsprechend Autos angezeigt. Zu jedem Auto gibt es eine Detaillansicht.

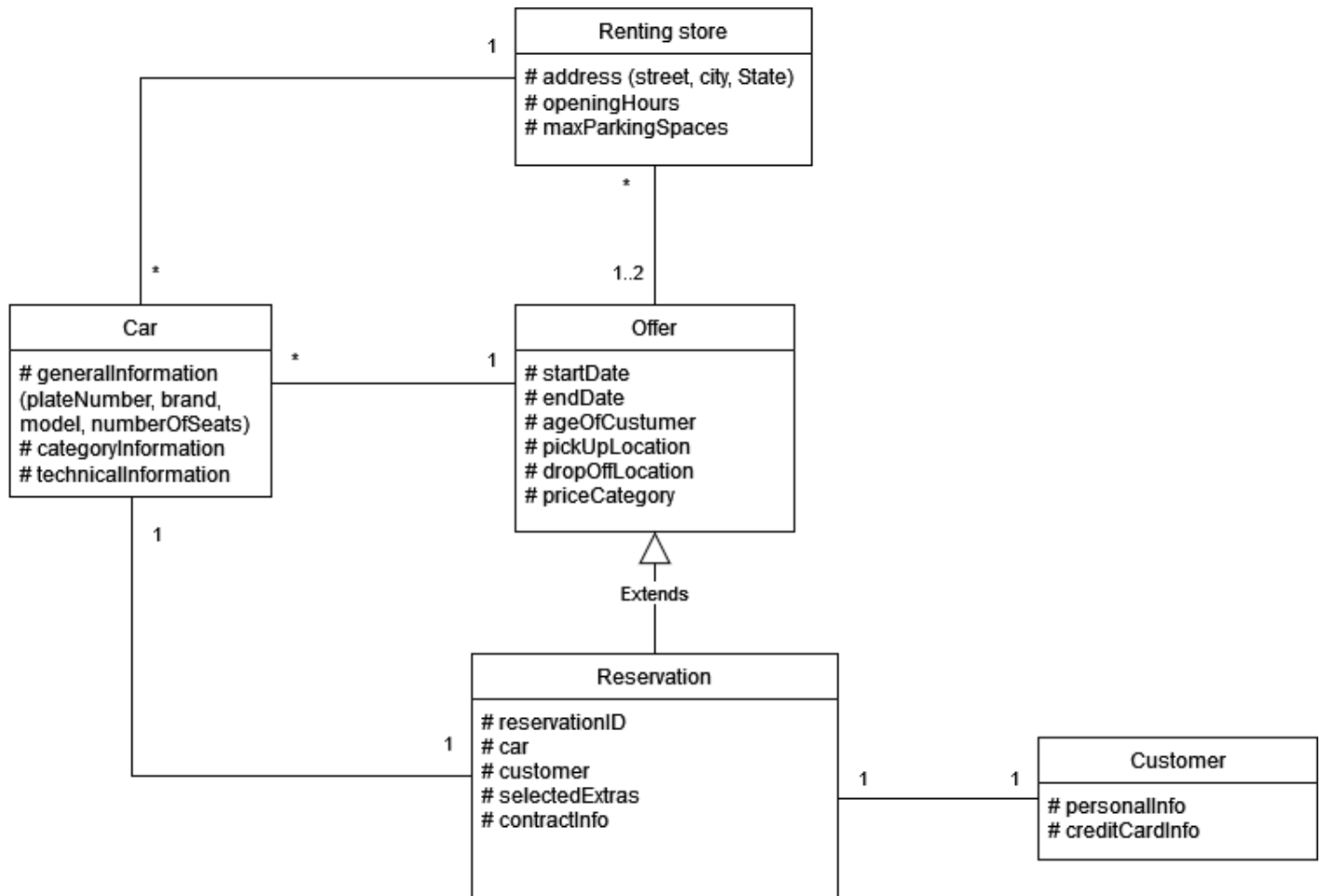
Das System sendet dem Kunden eine Email mit Informationen nachdem es die Kundenemail eingespeist hat.

Das System zeigt bei einer ewt. Buchung zusätzliche Autooptionen an (Versicherungsdetails, Kindersitz, uns...).

