

Softwaretechnik

Modellierung mit Use-Case UML-Diagrammen

Prof. Dr. Bodo Kraft

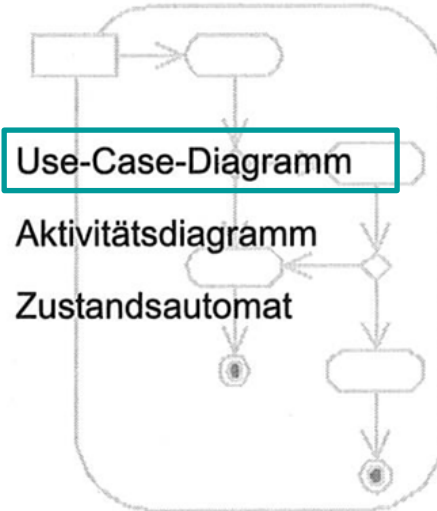
Übersicht UML-Diagramme

Diagramme der UML 2

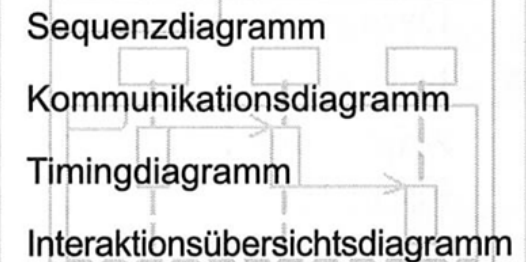
Strukturdiagramme



Verhaltensdiagramme



Interaktionsdiagramme

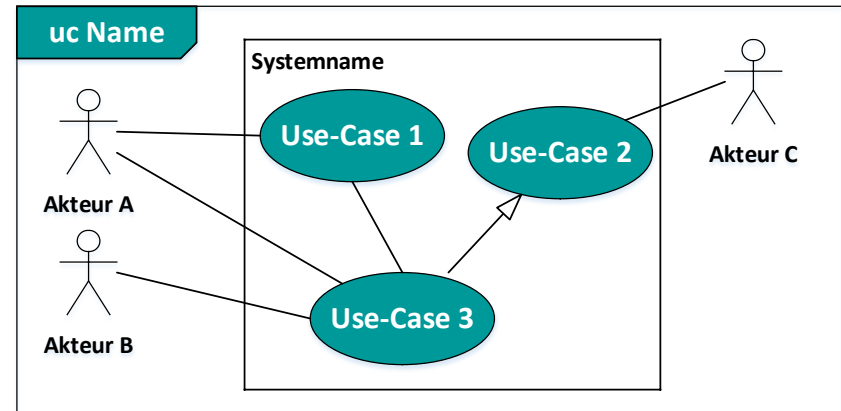


Quelle: UML 2 glasklar, Chris Rupp

Motivation

Use-Case-Diagramme

- Use-Case-Darstellungen existieren in verschiedenen Formen.
- Wir betrachten hier:
 - **Use-Case-Diagramme** (nach UML 2.x)
 - **Textuelle Beschreibungen** (bei uns Tabellen)



UC-Nummer, UC-Name	
Kurzbeschreibung	ein kurzer erklärender (strukturierter) Satz zur Übersicht
Akteure	Personen (Rollen) oder externe Systeme, die aktiv mit dem System interagieren oder einen Nutzen von dem Anwendungsfall haben. Ein Anwendungsfall kann mit mehreren Akteuren verbunden sein.
Kategorie	muss, soll, oder kann der Anwendungsfall realisiert werden?
Auslöser	ein Akteur oder eine Funktion [...] die den Ablauf startet.
Vorbedingung	eine Bedingung, die erfüllt sein muss, damit der Ablauf gestartet wird – Bsp.: Eine Kogge kann erst in See stechen, nachdem sie beladen wurde
Eingabe/Ausgabe	für den Ablauf benötigte Informationen / Ergebnis des Ablaufs
Nachbedingung	Eine Bedingung, die erfüllt sein muss, um den Anwendungsfall zu beenden.
Ablauf	beschreibt den Standardablauf – keine Sonderfälle

Use-Case-Diagramme:

- Ermöglichen **einfachen Einstieg** in die Analyse
- Zeigen das **externe Verhalten eines Systems** aus Sicht der Nutzer
- Zeigen wie **Nutzer mit dem System interagieren**, um ein Ziel zu erreichen
- (Use-Case \triangleq Anwendungsfall)

Allgemeines

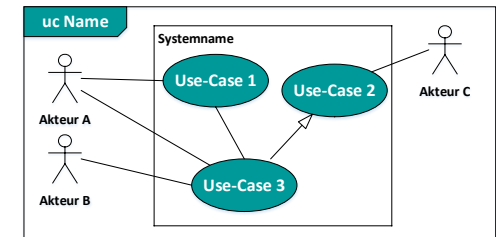
Use-Case-Diagramme

Use-Case Diagramme

- Liefern eine Antwort auf die zentrale Frage:

*Was soll mein System für seine Umwelt leisten
(Stakeholder, Nachbarsysteme)?*

- Hilft, nicht zu sehr ins Detail abzurutschen
- *Betonung auf WAS, nicht WIE*
- *Beschreibung des funktionalen Verhaltens*
 - *Funktionsumfang*



