

Software-Engineering Bahn Zahlsystem

Daniela Pointinger & Michael Haslauer
& Stefan Winkler & Johannes Briewasser

FH-Salzburg, Studiengang ITSB-M

17. Mai 2014

Übersicht

Intro

Requirements-Engineering

Architektur

Bahn Zahlsystem

- ▶ Entwicklung eines Zahlsystems für Bahnkunden
- ▶ Online-Shop für Speisewagen
- ▶ einfache Bedienung für User (z.B. per App)
- ▶ einfache Administration
- ▶ mehrere Zahlungsmöglichkeiten

Requirements-Engineering

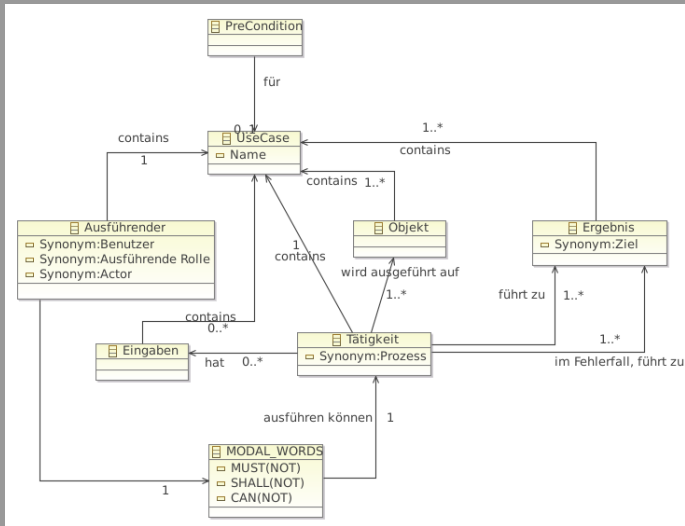
Metamodell

Entwicklung einer eigenen DSL für die Erstellung von Requirements und Use Cases

Gründe für ein Metamodell

- ▶ Standardisierte Form für einen Use Case
- ▶ Definition von Modalverben
- ▶ Sicherstellung aller Metadaten

Metamodell



Use Case Vorlage

Nummer	
Bezeichnung	
Priorität	MUST/SHALL/CAN
Szenario/Tätigkeit	
Ausführender	
PreCondition	
Eingaben	
Ergebnis	
Ergebnis Fehlerfall	
Testszenario	
Objekte	

Use Case Beispiel 1

Nummer	1
Bezeichnung	Waren suchen und auswählen
Priorität	MUST
Szenario/Tätigkeit	Fahrgast ist bereits eingeloggt und kann nun die angebotenen Speisen und Getränke durchsuchen; außerdem kann der Fahrgast nach speziellen Speisen suchen (Suchmaske); neben jedem Produkt kann die Anzahl der Produkte ausgewählt werden (Standard = 0); danach kann der Fahrgast mit einem Klick das Produkt in den Warenkorb legen
Ausführender	Fahrgast (m/w)
PreCondition	erfolgreich eingelogged
Eingaben	Produktsuche; Produktanzahl; Klick auf 'zu Warenkorb hinzufügen'
Ergebnis	Warenkorb befüllt
Ergebnis Fehlerfall	Produkt nicht zum Warenkorb hinzugefügt → Ausgabe Fehlermeldung
Testscenario	Suchen und Produkte auswählen → Produkte müssen im Warenkorb liegen
Objekte	Produkte

Use Case Beispiel 2

Nummer	2
Bezeichnung	Kunde will seinen Einkauf bezahlen
Priorität	MUST
Szenario/Tätigkeit	Fahrgast wählt Bezahlungsmöglichkeit aus; wenn die Bezahlungsmöglichkeit ausgewählt wurde, wird die Rechnung über die gewählte Bezahlungsmöglichkeit beglichen und die Bestellung wird abgeschickt
Ausführender	Fahrgast (m/w)
PreCondition	erfolgreich eingelogged; Warenkorb befüllt
Eingaben	Bezahlungsmöglichkeit (Kreditkarte, Barzahlung) und Bezahlung bestätigen
Ergebnis	Bestellung wird abgeschickt; Betrag wird vom jeweiligen Medium abgebogen (außer bei Barzahlung); Erfolgsmeldung wird für den Fahrgast angezeigt; Bestellung wird geliefert
Ergebnis Fehlerfall	Bestellung wird nicht abgeschickt; zur Information wird eine Meldung für den Fahrgast angezeigt
Testscenario	Testprodukte mit den unterschiedlichen Zahlungsmöglichkeiten kaufen
Objekte	Bestellung

Softwarearchitektur

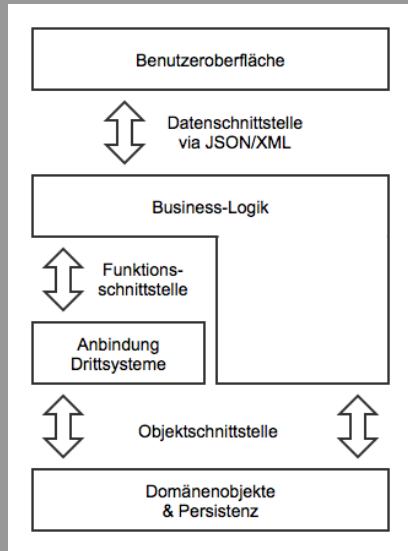
- ▶ Auswahl des Architekturmodells -> Schichtenmodell

Gründe für das Schichtenmodell

- ▶ Trennung von Aufgabenbereichen
- ▶ separate Entwicklung jeder Schicht -> Entwicklerteams
- ▶ lose Kopplung der Schichten
- ▶ keine zyklischen Abhängigkeiten

Schichtenmodell

- ▶ Domänenobjekte und Persistenz
- ▶ Business-Logik
- ▶ Benutzeroberfläche
- ▶ Anbindung von Drittsystemen



Alternative

- ▶ Alternatives Architekturmodell

Micro-Service-Architektur

- ▶ Aufteilung der einzelnen Komponenten als Service
- ▶ separate Entwicklung jedes Service
- ▶ lose Kopplung der Services
- ▶ jeweils eigener Technology-Stack
- ▶ Resilience Service Design -> Availability

Mehr Infos: plainoldobjects.com oder @crichardson