Bei einer Buchung sammelt das System die wichtigsten Userinformationen (z.B. Kreditkarte) und kontrolliert dessen Validität.

Bei einem korrektem Input, generiert das System eine Bestätigungsnr. und sendet eine E-Mail der Bestätigung an den Klienten.

### 3. Abstraktes Klassenmodell



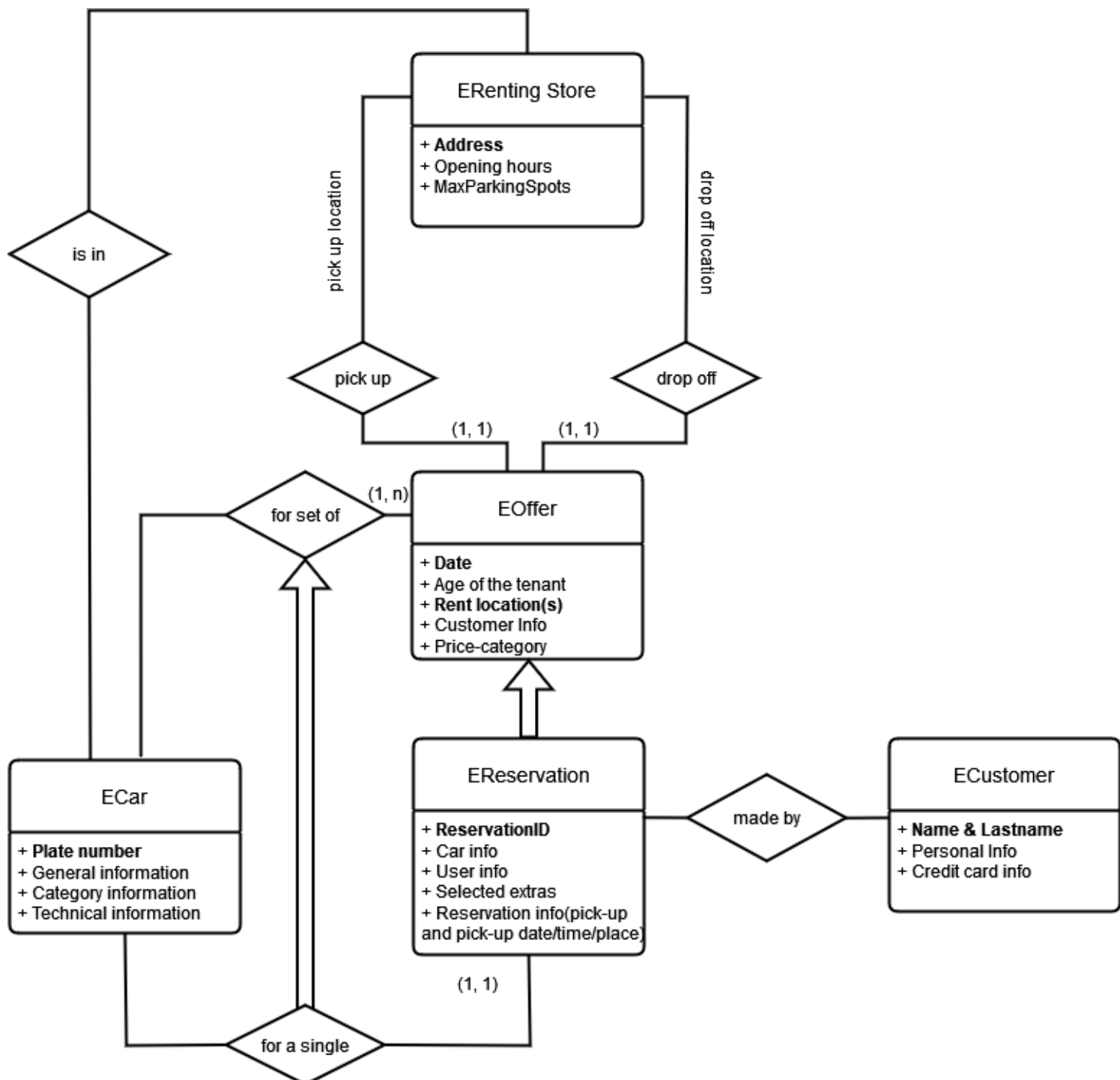
#### A3 Beschreibung Designentscheidung

Entwicklung abstraktes Klassenmodell mit den 5 wichtigsten Objekten bzw. Klassen mit deren Assoziationen und Multiplizitäten. Attribute und Funktionen werden auch dargestellt.