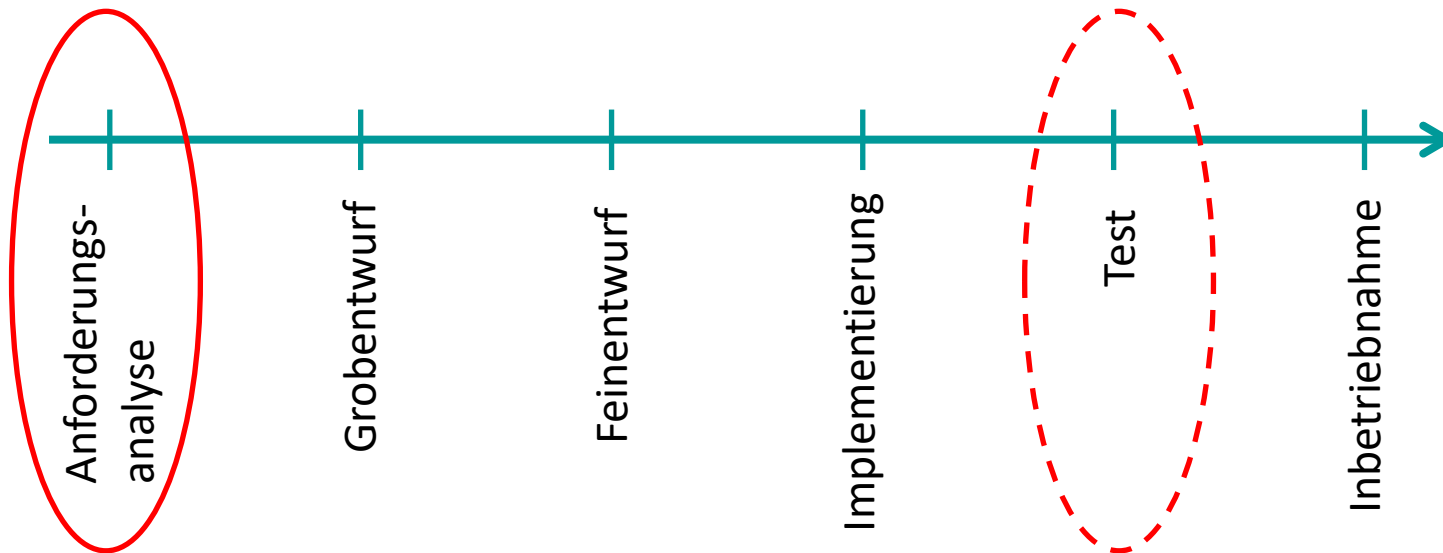
- Jeder Anwendungsfall zeigt die **Interaktion** zwischen dem *System* und den Benutzern (*Akteuren*) für ein bestimmtes Ziel.
- UC-Diagramme präsentieren die Außensicht auf das System
- Sind geeignet zur Kontextabgrenzung
- Hohes Abstraktionsniveau, einfache/schlichte Notation

Zeitliche Einordnung in SW-Lifecycle

Use-Case-Diagramme

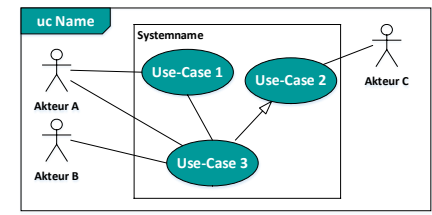
Bei welchen Schritten des Software-Lifecycle kann ich Use-Cases brauchen?

- Use-Case Diagramm werden hauptsächlich bei der Anforderungsanalyse verwendet.
- Die festgelegten Use-Cases können später über Tests abgehandelt werden, um die korrekte Funktionsweise des Programmes nachzuweisen



Grundlagen/Basiselemente (I)

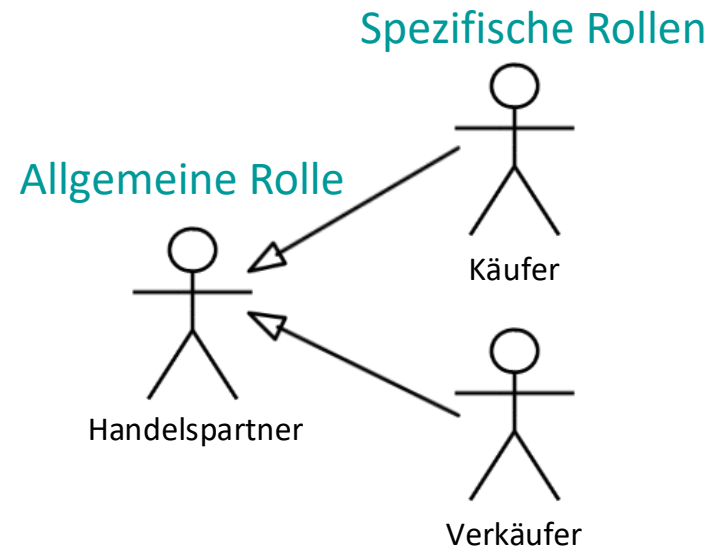
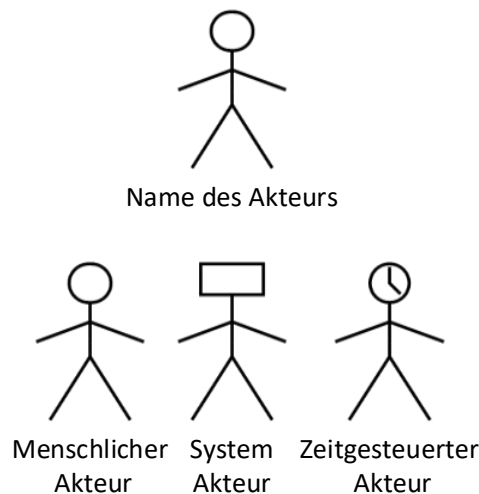
Use-Case-Diagramme



Die Notationselemente eines UC-Diagrammes kann man in vier Gruppen aufteilen.

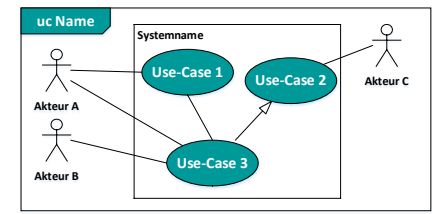
Gruppe1: **Akteure**

- Meist beschrieben durch Personen oder Rollen
- Sind die Initiatoren, Akteure stoßen Use-Cases an
- Symbol: stilisierte **Strichmännchen**
 - Viele weitere Symbole möglich ...



