

OOA, OOD

(Objektorientierte Analyse,
Objektorientiertes Design)

OOA: Objektorientierte Analyse

Lastenheft/Pflichtenheft



Situation/Problemstellung analysieren

Ziel: System von Objekten finden und strukturieren

- 1) Objekte und Klassen finden
- 2) Eigenschaften und Funktionen festlegen
- 3) Welche Objekte haben (welche) Beziehungen untereinander?

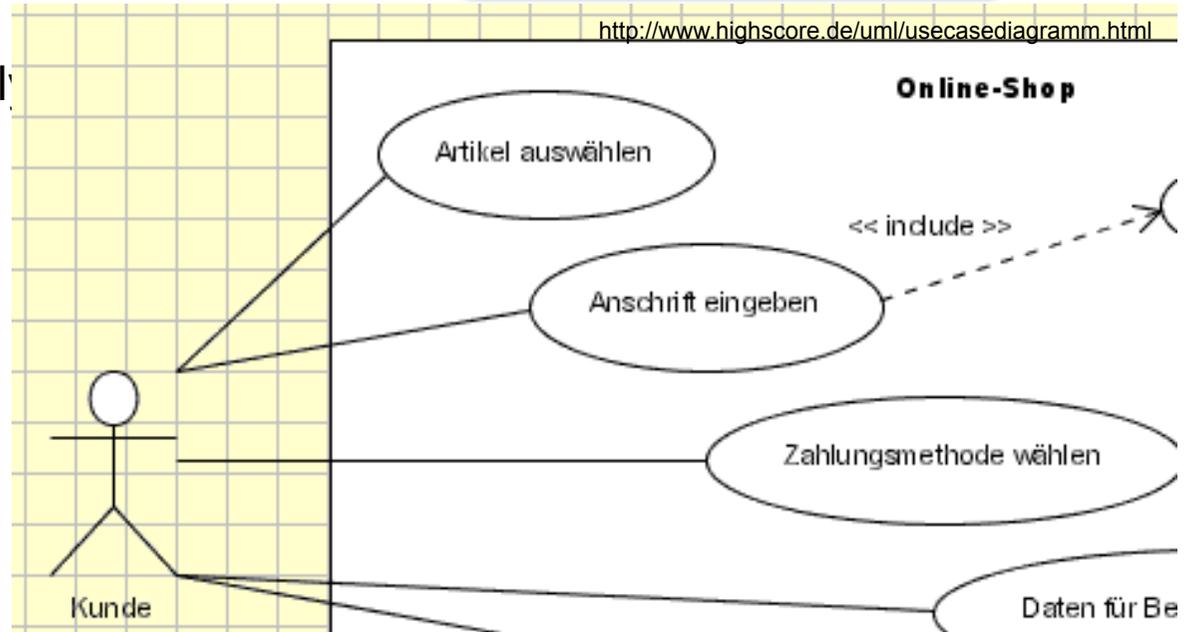
WAS

soll das Programm tun?

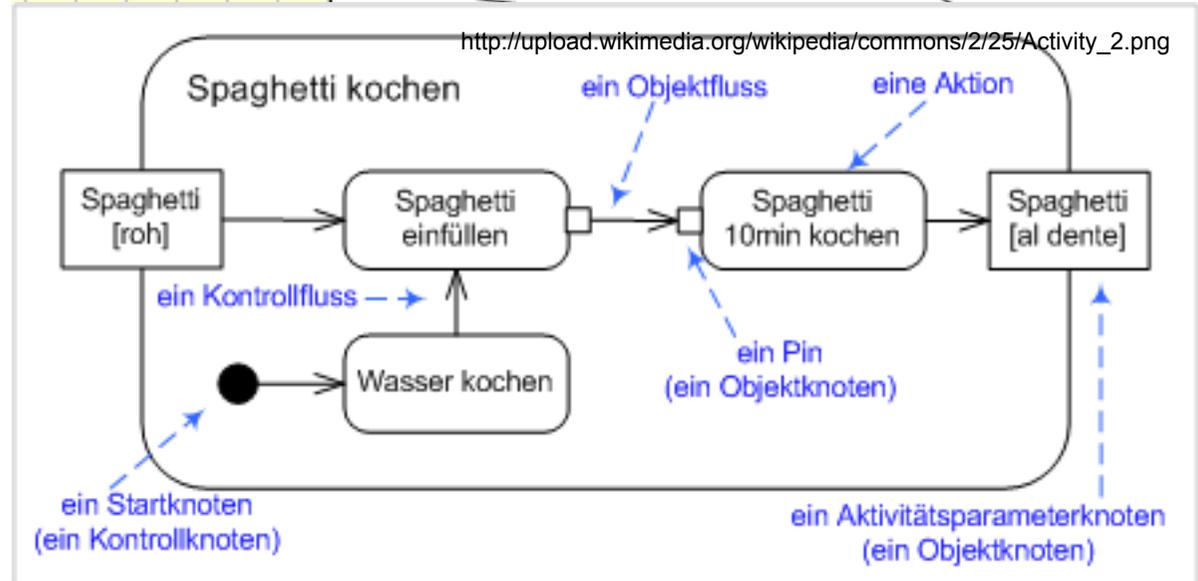
OOA: Objektorientierte Analyse

Lastenheft/Pflichtenheft

Situation/Problemstellung analysieren
Ziel: System von Objekten finden und strukturieren



- Ergebnis z.B.:
- Anwendungsfalldiagramme
 - Aktivitätsdiagramme
 - grobe Klassendiagramme



OOA: Objektorientierte Analyse

Situation/Problemstellung analysieren

Ziel: System von Objekten finden und strukturieren

- 1) Objekte und Klassen finden
- 2) Eigenschaften und Funktionen festlegen
- 3) Welche Objekte haben (welche) Beziehungen untereinander?

WAS
soll das Programm tun?

OOD: Objektorientiertes Design

Konkretisierung der Analyseergebnisse

Ziel: Endgültige Architektur festlegen

- 1) Zusammenspiel Fachklassen / Benutzeroberfläche (z.B. MVC)
- 2) Anbindung Datenhaltung (z.B. Datenbank)
- 3) Beziehung der Objekte untereinander bestimmen
- 4) Anpassung auf Programmiersprache

WIE
soll das Programm es tun?

Ergebnis z.B.:

- Klassendiagramme als Vorstufe der Implementierung

OOA: Objektorientierte Analyse

Situation/Problemstellung analysieren

Ziel: System von Objekten finden und strukturieren

- 1) Objekte und Klassen finden
- 2) Eigenschaften und Funktionen festlegen
- 3) Welche Objekte haben (welche) Beziehungen untereinander?

WAS
soll das Programm tun?

OOD: Objektorientiertes Design

Konkretisierung der Analyseergebnisse

Ziel: Endgültige Architektur festlegen

- 1) Zusammenspiel Fachklassen / Benutzeroberfläche (z.B. MVC)
- 2) Anbindung Datenhaltung (z.B. Datenbank)
- 3) Beziehung der Objekte untereinander bestimmen
- 4) Anpassung auf Programmiersprache

Kenntnisse
der OOP notwendig!

WIE
soll das Programm es tun?

oop: Objektorientierte Programmierung

OOA: Objektorientierte Analyse

Situation/Problemstellung analysieren

Ziel: System von Objekten finden und strukturieren

- 1) Objekte und Klassen finden
- 2) Eigenschaften und Funktionen festlegen
- 3) Welche Objekte haben (welche) Beziehungen untereinander?

WAS
soll das Programm tun?

oft keine scharfe Trennung möglich

OOD: Objektorientiertes Design

Konkretisierung der Analyseergebnisse

Ziel: Endgültige Architektur festlegen

- 1) Zusammenspiel Fachklassen / Benutzeroberfläche (z.B. MVC)
- 2) Anbindung Datenhaltung (z.B. Datenbank)
- 3) Beziehung der Objekte untereinander bestimmen
- 4) Anpassung auf Programmiersprache

WIE
soll das Programm es tun?

OOP: Objektorientierte Programmierung

OOA: Objektorientierte Analyse

Situation/Problemstellung analysieren

Ziel: System von Objekten finden und strukturieren

- 1) Objekte und Klassen finden
- 2) Eigenschaften und Funktionen festlegen
- 3) Welche Objekte haben (welche) Beziehungen untereinander?

WAS
soll das Programm tun?

oft keine scharfe Trennung möglich

OOD: Objektorientiertes Design

Konkretisierung der Analyseergebnisse

Ziel: Endgültige Architektur festlegen

- 1) Zusammenspiel Fachklassen / Benutzeroberfläche (z.B. MVC)
- 2) Anbindung Datenhaltung (z.B. Datenbank)
- 3) Beziehung der Objekte untereinander bestimmen
- 4) Anpassung auf Programmiersprache

WIE
soll das Programm es tun?

OOP: Objektorientierte Programmierung

Diagrammarten

statische Modellierung

= Beschreibung der strukturellen Aspekte
(z.B. Klassendiagramme)

dynamische Modellierung

= Abbildung von zeitlichen Verläufen und Interaktionen
(z.B. Aktivitätsdiagramm, Use-Case-Diagramme)