Assoziationen und Multiplizitäten:

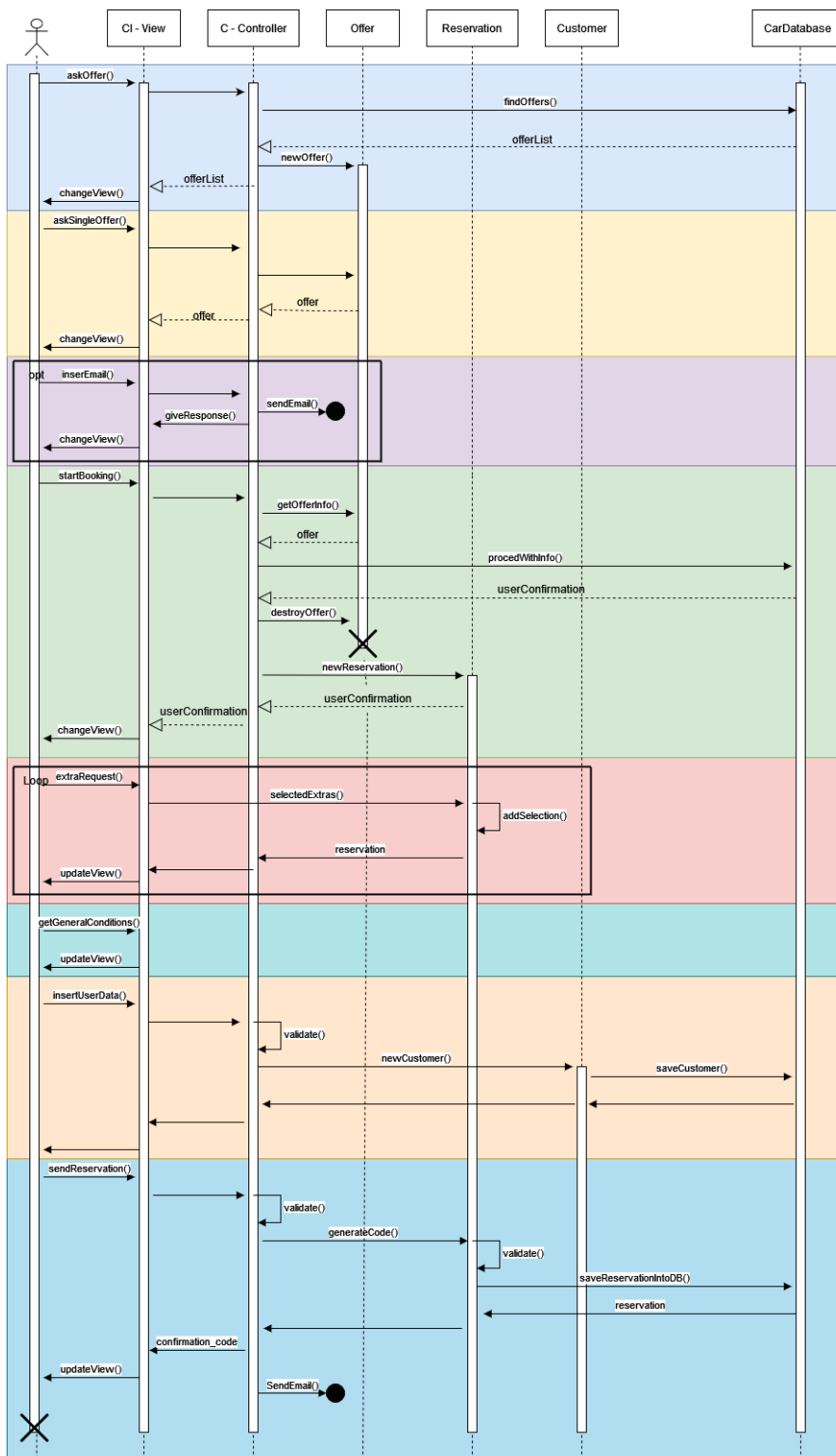
- Buchung mit Auto
- Buchung mit einer oder zwei Filialen
- Buchung mit einem Customer
- Suchergebnis mit Filiale
- Suchergebnis mit einem Customer

## 4. Abstrakten Entitätsklassendiagramm



A4 abstraktes Entitätsklassendiagramm

## 5. Sequenzdiagramm



### Offers request

Der User macht eine Suche, in denen Dauer, Ort und Alter angegeben werden.

