



Technische Universität Berlin

Software and Embedded Systems Engineering Group

Prof. Dr. Sabine Glesner

www.sese.tu-berlin.de    Sekr. TEL 12-4    Ernst-Reuter-Platz 7    10587 Berlin



# Softwaretechnik und Programmierparadigmen WiSe 2022/2023

Prof. Dr. Sabine Glesner  
Milko Monecke  
Simon Schwan

## Übungsblatt 11

Es gilt wieder die aus den vorhergehenden Übungsblättern bekannte textuelle Anforderungsspezifikation:

Eine Autowerkstatt möchte die Abfertigung ihrer Aufträge komfortabel mit einer Software verwalten. Dazu kann ein Mitarbeiter<sup>1</sup> im System Kundys anlegen und ihnen Fahrzeuge zuordnen. Für ein neues Kundy wird ein Name, eine Telefonnummer und eine Rechnungsadresse gespeichert und die Fahrzeuge werden mit Kennzeichen und Typ registriert.

Ein Auftrag kann entweder eine Inspektion, ein Reifenwechsel oder eine Reparatur sein. Einem neuen Auftrag wird ein Preis, ein Fahrzeug und automatisch ein Datumstempel zugewiesen. Eine Reparatur erhält außerdem eine genaue Tätigkeitsbeschreibung. Ein Auftrag kann vom Mitarbeiter als beendet markiert werden. In diesem Fall wird das Kundy automatisch vom System benachrichtigt. Außerdem wird für den Auftrag vermerkt, welches Mitarbeiter ihn beendet hat.



Um Missbrauch vorzubeugen, müssen sich die Mitarbeiters am Browser mit ID und Passwort sicher anmelden. Ein Administrator kann Mitarbeiters anlegen und entfernen.

---




<sup>1</sup>Entgendern nach Phettberg. Eine Erklärung und Begründung dazu gibt es auf der ISIS-Seite zum Kurs.

---

### Schlüssel:

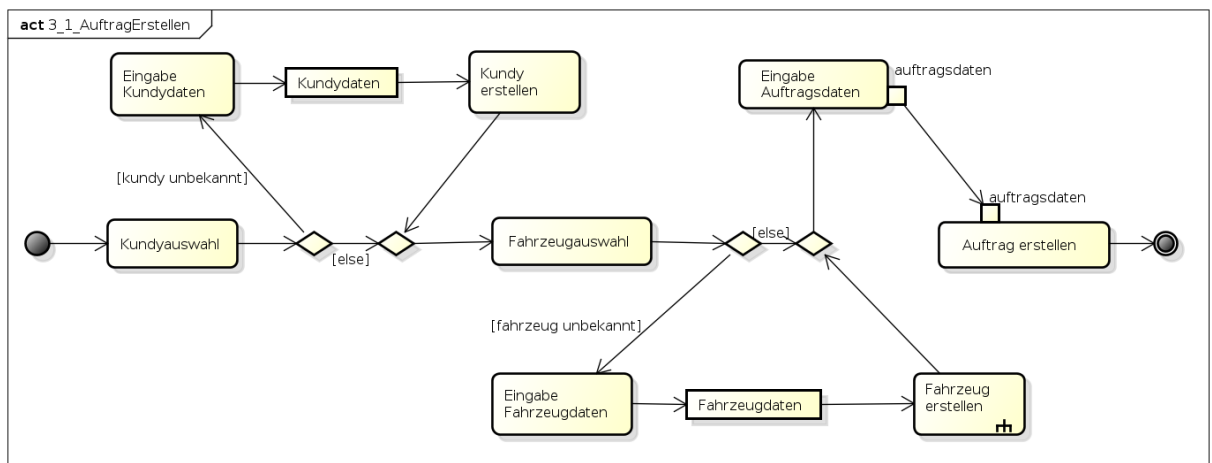
-  Ein ergänzendes Video wird zur Vor- oder Nachbereitung veröffentlicht.
-  Wird im Tutorium besprochen.

