

Autovermietung:

- Dies ist ein Projekt Aufgabe zur Anforderungsanalyse und zum Softwaredesign.
- Der Anwendungsbereich ist ein webbasiertes Autovermietungs-system.

Problemstellung für den Auftrag - Autovermietung (AV)

Die Aufgabe besteht darin, eine Anforderungsanalyse für eine Autovermietungsanwendung (AV) zu erstellen. Es handelt sich um eine E-Commerce-Anwendung, die es Kunden ermöglicht, Autos über das Internet zu mieten. Der Kunde kann Einzelheiten zur Autovermietung eingeben und ein Angebot für ein Fahrzeug anfordern. Nachdem er den Kostenvoranschlag erhalten hat, kann er mit der Buchung des Fahrzeugs fortfahren, indem er seine Kreditkartendaten angibt. Diese Angaben werden dann überprüft, und die Anwendung sendet die Bestätigung der Autovermietung per E-Mail an die vom Kunden gewählte E-Mail-Adresse und aktualisiert dann die Datenbank der Autovermietung entsprechend.

Für die Zwecke dieses Auftrags ist der Umfang des AV-Systems auf die folgenden Funktionen beschränkt:

1. Die Kunden sollten die Möglichkeit haben, einen Kostenvoranschlag für die Anmietung eines Fahrzeugs anzufordern. Um dies zu erleichtern, sollten sie den gewünschten Abhol- und Rückgabeort, das Abhol- und Rückgabedatum und ihr Alter eingeben können, um eine Liste der verfügbaren Fahrzeugtypen auf der Grundlage dieser Suchkriterien zu erhalten.
2. Die Suchergebnisse enthalten Angaben zu Fahrzeugtypen, Preisen für die telefonische Buchung und Preisen für die Online-Buchung.
3. Ausgehend von den angezeigten Suchergebnissen sollte der Kunde in der Lage sein, detailliertere Informationen über einen bestimmten Fahrzeugtyp anzufordern, einschließlich der Fahrzeugmarken und -modelle des gewählten Typs, der Sitzplatzkapazität und der Gepäckkapazität.
4. Ausgehend von den angezeigten Suchergebnissen sollte der Kunde auch die Möglichkeit haben, sich das Angebot für die Anmietung eines ausgewählten Fahrzeugtyps per E-Mail zusenden zu lassen. Zu diesem Zweck muss der Kunde seine E-Mail-Adresse angeben. Der Versand der E-Mail wird dem Kunden bestätigt.
5. Von den angezeigten Suchergebnissen aus sollte der Kunde auch in der Lage sein, mit der Buchung eines Fahrzeugs fortzufahren. Nach Auswahl dieser Option kann der Kunde überprüfen, ob die Angaben für die Anmietung korrekt sind.
6. Vor der Buchung kann der Kunde aus einer Reihe von Optionen wählen, wie z. B. die Reduzierung der Selbstbeteiligung gegen eine zusätzliche Gebühr, den Verzicht auf einen Stornierungsschutz oder die Eingabe eines Aktionscodes. Außerdem können einige Extras wie zusätzliche Fahrer, Babykapseln und Kindersitze angegeben werden.
7. Der Kunde muss weitere wichtige Informationen erhalten, z. B. Altersbeschränkungen, Kraftstoffbedingungen und andere allgemeine Beschränkungen sowie Informationen über Versicherungsdetails und Abholanweisungen.
8. Noch vor der Buchung muss der Kunde persönliche Daten (wie Name, Adresse, Telefonnummer und E-Mail-Adresse) sowie Zahlungsinformationen (Kreditkartendaten) eingeben.
9. Bei der Bearbeitung der Buchungsanfrage prüft das System die Vollständigkeit und Korrektheit der vom Kunden eingegebenen Daten (insbesondere die Kreditkarteninformationen). Nach erfolgreicher Validierung der eingegebenen Daten bestätigt das System die Buchung mit einer Mietbestätigungsnummer. Als zusätzliche Bestätigung der Buchung wird eine E-Mail an die E-Mail-Adresse des Kunden gesendet.