### Single Offer

Der User wählt ein Auto aus der Liste aus.

### Insert Email

Optional: der User kann sich eine Email mit der bisherigen Informationen zuschicken lassen.

### start Booking

Der User kann mit der Buchung beginnen, die Informationen von Offer werden von der neu erstellten Reservation aufgenommen.

### choose Options

Der user can Autooptionen auswählen. Jedes mal dass etwas ausgewählt wird, wird die UI verändert.

### agree general conditions

### insertion user data

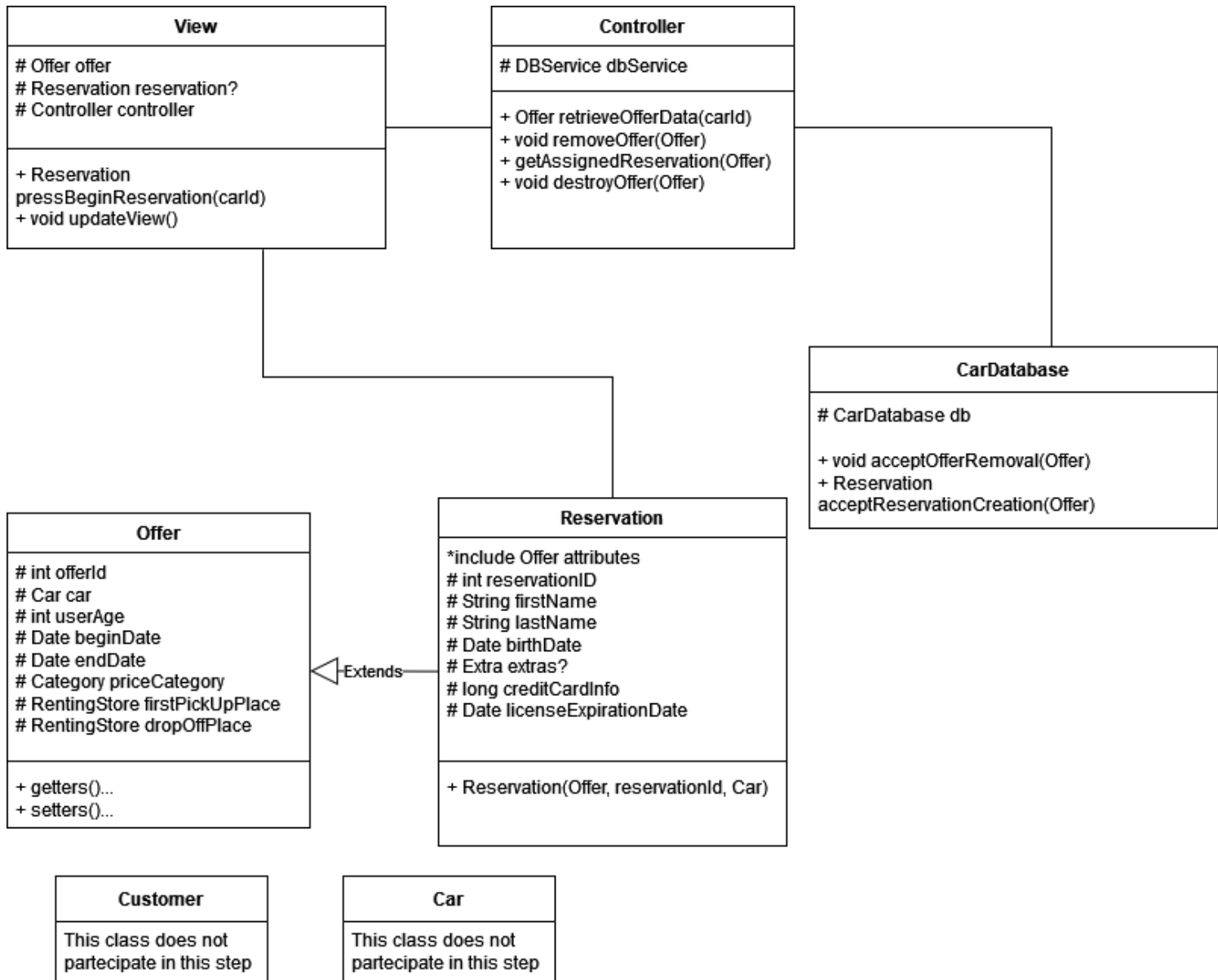
Es wird eine neuer User erstellt, da die persönlichen Informationen des Nutzers angegeben werden.