## 1. Workflow-Modellierung

- Erstellt ein Aktivitätsdiagramm zur Modellierung des Arbeitsablaufes (Workflows) eines Mitarbeiters, wenn ein Kunde die Autowerkstatt betritt und sein Fahrzeug zur Reparatur abgibt. Dabei sollen fehlende Kund- und Fahrzeugdaten im Bedarfsfall direkt mit angelegt werden. 
- Wie müsste das Aktivitätsdiagramm erweitert werden, um die Abarbeitung beliebig vieler Kunden an einem Arbeitstag zu modellieren? 
- Welche Aktionen könnten sinnvoll als Aktivitätsaufrufe in eurem Aktivitätsdiagramm modelliert werden? Welchem Zweck dient eine detailliertere Modellierung? 

– Lösung Anfang – 30 min –




a) Beispiel-Lösung:



- bspw. Kante zurück zu Kundenauswahl, wenn nicht Feierabend
- Für jeden Use Case/ jede Aktion (Kundy erstellen, Fahrzeug erstellen, Auftrag erstellen) könnte ein eigenes Aktivitätsdiagramm erstellt werden. Damit könnte die Implementierung vorbereitet oder dokumentiert werden. In beiden Fällen macht man sich zu Nutze, dass die graphische Darstellung intuitiv verständlich ist und der Ablauf damit oft einfacher zu verstehen und nachzuvollziehen. Möglicher vorteil: Randfälle werden schon dokumentiert, damit der Programmierer sie nicht vergisst.