Im Allgemeinen würde das AV-System über ähnliche Funktionen verfügen wie andere typische Online-Mietwagensysteme. Um zusätzliches Fachwissen zu erlangen, können Sie vorgefertigte COTS (Commercial-Of-The-Shelf)-Lösungen für Autovermietungssoftware studieren, wie <https://www.prohire.co.uk/> oder <https://www.assetmindersolutions.co.uk/>.

ANFORDERUNGSANALYSE

Aufgabe 1: Entwickeln Sie ein abstraktes Anwendungsfallmodell für das AV-System. Wenden Sie eine solche Abstraktionsebene an, dass Ihr Modell nur die wichtigsten Anwendungsfälle identifiziert, die für den Kunden von Interesse sind (d.h. das Modell sollte die Sichtweise des Akteurs (und nicht die des Systems) auf die Anwendungsfälle einnehmen. Verwenden Sie <<extend>>- und/oder <<include>>-Beziehungen, um den Arbeitsablauf (zeitliche und andere Verarbeitungs-abhängigkeiten) zwischen Anwendungsfällen zu visualisieren.

Aufgabe 2: Entwickeln Sie ein verfeinertes Anwendungsfallmodell für das AV-System. Wenden Sie eine solche Abstraktionsebene an, dass Ihr Modell aus zehn bis fünfzehn Anwendungsfällen besteht. Modellieren Sie für ein "glückliches Szenario", d. h. berücksichtigen Sie keine Ausnahmebedingungen, Fehlermeldungen usw.

Stellen Sie sicher, dass die Anwendungsfälle die Anforderungen der Problemstellung abdecken. Zeigen Sie Anwendungsfälle und die <<erweitern>> und <<einschließen>> Beziehungen zwischen ihnen, so dass Sie zeitliche und bedingte Abhängigkeiten zwischen Anwendungsfällen ausdrücken. Zeigen Sie keine Akteure.

Im Gegensatz zu Aufgabe 1 sollten die Anwendungsfälle in Aufgabe 2 die Perspektive des internen Systems einnehmen, nicht die Perspektive des Akteurs (d.h. die Anwendungsfälle sollten so benannt werden, dass sie die Funktionen des Systems widerspiegeln, da sie schließlich implementiert werden müssen).

Aufgabe 3: Entwickeln Sie ein abstraktes Klassenmodell für das AV-System. Wenden Sie eine solche Abstraktionsebene an, dass Ihr Modell etwa fünf wichtige Konzepte (oder Objekte) aus den Anwendungsbereich identifiziert.

Zeigen Sie Assoziationen zwischen den Konzepten und bestimmen Sie ihre Multiplizitäten. Achten Sie auf ein minimales Modell, d. h. zeigen Sie keine Assoziationen, die sich aus den bereits im Modell vorhandenen Assoziationen ableiten lassen. Erläutern Sie die Bedeutung des Diagramms.

Aufgabe 4: Entwickeln Sie ein Entitätsklassendiagramm (d. h. nur Entitätsklassen) für AV. Stellen Sie dem Namen jeder Klasse den Buchstaben "E" voran (d.h. ECustomer, nicht Customer). Wenden Sie eine Abstraktionsebene an, die ausreicht, um die Datenstrukturen zu modellieren, auf die in den in der Aufgabenstellung aufgeführten Anforderungen Bezug genommen wird (dies sollte zu etwa zehn Entitätsklassen führen)

Bitte Hauptattribute in Klassen anzeigen (Typen und Sichtbarkeit müssen nicht angegeben werden, ebenfalls keine Operationen). Geben Sie die Beziehungen zwischen den Klassen und ihre Multiplizitäten an. Erläutern Sie im Text alle nicht offensichtlichen Modellierungsentscheidungen.

SYSTEMARCHITEKTUR UND -ENTWURF

Aufgabe 5: Entwickeln Sie ein Sequenzdiagramm für einen Anwendungsfall, den Sie aus Ihren vorherigen Diagrammen auswählen können.

Annahme des Szenarios "glücklicher Pfad", d. h. es besteht keine Notwendigkeit, alternative Flüsse zu berücksichtigen. Wenden Sie eine Namenskonvention für Klassen an, die einen einzelnen Buchstaben als Präfix zur Identifizierung von Präsentations- (P), Kontroll- (C) und Entitäts- (E) Objekten verwendet. Interaktionen des Akteurs (Kunden) anzeigen. Zeigen Sie Nachrichtenargumente (nur Namen, keine Typen). Erklären Sie alle nicht offensichtlichen Modellierungsentscheidungen.