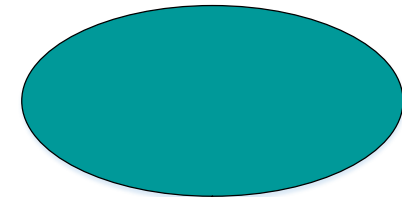
Grundlagen/Basiselemente (II)

Use-Case-Diagramme



Gruppe2: Use-Cases (oder Anwendungsfälle)

- Namenskonvention:
 - Besteht aus Substantiv + Verb
 - Bsp.: „Handel betreiben“
- Symbol zur Darstellung:
 - **Oval** oder **Ellipse**
- Kommt in 3 Varianten vor:
 - **Variante1(Standard)**: Ellipse beinhaltet den Namen des Use-Case
 - **Variante2**: Bei langen Namen wird der Text häufig ausgelagert
 - **Variante3(selten)**: meist vorgegeben durch SW-Tools:
Rechteck beinhaltet den Namen -
eine kleine Ellipse in der Ecke rechts oben zeigt den Use-Case an

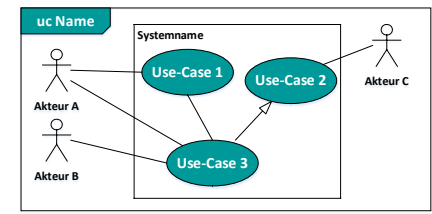


Use-Case Name



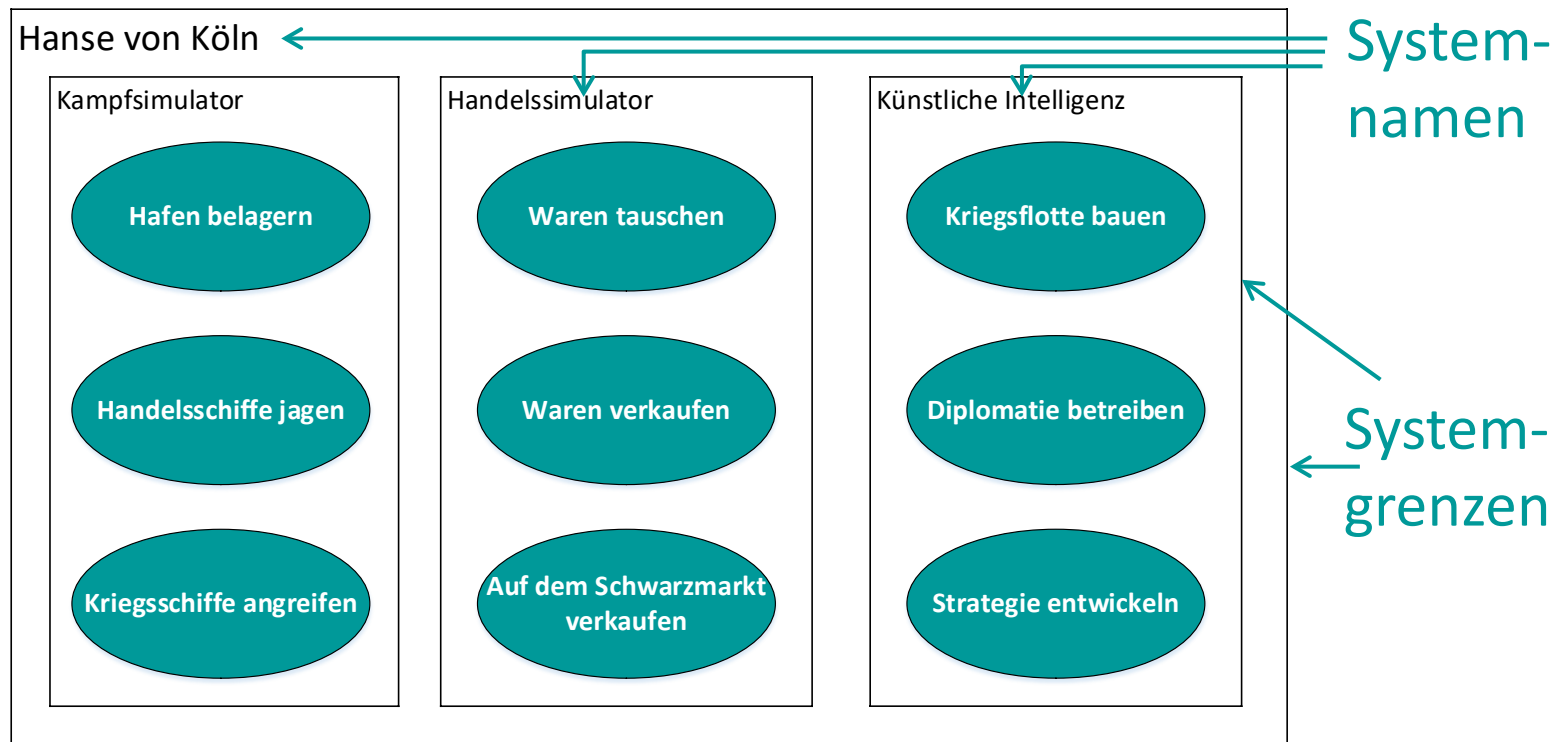
Grundlagen/Basiselemente (III)

Use-Case-Diagramme



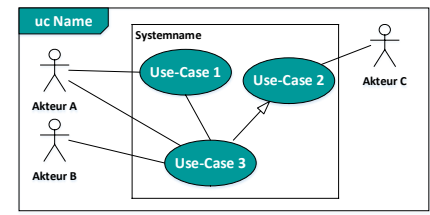
Gruppe3: Systeme und Grenzen

- Fachliche Sicht und Einordnung des Gesamtsystems
- Zur Abgrenzung von **fachlichen** Teilen, Aufgaben und/oder Verantwortungsbereichen
- Keine technische Analyse!