### confirm reservation

Nachdem der User und das System die korrekte Angabe der Daten bestätigt hat, wird die Buchung bestätigt. Ein Code wird generiert und eine Email an den Kunden verschickt.

Das Team hat sich entschieden das gesamte System darzustellen, damit dem klar wird wie die einzelnen Prozesse miteinander verknüpft sind.

## 6. Klassendiagramm 'Anfang der Buchung'



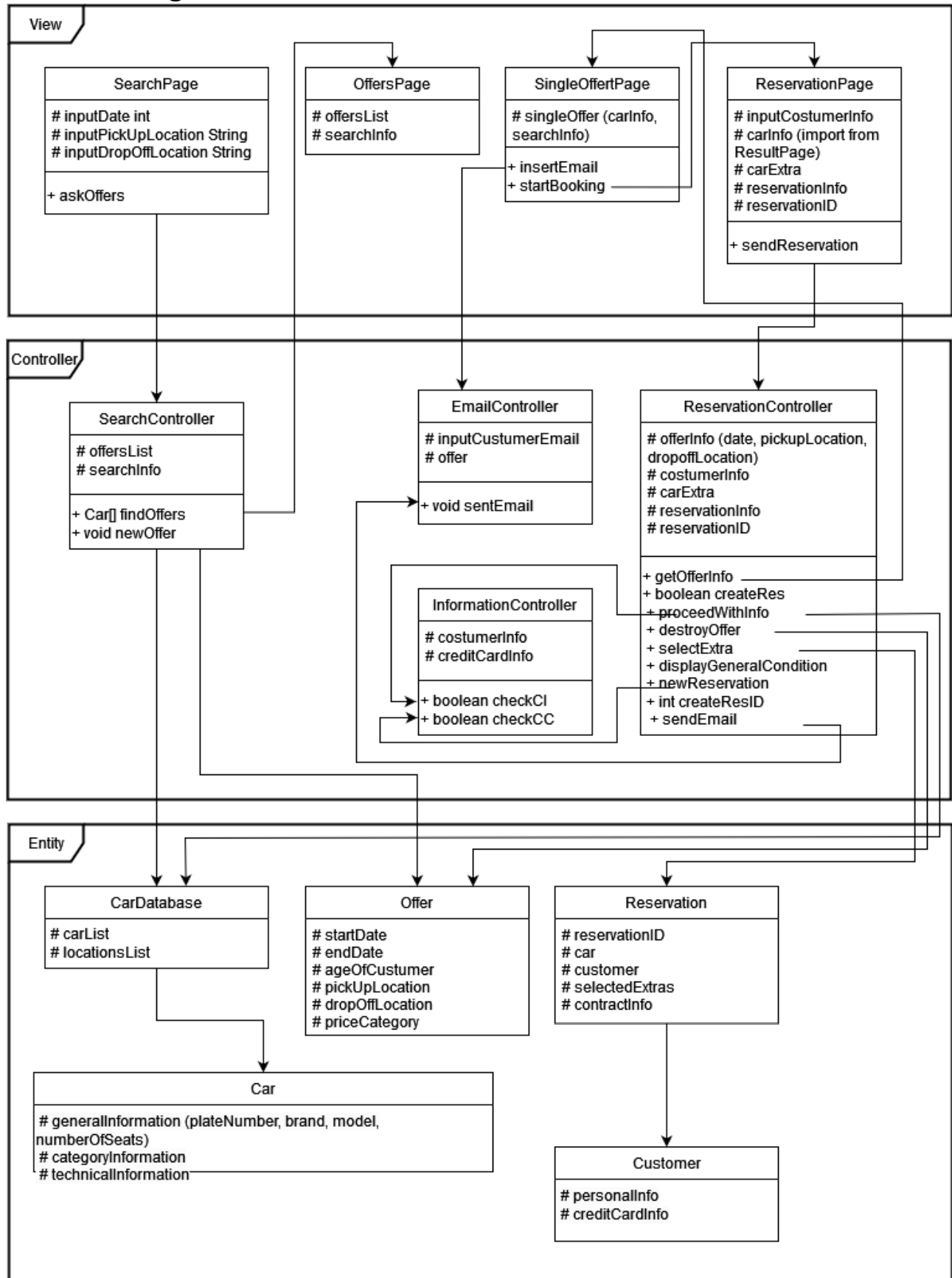
A6 Beschreibung Klassendiagramm "Anfang der Buchung"