Verwenden Sie für Objekte, die im Sequenzdiagramm instanziiert werden, entweder die Konstruktornamen oder die new()-Operationen und stellen Sie sicher, dass die Instanziierungsnachrichten an die Objektboxen und nicht an die Lebenslinien der Objekte gehen. Fügen Sie ein CarDatabase-Objekt ein (dies könnte ein Akteur sein), um "Fahrten" zur Datenbank zu zeigen.

Erlauben Sie nur Steuerobjekten die Kommunikation mit der CarDatabase. Sie können davon ausgehen, dass die Anwendung zu Beginn des Anwendungsfalls bereits mindestens eine Steuerklasse instanziiert hat, die in der Lage ist, die Ereignisse des Benutzers auf der Fahrzeugsuchseite abzufangen.

Aufgabe 6: Entwickeln Sie ein Klassendiagramm, das Ihrem Sequenzdiagramm aus Aufgabe 5 entspricht. Zeigen Sie die Operationen in den Klassen (gib die Namen der Argumente und Rückgabetypen der Operationen an). Zeigen Sie Konstruktoren für die in Aufgabe 5 gezeigten (instanziierten) Klassen. Geben Sie Beziehungen zwischen Klassen an. Verwenden Sie unidirektionale Assoziationen, es sei denn, die Kooperationen in Aufgabe 5 beinhalten eine bidirektionale Kommunikation. Erkläre nicht offensichtliche Modellierungsentscheidungen.

Beachten Sie, dass CarDatabase in Aufgabe 5 keinen echten Akteur darstellt. Er wurde nur aus Gründen der Modellierung verwendet und ist nicht Teil der Anwendung (er stellt eine Server-Datenbank dar, mit der eine Client-Anwendung kommuniziert). Auf der anderen Seite ist Customer ein (externer) Akteur, der auch intern als Klasse (ECustomer) existiert.

Aufgabe 7: Betrachten Sie eine Schichtenarchitektur (Präsentation, Steuerung/Verarbeitung, Daten/Entität/Modell). Zeichnen Sie ein Klassenmodell, das die AV-Subsysteme (Schichten oder Subsysteme innerhalb von Schichten), die AV-Klassen innerhalb der Subsysteme und alle Beziehungen (Assoziationen und Abhängigkeiten) zwischen Subsystemen und Klassen darstellt. Erläutern Sie den Platz und die Rolle jeder Nicht-Entitätsklasse in der architektonischen Lösung (die oben erwähnten Entitätsklassen müssen nicht erläutert werden). Um Ihnen bei der Erklärung zu helfen, zeigen Sie bitte die Hauptmethoden innerhalb der Nicht-Entitätsklassen in Ihrem Klassendiagramm.

Aufgabe 8: Beziehen Sie sich auf eine Klasse, die Vermietungen repräsentiert (sie könnte ERental heißen, z. B. in Aufgabe 4). Überlegen Sie, in welchen Zuständen sich das Objekt ERental befinden kann (diese Zustände werden im Statusattribut von ERental festgehalten). Zeichnen Sie ein Zustandsdiagramm für die Klasse ERental. Geben Sie die notwendigen Erklärungen im Text.

SOFTWAREENTWICKLUNG UND -BEREITSTELLUNG

Aufgabe 9

Entwickeln Sie ein Komponentendiagramm für die AV-Anwendung. Dies könnte zeigen, wie einige ausgewählte Komponenten als Anbieter und Nachfrager von Diensten zusammenarbeiten. Sie können von der Klassenarchitektur in Aufgabe 7 ausgehen oder zwei oder drei beliebige interagierende Komponenten auswählen.

Aufgabe 10

Zeichnen Sie ein Bereitstellungsdiagramm. Gehen Sie davon aus, dass die AV-Anwendung als webbasierte Anwendung eingesetzt werden soll. In diesem Fall werden Sie Web- und Anwendungsserver wie Tomcat einsetzen, um Ihre eingesetzte Softwarekomponente zu verwalten. Sie würden eine .jar-Datei bereitstellen, wenn Ihre Entwicklungssprache Java ist.