



2. Domainklassen

3. Abnahmetests

von
Prof. Dr. Wolfgang Weber

[Wi05, S. 227f], [Vi05, S. 48f]



2. Domainklassen

- Domainklassenmodell gehört genau so wie das Anwendungsfallmodell zur Anforderungsermittlung und ist auch Grundlage des Reviews mit den Anwendern / Stakeholders.

- Bei Beschreibung der Anwendungsfälle stößt man auf materielle und immaterielle Dinge mit zu verwaltenden Informationen.
 - Einbettung dieser Informationen ins Domainklassenmodell, d. h.
 - Identifizieren von Objekten, Klassen, Attributen, Assoziationen, Aggregationen
 - nur Informationen, d. h. keine Operationen
 - keine Navigationsrichtungen
 - evtl.: Gemeinsamkeiten => Generalisierungsbeziehungen
 - evtl. Spezifikation weiterer Eigenschaften der Klassen, die nicht in Diagrammform ausdrückbar => Textform



Domainklassen

- Regeln:
 - jede Klasse muss in mindestens einem Anwendungsfall vorkommen
 - jede Domainklasse sollte mindestens ein Attribut haben



3. Abnahmetests

Was muss getestet werden?

- gemäß FURPS-Modell:
 - **F**unctionality (Funktionalität)
 - **U**sability (Benutzbarkeit)
 - **R**eliability (Zuverlässigkeit)
 - **P**erformance (Leistung)
 - **S**upportability (Wartbarkeit)

Wie teste ich die Funktionalität?

- Ich muss Testdaten für die Eingaben finden und die Ausgabedaten dazu herleiten.
 - Annahme: wir haben Anwendungsfälle beschrieben.
 - In Anwendungsfällen sind die Eingaben ins System und Ausgaben vom System dokumentiert
 - Beim Abnahmetest lässt man die einzelnen Anwendungsfälle ein- oder mehrmals ablaufen. Um nachzuprüfen, ob der Test richtig abläuft, benötigt man Eingabedaten und die zu den Eingabedaten passenden Ausgabedaten.
 - **Frage:** Wie finde die Testfälle und die dazugehörenden Daten?

