# Mini-Projekt Softwarearchitektur- und Design

# 3.1 - C4 Diagramm

Ein Bild, das Text, Screenshot, Schrift, Diagramm enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Screenshot, Diagramm, Schrift enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Screenshot, Diagramm, Schrift enthält.

Automatisch generierte Beschreibung

# 5.1 - Domain Model

## Klassendiagramm

Ein Bild, das Text, Screenshot, Diagramm, Schrift enthält.

Automatisch generierte Beschreibung

# 5.1 – Use Cases

1. Verwalten von Kunden:

* Der Sachbearbeiter kann neue Kunden im System erfassen, indem er deren Namen, Adresse und Kundennummer eingibt.
* Der Sachbearbeiter kann bestehende Kunden im System bearbeiten, um Informationen wie Namen oder Adresse zu aktualisieren.
* Der Sachbearbeiter kann Kunden aus dem System löschen.
* Der Sachbearbeiter kann nach Kunden suchen, indem er entweder den Namen oder die Kundennummer eingibt.

2. Verwalten von Autos:

* Der Sachbearbeiter kann Autos von CarRent im System erfassen, indem er Marke, Typ, eindeutige Identifikation und Klasse (Luxusklasse, Mittelklasse oder Einfachklasse) angibt.
* Der Sachbearbeiter kann nach Autos suchen.

3. Zuordnen von Klassen und Tagesgebühren:

* Jede Auto-Klasse (Luxusklasse, Mittelklasse, Einfachklasse) besitzt eine entsprechende Tagesgebühr.

4. Erstellen einer Reservation:

* Der Kunde kann eine neue Reservation erstellen, indem er ein Auto aus einer bestimmten Klasse auswählt und die Anzahl der Tage angibt, für die er das Auto mieten möchte.
* Dabei werden die Gesamtkosten berechnet.
* Wenn die Reservation gespeichert wird, erhält sie eine eindeutige Reservationsnummer.

5. Umwandeln einer Reservation in einen Mietvertrag:

* Bei der Abholung des Autos kann der Sachbearbeiter die bestehende Reservation in einen Mietvertrag umwandeln, um die Miete offiziell zu starten.

# 13 - Domain Driven Design

## 1. Identifizieren Sie die Haupt-Entities im System:

* Customer (Kunde)
* Car (Auto)
* Reservation (Reservation)
* RentalContract (Mietvertrag)
* CarClass (Klasse)

## 2. Identifizieren Sie ValueObjects:

* Address (Addresse des Kunden)
* Car-Info (Marke und Typ des Autos

## 3. Identifizieren Sie DomainEvents:

* ReservationCreated (Reservation erstellt)
* RentalContractCreatedEvent (Mietvertrag erstellt)

## 4. Identifizieren Sie Repositories:

* CustomerRepository (Repository für Kunden)
* CarRepository (Repository für Autos)
* ReservationRepository (Repository für Reservierungen)
* RentalContractRepository (Repository für Mietverträge)
* CarClassRepository (Repository für Auto-Klassen)

## 5. Entwerfen Sie die Aggregate:

* ReservationAggregate (Reservierung)
* CarAggregate (Auto)

## 6. Entwerfen Sie die Schnittstellen für die Repositories:

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung