
Objektorientierte Modellierung mit der Unified Modeling Language

Michael Neuhold MSc.

Anwendungsfallbeschreibung

Inhalt

- Einführung
- Struktur
- Fazit und Zusammenfassung

Sind Anwendungsfalldiagramme ausreichend?

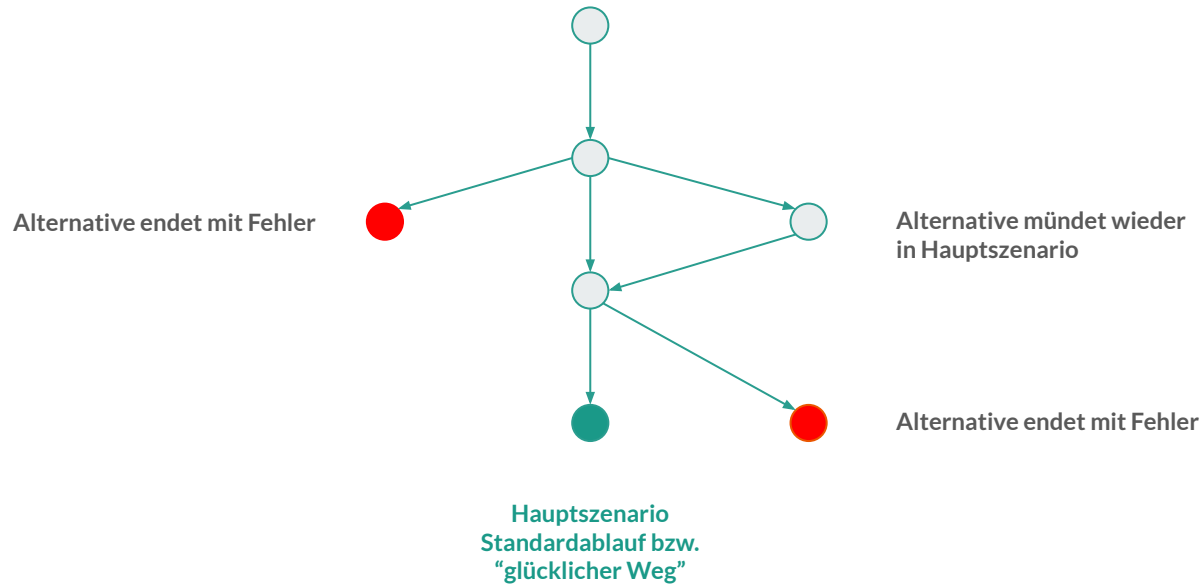
- Anwendungsfalldiagramme bieten einen guten Überblick über die Funktionen eines Systems beschreiben selbst jedoch kein Verhalten und keine Abläufe!
- Anwendungsfalldiagramme spielen nur in Frühphase der Anforderungserhebung eine Rolle!
- Anwendungsfälle sind ausführliche Textdokumente, die zusammenhängende “Geschichten” erzählen, wie sich das System im Betrieb verhalten wird!
- Anwendungsfall-Erhebung bedeutet, Text zu schreiben!

Anwendungsfallbeschreibung

Textuell aber in strukturierter Form!

Anwendungsfallbeschreibung

- Ein Anwendungsfall ist eine Abfolge von erfolgreichen und nicht erfolgreichen Aktionen



Anwendungsfallbeschreibung

- **Hauptszenario** (Standardablauf bzw. “glücklicher Weg”)
 - ein typisches (erfolgreiches) Szenario eines Anwendungsfalls, ohne Verzweigungen bzw. alternative Abläufe
 - besteht aus einzelnen Schritten/Aktionen wie bspw.
 - Interaktionen zwischen Akteuren
 - Validierungen (normalerweise durch das System)
 - Zustandsänderungen durch das System (zB speichern, ändern)
 - Aufbau anderer Anwendungsfälle
- **Alternativ- und Fehlerabläufe** (Erweiterungen)
 - Konkretisierungen sowie Erweiterungen des Standardablaufs
 - ein derartiger Ablauf kann sich auf mehrere Schritte im Standardablauf beziehen
 - besteht meist aus einer Bedingung und Aktionen
 - nach Ausführung Rückkehr in Standardablauf
 - oftmals umfangreicher als der Standardablauf

Struktur

Use-Case-Name/Nummer/Version	Objekt + Verb / UC {number} / Version
Kurzbeschreibung	Kurze, abstrakte Beschreibung des Ablaufs (ca. zwei bis drei Sätze)
Primäre-Akteure (mit Ziel)	Nutzt einen Dienst des Systems, um ein Ziel zu erfüllen
Sekundär-Akteure (mit Ziel)	Andere am Use-Case beteiligte (unterstützende) Akteure und Systeme, die vom System während der Abarbeitung des Use-Cases benötigt werden
Vorbedingungen	Bedingungen, die erfüllt sein müssen, damit der Use-Case ausgeführt werden kann
Ergebnis bei Erfolg	Der Zustand des Systems nach erfolgreicher Beendigung des Use-Cases
Ergebnis bei Fehler	Der Zustand des Systems, wenn der Use-Case nicht erfolgreich beendet wurde
Hauptszenario (Standardablauf)	Das Szenario für einen typischen, erfolgreichen Durchlauf durch den Use-Case
Alternativ- und Fehl Abläufe	Alternative Abläufe zum Hauptszenario für Erfolg oder Misserfolg
Spezielle Anforderungen	Zum System gehörige, nicht funktionale Anforderungen
Weitere Informationen	Beeinflusst die Projektplanung, die Projektaufwandsschätzung, das Testen, etc. Typische Informationen sind: Priorität, Ausführungshäufigkeit, Komplexität, etc.
Status	Status beschreibt, ob der Use-Case ein Entwurf, bereit zum Review oder abgenommen ist
Offene Punkte	Offene Punkte/Annahmen, die den Use-Case betreffen

Anwendungsfallbeschreibung (Beispiel)

Termin erfassen | UC1 | v1

Fazit

- Wie viel Anwendungsfall-Modellierung braucht man?
 - Anwendungsfälle sind ein Vertrag zwischen Kunden und Entwicklerteam
 - Beide müssen diesem guten Gewissens zustimmen können, auch wenn sich Details später ändern können.
- Wie viele Anwendungsfälle sollte man haben?
 - Hängt vom Projekt ab
 - In der ersten Iteration ca. 20 - 30% detaillierter beschrieben
- Wie lange soll ein Anwendungsfall sein?
 - Hängt vom Anwendungsfall ab, 5-12 Schritte bzw. 0.5-2 Seiten
- Was ist das beste Format
 - Es gibt nicht das beste Format (aber **strukturiert!**)

Anwendungsfälle sind eine gute Basis...

- für die **Kommunikation** zwischen Kunden und Entwicklern
 - um zu verstehen, was das Problem des Kunden ist
 - um das Problem spezifizieren und somit entwickeln zu können
- **Zum identifizieren von Objekten, deren Funktionalität, Interaktion und Schnittstellen**
- für die **Aufwandsschätzung**
- für die Planung der Entwicklung
 - Aufteilung in mehreren **Releases/Iterationen**
- für die Definition der funktionalen Testfälle und Akzeptanztests
- für die Erstellung von Dokumentation für den Endanwender
- ...

Aufwandsschätzung auf Basis von Anwendungsfällen

- Erstelle eine List der Anwendungsfälle
- Bewerte jeden Anwendungsfall
- Gewichte die Anwendungsfälle mit Aufwandsfaktor -> eg. Arbeitstage (PD)

Definition von Testfällen

- Ableitung von Testfällen aus den Anwendungsfällen
 - 1 Testfall für Hauptszenario als “Schönwetterfall”
 - 1 Testfall pro Pfad, der sich aus Alternativen ergibt, die wieder zum Hauptszenario zurückführen oder mit Erfolg enden.
 - 1 Testfall für alle Fehler, bei denen Alternativen mit Fehler enden
- Diese Testfälle sind funktionale Black-Box-Tests, so wie auch der Benutzer das System verwenden würde.
 - ein und der Selbe Akteur pro Testfall
- Man kann Anwendungsfälle auch zu Szenarien kombinieren.
 - Dadurch erreicht man End-to-End Abläufe
 - es können auch mehrer Akteuren am gesamten Szenario beteiligt sein

Andere Möglichkeiten um Anforderungen zu beschreiben

- User Stories (Extreme Programming)
- Product Backlog (Scrum)
- Features (Feature Driven Development)
- Pflichtenheft

Use Case vs. User Story

- Gemeinsamkeiten
 - Beschreiben Anforderungen an ein System aus Nutzersicht
 - User Stories definieren wer (Rolle), was (Ziel/Wunsch) und warum (Nutzer)
 - Use Cases definieren Akteure, Hauptszenarien und Ergebnis bei Erfolg und vieles mehr
- Unterschiede
 - Use Cases
 - Detaillierte und vollständige Spezifikation von Anforderungen
 - Gesamtüberblick über das System mittels UML-Anwendungsfalldiagramm
 - dienen als Anforderungsdokumentation
 - User Stories
 - Kurz gehaltene (auch unvollständige) Spezifikation von Anforderungen
 - Dienen als Planungsinstrument (erleichtern die Release- und Iterationsplanung)

Wie behandelt man Daten

- Halte die Daten abstrakt
 - Agent gibt einige Kundeninformationen ein
- Expandiere die Daten im Abschnitt “Weitere Informationen”
 - “Kundeninformation: Vorname, Zuname, Adresse,...”
 - Übertrage diese Information später eventuell in ein anderes Dokument, damit sie von überall aus referenziert werden können.
- Warum funktioniert das?
 - Der Fokus liegt auf der Verwendung des Systems.
 - Die Datenbeschreibungen stehen beieinander.
 - Die Datenbeschreibungen können leicht innerhalb eines Anwendungsfalles bzw. über Anwendungsfälle hinweg verwendet werden.
- Anmerkungen:
 - Es geht dabei um Interaktion, nicht um den Datensatz, das Objekt oder die Kundenidentität
 - Die Definitionen können zentral gehalten werden, auf die man von den Anwendungsfällen verweist

Use Cases passen nicht ...

- ... wenn es hauptsächlich um Zustände geht
 - Call-Center-Telefonie
 - Der Agent hat einen Zustand, der Anrufer,...
 - Am besten durch endliche Automaten beschreiben
 - Garantierten Vollständigkeit
 - Use Cases können trotzdem bei der Diskussion mit Endanwendern helfen
- ... für Prozess, die automatisiert sind
 - Am besten durch Aktivitätsdiagramm beschrieben
- ... für Anpassungen von Standardsoftware an eigenen Bedürfnisse (zb. SAP Customizing)
 - weil es hier keine neu zu entwickelnde Funktionalität gibt, sondern nur ein Abbild von eigenen Geschäftsprozessen auf bestehende Transaktionen
- ... wenn man nicht einmal eine Vision hat, was das System können sollte, was es umfassen soll, wozu es gut sein soll, wer es verwenden wird?