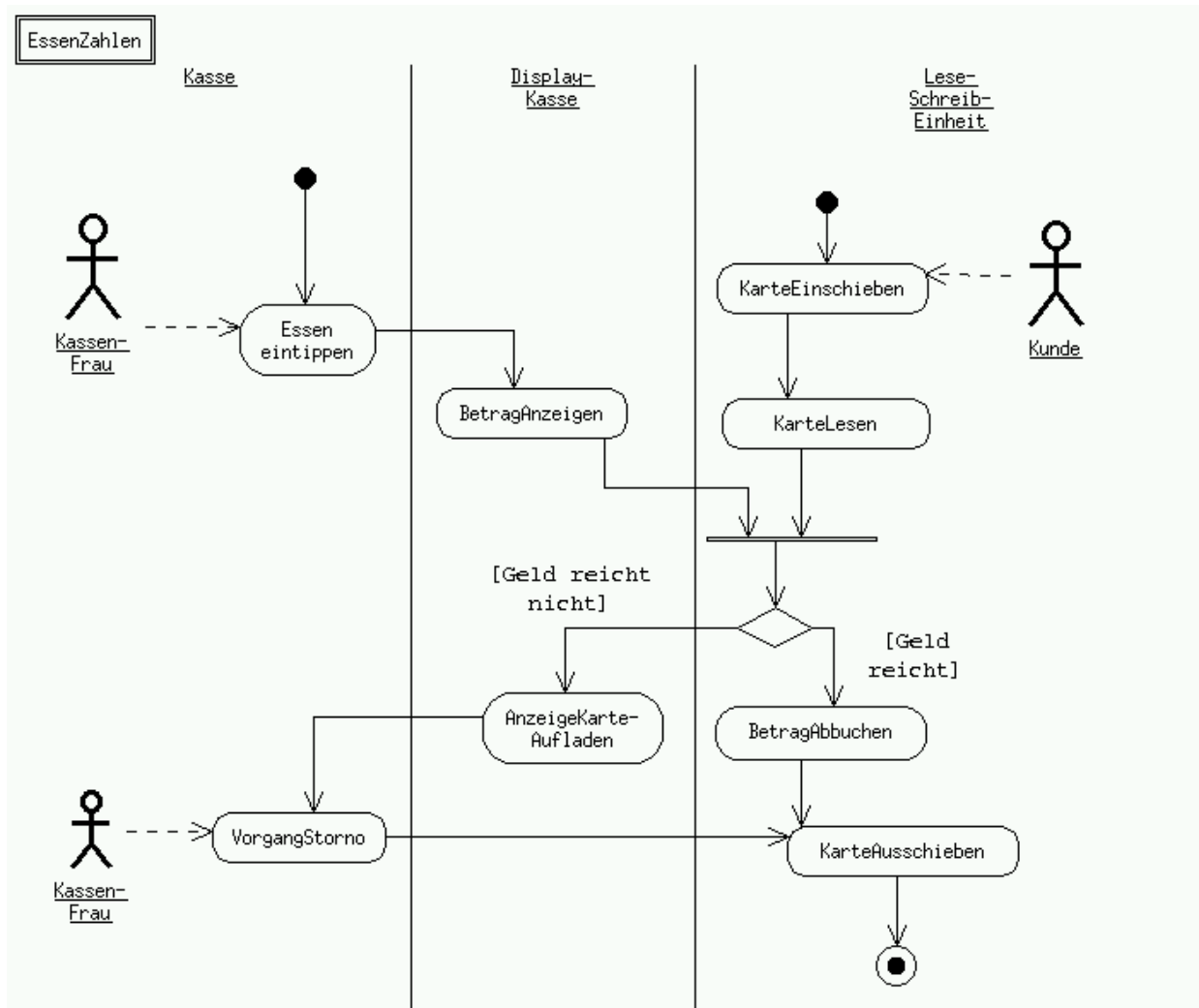


Abnahmetests

- Man kann das System nicht mit allen möglichen Eingabedaten testen.
 - ⇒ man teilt die Menge der Testfälle in Klassen ein, so dass man sagen kann:
 - Falls ein Testfall aus der Menge der Testfälle der Klasse richtig abläuft, so laufen auch die anderen mit großer Wahrscheinlichkeit richtig ab.
- 2 Arten der Einteilung in Klassen:
 - Zweigtesten
 - Pfadtesten

- Zweigtesten
 - durch mindestens einmaliges Durchlaufen aller Action-Steps in Basic und Alternative Courses
 - oder man erstellt zu dem Ablauf im Use Case ein Aktivitätendiagramm, das auch alle Ausnahme- und Fehlerverzweigungen enthält, und bestimmt Testfalldaten, so dass jeder der Zweige des Aktivitätendiagramms durchlaufen wird (Im Kapitel 1 ist ein Anwendungsfall "Essen Zahlen" mit Hilfe eines Aktivitätendiagramms beschrieben.)

Use-Case-Beschreibung mit Aktivitätsdiagrammen



[BGKRSW05]

- Pfadtesten
 - Jedes mögliche Szenario wird mindestens einmal getestet
 - d. h. jeder mögliche Pfad in den zu den Use-Cases gehörenden Aktivitätendiagrammen wird getestet.
 - Was ist der Unterschied zwischen Zweig und Pfad?
 - siehe Tafelbild

- Schwierigkeit bei beiden Verfahren: Ich habe oft nicht das gesamte System, sondern nur einige Szenarien zu den Use-Cases definiert.
=> Ich muss mir überlegen, welche Szenarien ich zu Testzwecken noch zusätzlich spezifizieren sollte.