(Auswahl einer der in A5 dargestellten Anwendungsfälle)

Die Klasse DBService ist eine Schnittstelle die mit der CarDatabase kommuniziert.  
Die Klasse Offer greift aus einer Liste mit Cars auf einen Car zu.



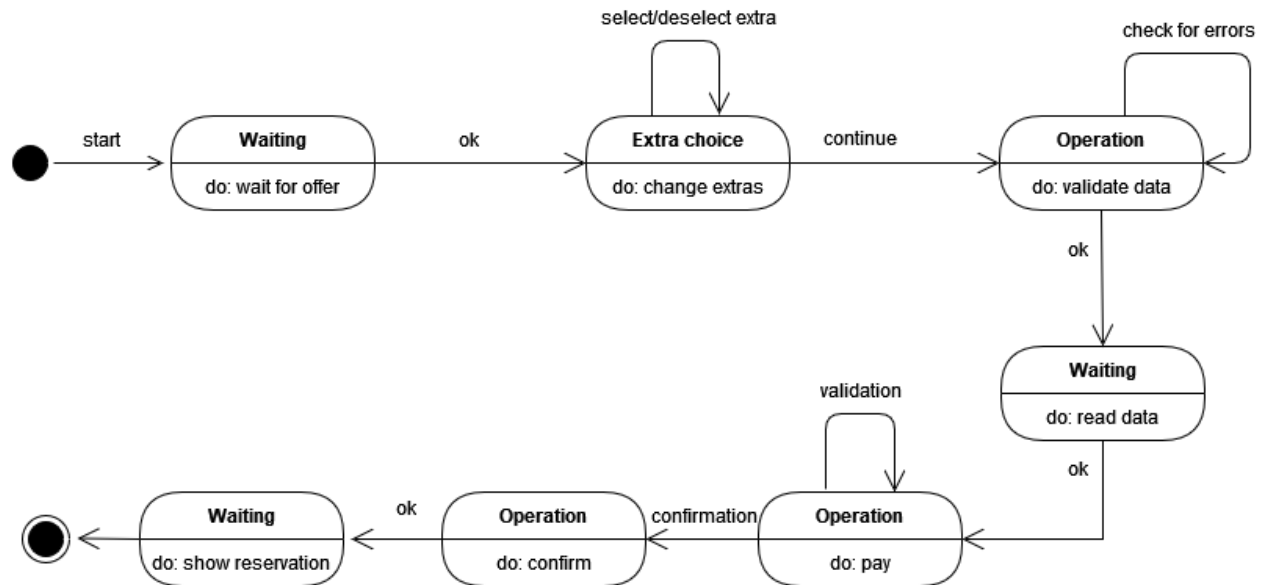
## 7. Klassendiagramm MVC



A7 Beschreibung Klassendiagramm MVC

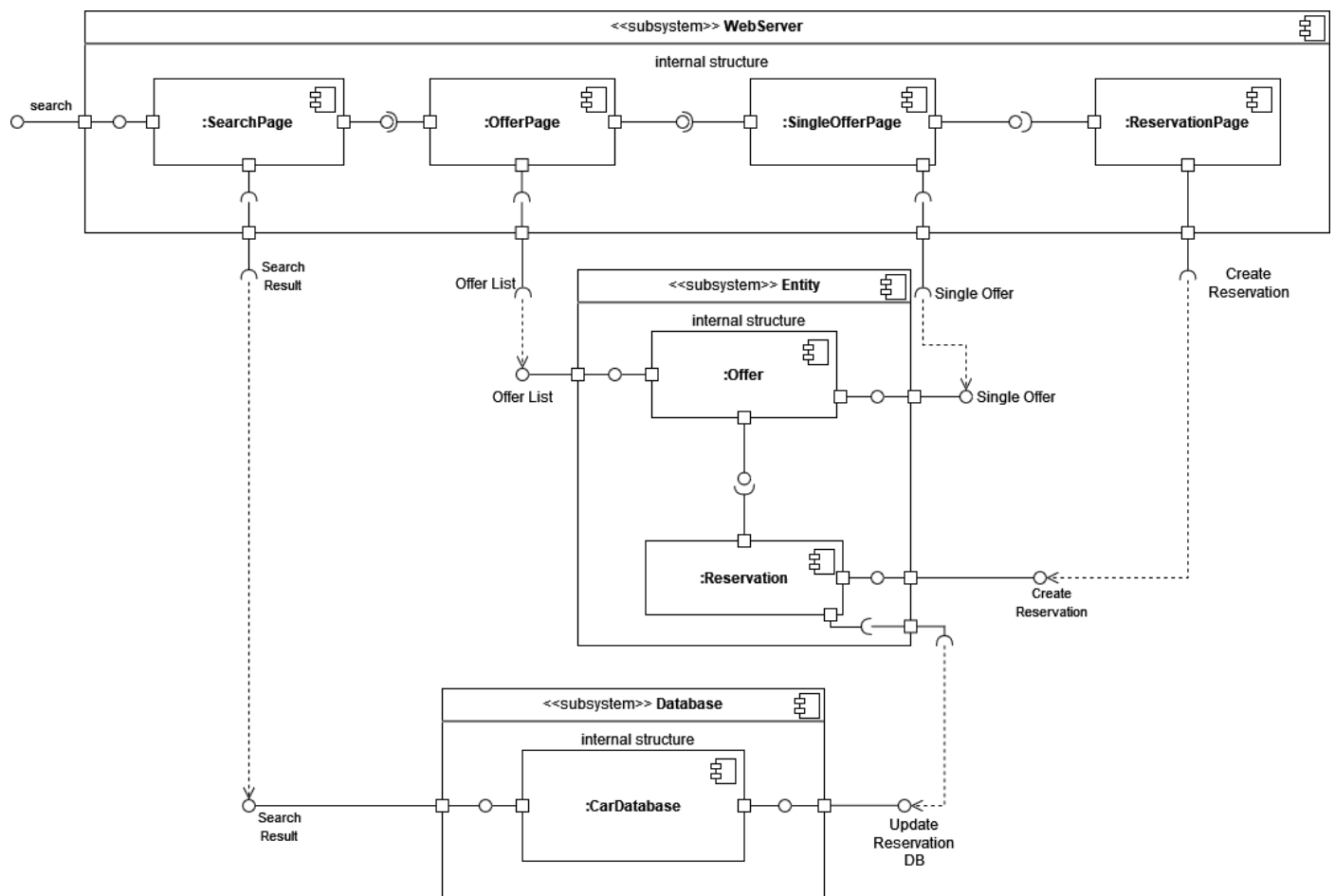
Der Customer kann in der Webseite die Elemente der "View" Ansicht sehen und damit interagieren. Die Controller greifen auf die Entitäten zu um den View zu verändern.

## 8. Zustandsdiagramm EReservation



Das Diagramm zeigt den, nach den jeweiligen Pfeilen sortierten Ablauf dar, in dem sich das System bei der Reservierung befindet.

## 9. Komponentendiagramm



Das Diagramm beinhaltet einige interagierende Komponenten die das Team aus dem A7 der Klassenarchitektur entnommen hat.

Bemerkung: das Subsystem "WebServer" beinhaltet die gleichen Komponenten die im "View" in der A7 zu sehen sind.

## 10. Bereitsstellungsdiagramm