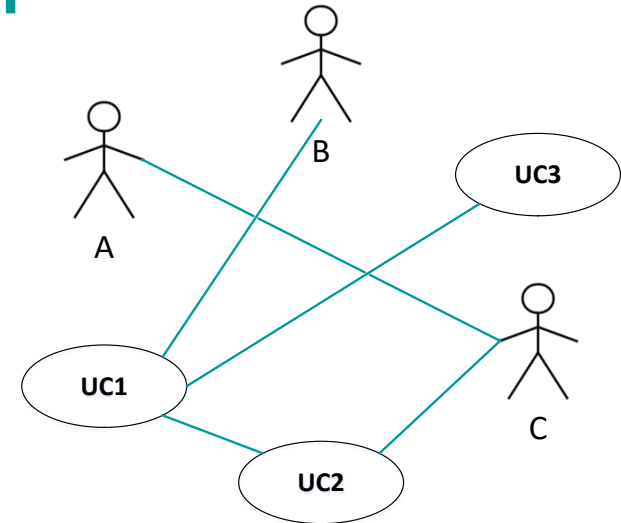
Grundlagen/Basiselemente (IV)

Use-Case-Diagramme



Gruppe4: **Kanten bzw. Assoziationen**

- Modellieren Interaktionen zwischen
 - Akteur und Akteur
 - Akteur und Use-Case
 - Use-Case und Use-Case
- Darstellung erfolgt über Kanten oder Linien im U.C.-Diagramm
- Ggfs. Anreicherung über zusätzliche Annotationen
- Antwort auf die Frage:
Wer steht mit wem in welcher Beziehung?

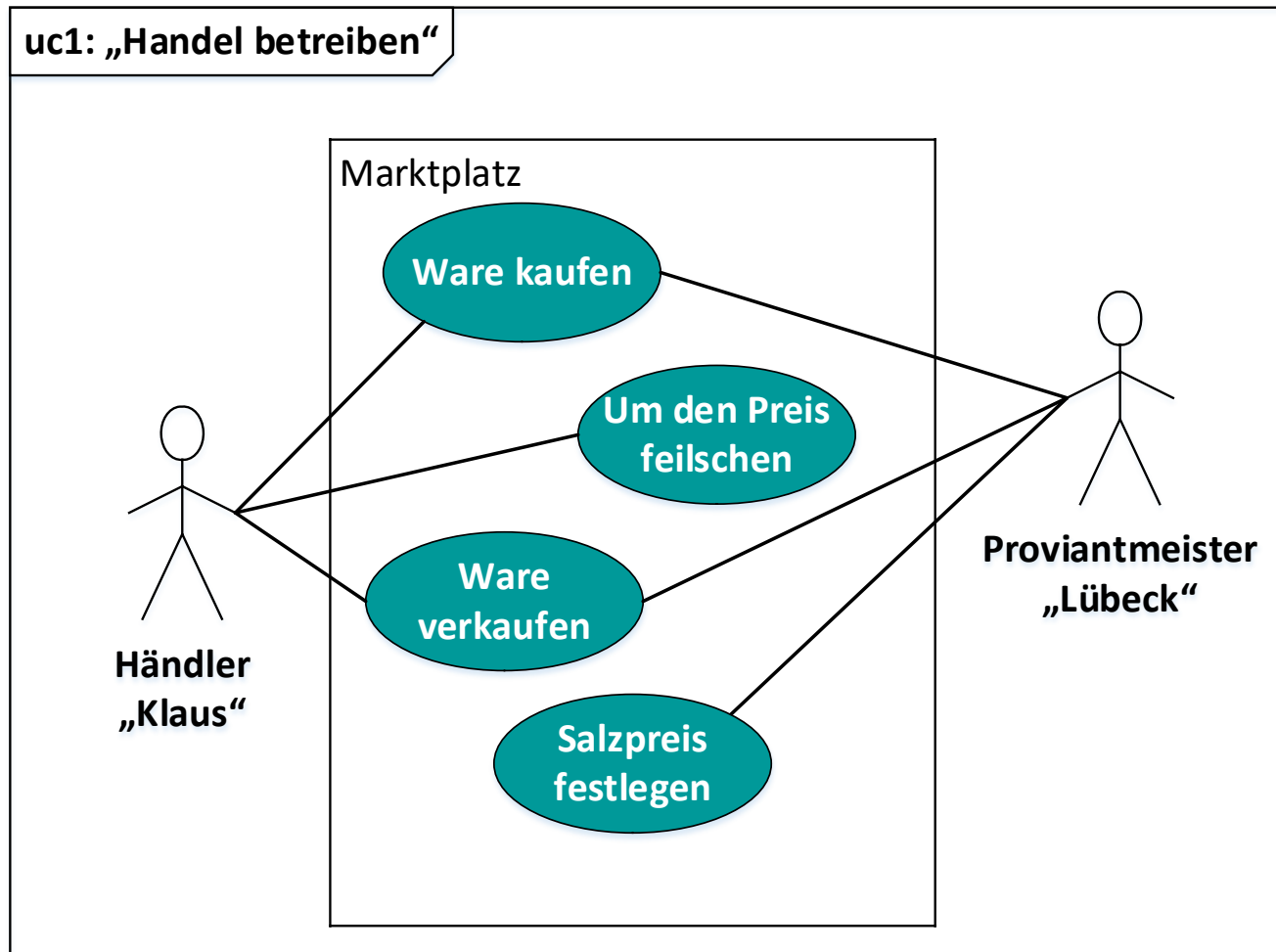


Ein Beispiel dazu →

Grundlagen/Basiselemente (V)

Use-Case-Diagramme

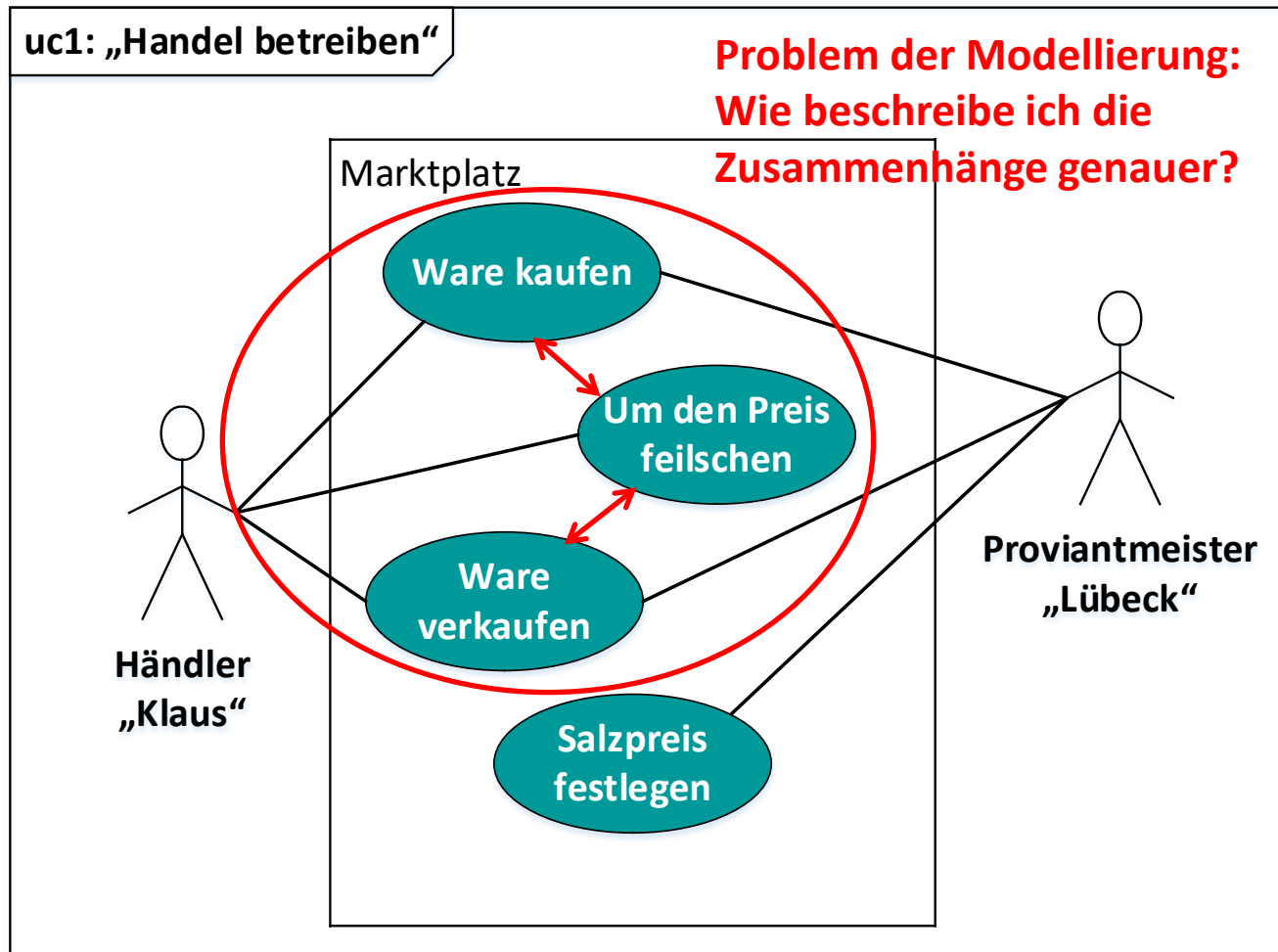
Die Standardmodellierung eines Systems



Grundlagen/Basiselemente (V)

Use-Case-Diagramme

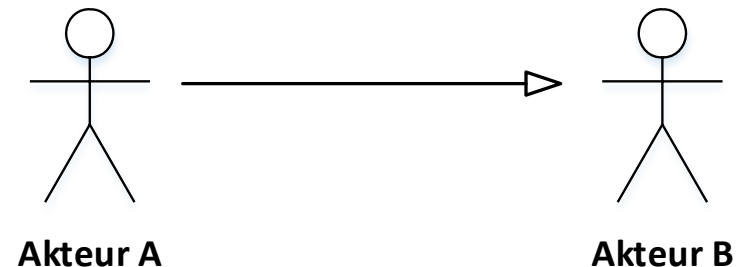
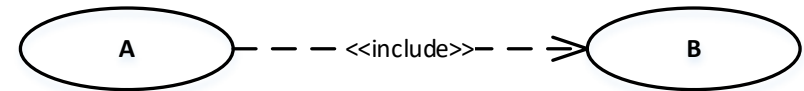
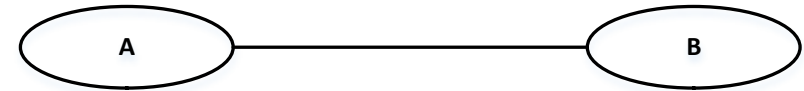
Die Standardmodellierung eines Systems



Assoziationen/Beziehungstypen

Use-Case-Diagramme

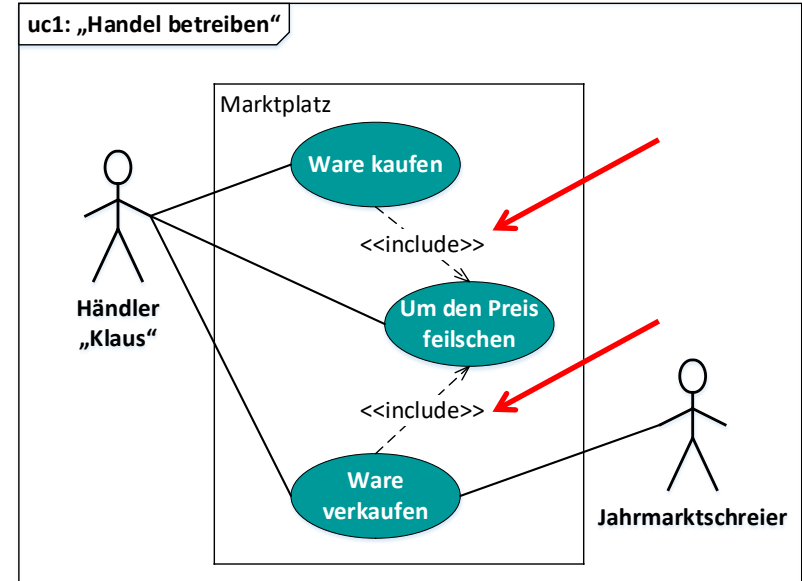
Assoziationstyp	Bedeutung
Binäre Assoziation (ungerichtet)	Richtung nicht eindeutig , i.d.R. ausgehend von einem der beteiligten Akteure
Import/Include-Beziehung	A umfasst das Verhalten von B vollständig. D.h. B ist eine Teilfunktion von A. A beinhaltet aber noch weitere Teile. (klassische Anwendung um Redundanz zu vermeiden: Mehrere Use-Cases führen denselben Teil aus, dann wird er ausgelagert und gemeinsam genutzt)
Erweiterungs-Beziehung	A erweitert B unter einer speziellen Voraussetzung
Generalisierungs-Beziehung	„ is-A “ Beziehung. Interessant bei abstrakten Akteuren und/oder Teilsystemen



Include-Beziehung

Use-Case-Diagramme

- Feilschen gehört in Lübeck zur Tradition
- **Kein Ankauf/Verkauf findet ohne vorherige Preisverhandlung statt**
- „Um den Preis feilschen“ ist Teil der übergeordneten Use-Cases
- „Um den Preis feilschen“ kann auch eigenständig ausgeführt werden (kein Kompromiss beider Handelspartner)
- „Um den Preis feilschen“ wird gemeinsam genutzt, um redundanzfrei zu modellieren
- „Um den Preis feilschen“ wird zwingend ausgeführt bei Aufruf des übergeordneten Use-Case (**ist nicht optional!!**)

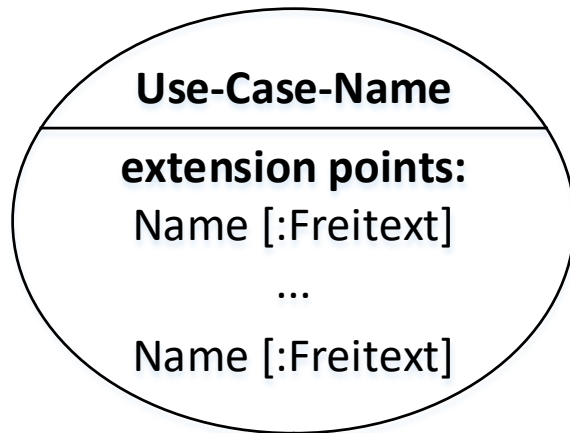


Extension Points

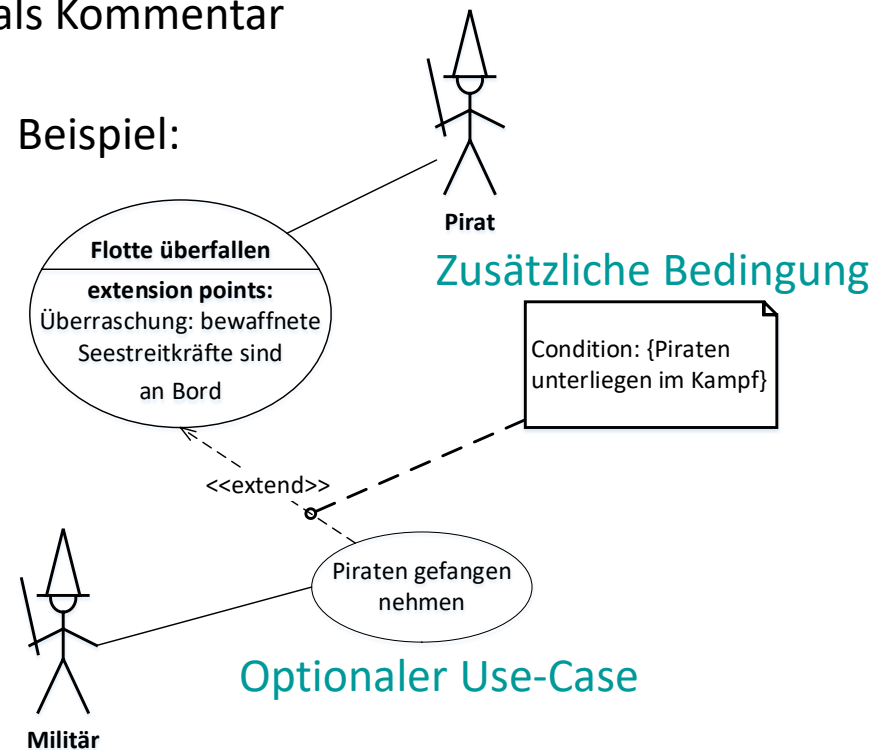
Use-Case-Diagramme

- Use-Cases können um **optionale** oder **bedingte Use-Cases** erweitert werden
- Erweiterung durch **Extension Points**
- Ein Extension Point definiert den Eintrittspunkt formal
- Umgangssprachliche Umschreibungen können zusätzlich im **Freitextfeld** angegeben werden.
- **Bedingungen** können angegeben werden als Kommentar

Syntax/Symbolik:

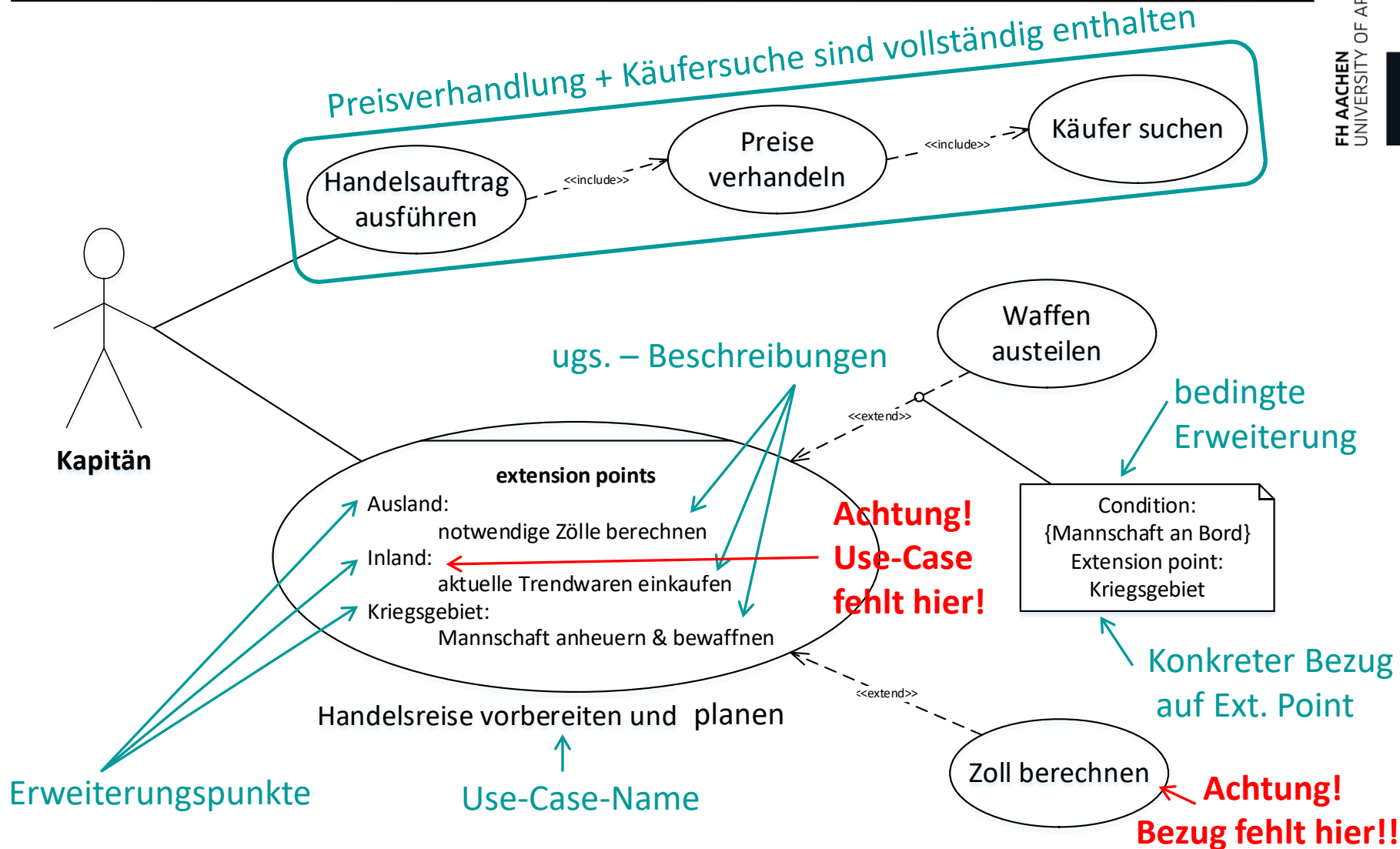


Beispiel:



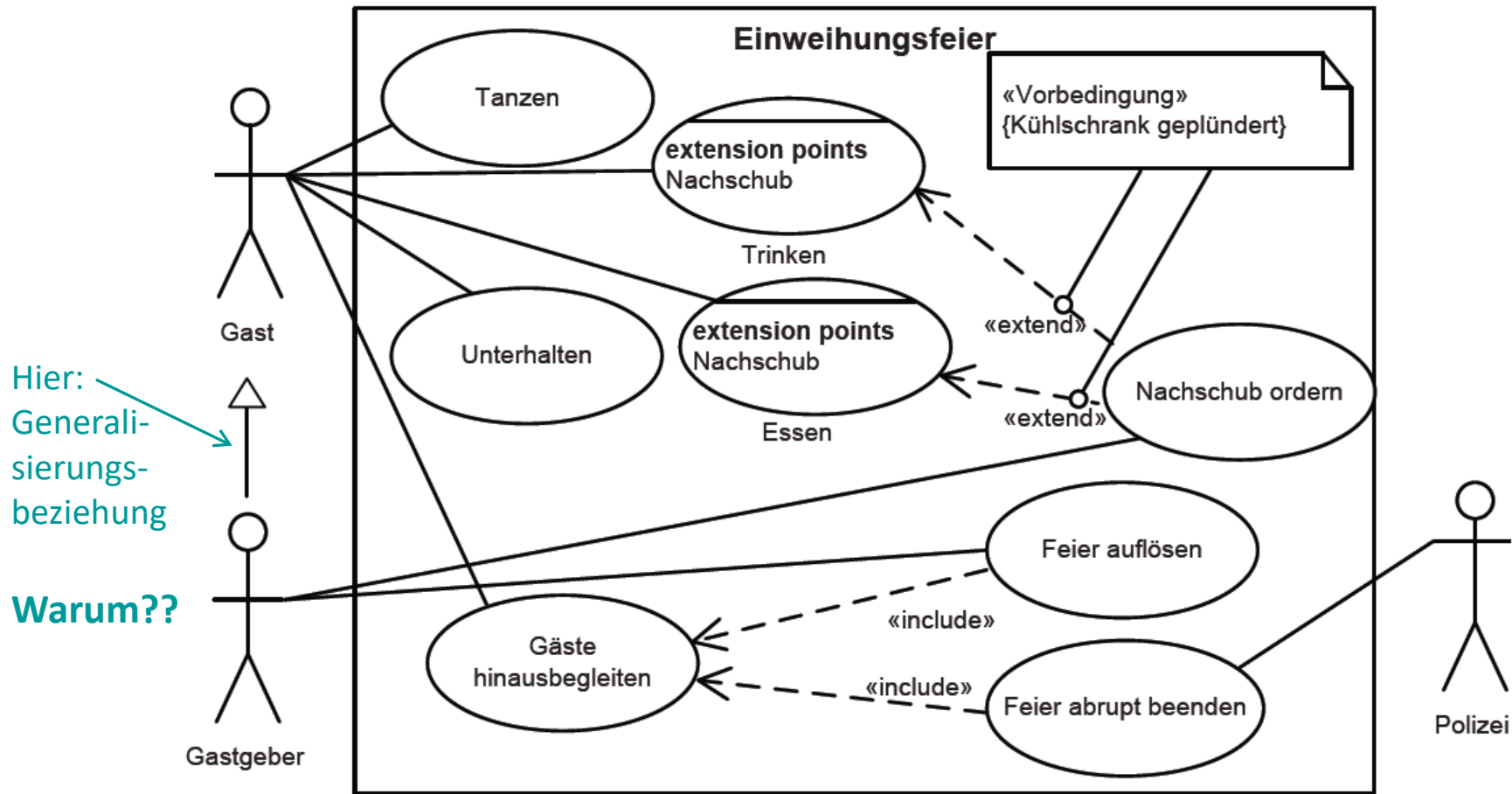
Include & Extend Beziehungen

Use-Case-Diagramme



Ein angewandtes Beispiel aus dem Alltag

Use-Case-Diagramme



Erstellen Sie ein Use Case-Diagramm zu folgendem Szenario:

- Wir befinden uns im Büro unseres Handelssimulators. Hier hat der Spieler die Möglichkeit, die vorhandene Lagerkapazität einzusehen, den Warenbestand zu checken und einen neuen Speicher zu bauen.
- Einen Sonderfall stellt die Wareneinlagerung dar: Dazu muss das Schiff entladen werden und wenn das Lager voll ist, muss die Lagerkapazität erweitert werden.

Versuchen Sie in diesem Sonderfall die Schlüsselwörter **extend** und **include** zu verwenden.

Anwendungsfälle modellieren

Präsenzaufgabe - Lösung

Präsenzaufgabe

Textuelle Beschreibung

Use-Case-Diagramme

- Neben den UML-Diagrammen werden Use-Cases auch häufig textuell beschrieben, bzw. näher spezifiziert
- Häufig in Form von Tabellen

UC-Nummer, UC-Name	
Kurzbeschreibung	ein kurzer erklärender (strukturierter) Satz zur Übersicht
Akteure	Personen (Rollen) oder externe Systeme, die aktiv mit dem System interagieren oder einen Nutzen von dem Anwendungsfall haben. Ein Anwendungsfall kann mit mehreren Akteuren verbunden sein.
Kategorie	muss, soll, oder kann der Anwendungsfall realisiert werden?
Auslöser	ein Akteur oder eine Funktion [...] die den Ablauf startet.
Vorbedingung	eine Bedingung, die erfüllt sein muss, damit der Ablauf gestartet wird – Bsp.: Eine Kogge kann erst in See stechen, nachdem sie beladen wurde
Eingabe/Ausgabe	für den Ablauf benötigte Informationen / Ergebnis des Ablaufs
Nachbedingung	Eine Bedingung, die erfüllt sein muss, um den Anwendungsfall zu beenden.
Ablauf	beschreibt den Standardablauf – Was soll passieren, wenn der Anwendungsfall gestartet ist? Achtung, hier werden keine Ausnahmen behandelt!

Anwendungsbeispiel: Textuelle Beschreibung Use-Case-Diagramme

UC3: Waren einlagern

Kurzbeschreibung	Im Hafen eintreffende Koggen werden entladen und die Ladung im Kontor gespeichert.
Akteure	Aktueller Spieler
Kategorie	Muss
Auslöser	Eintreffen einer Handelsflotte in Lübeck
Vorbedingung	Koggen im Hafen, Koggen sind beladen, Lager hat ausreichend Kapazitäten
Eingabe/Ausgabe	E: Volle Schiffsladung bzw. Menge an verschiedenen Rohstoffen A: Wareneingang verrechnet und im Kontor gutgeschrieben, Schiffe sind entladen, Koggen liegen vor Ort im Hafen
Nachbedingung	Koggen liegen auftragsbereit im Hafen
Ablauf	<ol style="list-style-type: none">1) Handelsflotte trifft ein2) Rohstoffe werden auf dem Kontor gut geschrieben3) Rohstoffe werden aus der Flotte entfernt4) Flotte wird aufgelöst, die bestehenden Schiffe den im Hafen ankernden Schiffen hinzugefügt.

Anwendungsbeispiel: Textuelle Beschreibung

Vervollständigen Sie die Tabelle zu folgendem Use-Case:

UC4: gegnerische Flotte plündern

Kurzbeschreibung	
Akteure	
Kategorie	
Auslöser	
Vorbedingung	
Eingabe/Ausgabe	
Nachbedingung	
Ablauf	

- Bei Projekten mit enger Kundenbindung (z.B. bei engen Beziehungen zwischen AG und IT-Abteilung bei Inhouse-Projekten) können **Use Cases** (oder Nutzer Stories) **als Anforderungsdokumentation ausreichen**, wenn das Projekt in kleinen Iterationen und der Möglichkeit eines großen Kundeneinflusses entwickelt wird
- Oftmals ist die Beschreibung der **Use Cases** aber **zu ungenau**, gerade bei der Darstellung typischer und alternativer Abläufe stellt sich die **rein sprachliche Beschreibung als recht aufwändig** heraus
- **Reihenfolgen und konkrete Abläufe werden i.d.R. nicht ausreichend modelliert für konkrete Implementierung**
- **UML als graphische Sprache stellt auch für Ablaufbeschreibungen eine grafische Darstellungsmöglichkeit zur Verfügung, nämlich *Aktivitätsdiagramme*.**

Literatur:

- [RS] C. Rupp, SOPHIST GROUP, Requirements- Engineering und – Management, Hanser Fachbuchverlag, 2004
- [OW] B. Oestereich, C. Weiss, C. Schröder, T. Weilkiens, A. Lenhard, Objektorientierte Geschäftsprozessmodellierung mit der UML, dpunkt.Verlag, 2003

Vielen Dank!