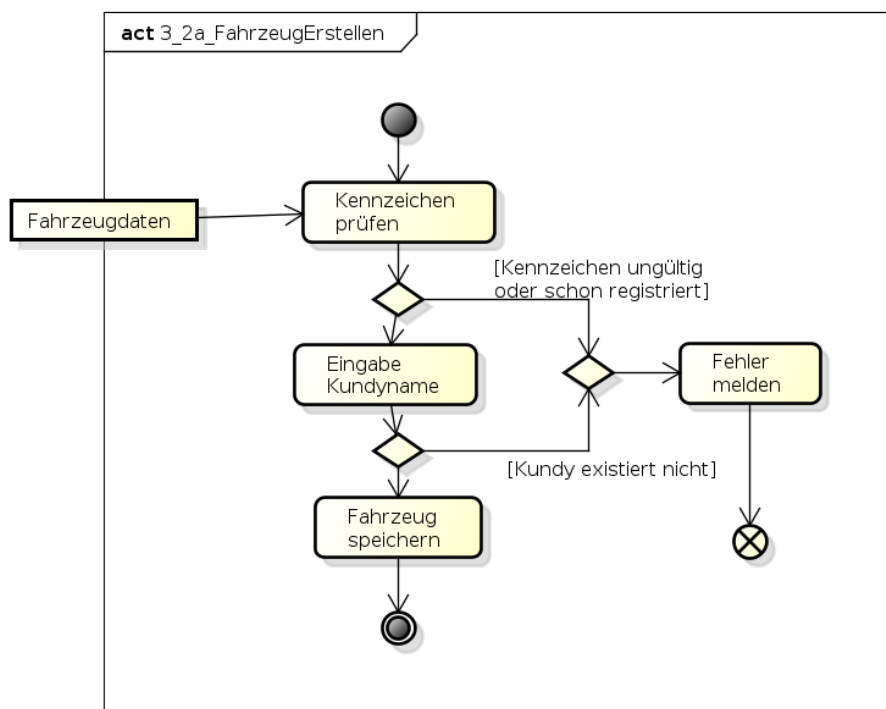
– Lösung Ende –

## 2. Kontrollfluss-Modellierung

- a) Erstellt ein Aktivitätsdiagramm zur Modellierung des Kontrollflusses innerhalb einer Aktion, die in Aufgabe 1 als Aktivitätsaufruf modelliert wird. Ggf. können Designentscheidungen getroffen werden, z.B. wie man nach Kundys, Fahrzeuge oder Aufträge suchen kann und welche mögliche Fehler berücksichtigt werden. Haben beide Modelle ein gleiches Detaillierungsgrad? 
- b) Erstellt ein Aktivitätsdiagramm zur Modellierung des Kontrollflusses innerhalb der Operation **auftragBeenden**. Dabei sollen folgende Zusatzanforderungen berücksichtigt werden:  
- Es soll eine Rechnung für die gegebene Auftrags-ID erstellt und verschickt werden.
  - Beim Erstellen der Rechnung sollen die einzelnen Posten genau eingegeben werden. Dabei soll die Arbeitszeit in Minuten eingegeben werden oder die Kosten für ein Ersatzteil bei der Warenverwaltung angefragt werden.
  - Die Rechnung soll jeweils um 0 Uhr an das Kundy und an die Buchhaltung verschickt werden.

– Lösung Anfang – 45 min –

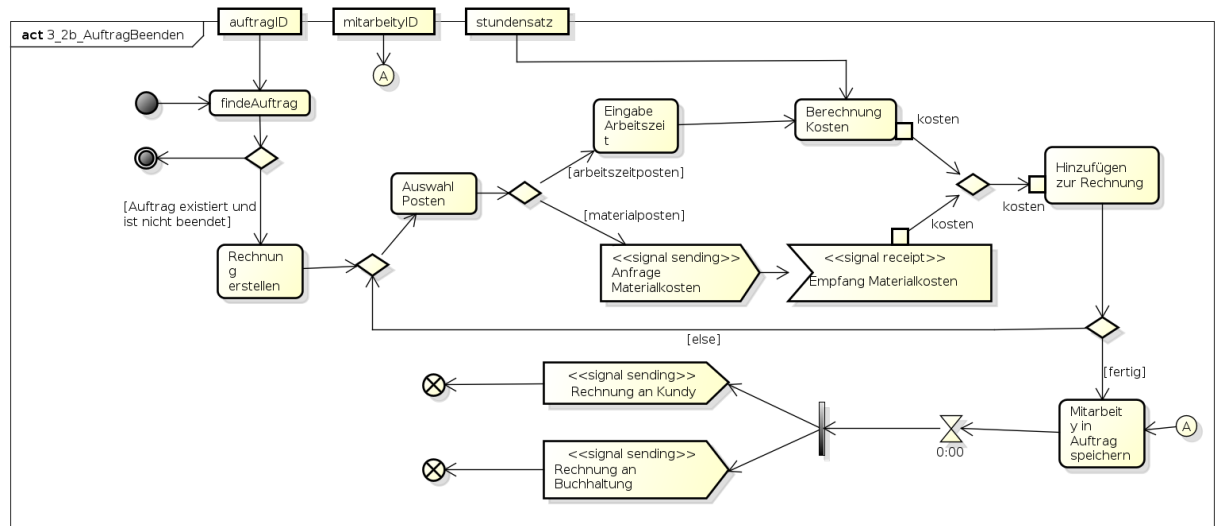
a) Beispiel-Lösung:



Im Video werden die mögliche Detaillierungsgrade besprochen, und welche Aspekte modelliert oder weggelassen werden können. Oft sind Modellen für eine Implementierung detaillierter. Mehr Details helfen nicht immer, z.B. beim FahrzeugErstellen würden weitere Details zum Datenfluss (Fahrzeugdaten auch zu “Fahrzeug speichern”, explizite Pins zum Kundenname) eher verwirren. ▶

Alternativ könnte man den Ablauf textuell beschreiben. I.d.R. sind Diagramme für komplexere Abläufe verständlicher. Es gibt keine feste Grenze, ab wann ein Diagramm sich lohnt. ▶

b) Beispiel-Lösung:



Im Video wird eine fehlerhafte Version des Diagramms gezeigt und auf den einzelnen Fehler eingegangen. ▶

– Lösung Ende –