

# 46116 Herbst 2014

Softwaretechnologie / Datenbanksysteme (nicht vertieft)

Aufgabenstellungen mit Lösungsvorschlägen

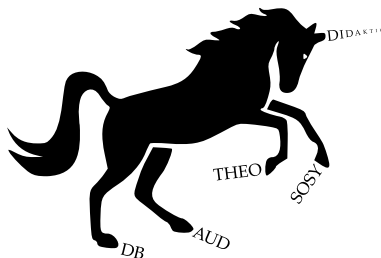


**Die Bschlangaul-Sammlung**

Hermine Bschlangaul and Friends

# Aufgabenübersicht

Thema Nr. 2 . . . . .	3
Teilaufgabe Nr. 1 . . . . .	3
Aufgabe 1 [Multiple-Choice: Allgemeine SWT, Vorgehensmodelle und Requirements] . . . . .	3
Aufgabe 1: Allgemeine SWT, Vorgehensmodelle und Requirements Engineering .	3
Aufgabe 2: Vorgehensmodelle [Vorgehensmodelle] . . . . .	4
Aufgabe 3: UML Diagramme in der Anwendung [Bestellsystem] . . .	7



## Die Bschlangaul-Sammlung

Hermine Bschlangaul and Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.



Diese Materialsammlung unterliegt den Bestimmungen der Creative Commons Namensnennung-Nicht kommerziell-Share Alike 4.0 International-Lizenz.

# Thema Nr. 2

## Teilaufgabe Nr. 1

Aufgabe 1 [Multiple-Choice: Allgemeine SWT, Vorgehensmodelle und Requirements]

### Aufgabe 1: Allgemeine SWT, Vorgehensmodelle und Requirements Engineering

Kreuzen Sie für die folgenden Multiple-Choice-Fragen genau die richtigen Antworten deutlich an. Es kann mehr als eine Antwort richtig sein.

Jedes korrekt gesetzte oder korrekt nicht gesetzte Kreuz wird mit 1 Punkt gewertet. Jedes falsch gesetzte oder falsch nicht gesetzte Kreuz wird mit -1 Punkt gewertet. Eine Frage kann entwertet werden, dann wird sie nicht in der Korrektur berücksichtigt. Einzelne Antworten können nicht entwertet werden. Entwerten Sie eine Frage wie folgt

Die gesamte Aufgabe wird nicht mit weniger als 0 Punkten gewertet.

(a) Welche Aussage ist wahr?

- ☐ Je früher ein Fehler entdeckt wird, umso teurer ist seine Korrektur.
- ☐ Je später ein Fehler entdeckt wird, umso teurer ist seine Korrektur.
- ☐ Der Zeitpunkt der Entdeckung hat keinen Einfluss auf die Kosten.

Lösungsvorschlag

2 ist richtig: Je später der Fehler entdeckt wird, desto mehr wurde er schon in das Projekt „eingearbeitet“, daher dauert das Beseitigen des Fehlers länger und das kostet mehr Geld.

(b) Mit welcher Methodik können Funktionen spezifiziert werden?

- ☐ Als Funktionsvereinbarung in einer Programmiersprache
- ☐ Mit den Vor- und Nachbedingungen von Kontrakten
- ☐ Als Zustandsautomaten

Lösungsvorschlag

2 und 3 ist richtig: Die Spezifikation soll unabhängig von einer Programmiersprache sein.

(c) Welche Vorgehensmodelle sind für Projekte mit häufigen Änderungen geeignet?

- ☐ Extreme Programming (XP)
- ☐ Das V-Modell 97
- ☐ Scrum

Lösungsvorschlag

1 und 3 ist richtig. Das V-Modell ist ein starres Vorgehensmodell, bei dem alle Anforderungen zu Beginn vorhanden sein müssen.

(d) Welche der folgenden Aussagen ist korrekt?

- ☐ Mittels Prototyping versucht man die Anzahl an nötigen Unit-Tests zu reduzieren.
- ☐ Ein Ziel von Prototyping ist die Erhöhung der Qualität während der Anforderungsanalyse.
- ☐ Mit Prototyping versucht man sehr früh Feedback von Stakeholdern zu erhalten.

Lösungsvorschlag

2 und 3 ist richtig: Prototypen müssen auch getestet werden. Es kann nicht an Tests gespart werden. Durch das häufige Feedback des Kunden / der Stakeholder können die Anforderungen immer genauer und klarer erfasst werden.

(e) Welche der folgenden Aussagen ist korrekt?

- ☐ Bei der Architektur sollten funktionale und nicht-funktionale Anforderungen beachtet werden.
- ☐ Bei der Architektur sollten nur funktionale Anforderungen beachtet werden.
- ☐ Bei der Architektur sollten nur nicht-funktionale Anforderungen beachtet werden.
- ☐ Bei der Architektur sollte auf die mögliche Änderungen von Komponenten geachtet werden.

Lösungsvorschlag

1 und 4 ist richtig: Mögliche Änderungen werden durch klar definierte Schnittstellen und wenig Kopplung der Komponenten erleichtert. (Kopplung handelt von Abhängigkeiten zwischen Modulen. Kohäsion handelt von Abhängigkeiten zwischen Funktionen innerhalb eines Moduls.)

## Aufgabe 2: Vorgehensmodelle [Vorgehensmodelle]

Software wird oft in definierten Prozessen entwickelt. Diese nennt man Vorgehensmodelle.

Allgemein

(a) Was sind die Aufgaben eines Vorgehensmodells im Allgemeinen?

- liefert Erfahrungen und bewährte Methoden
- beschreibt die am Projekt beteiligten Rollen
- legt Aufgaben und Aktivitäten fest
- definiert einheitliche Begriffe
- gibt Techniken, Werkzeuge, Richtlinien / Standards an

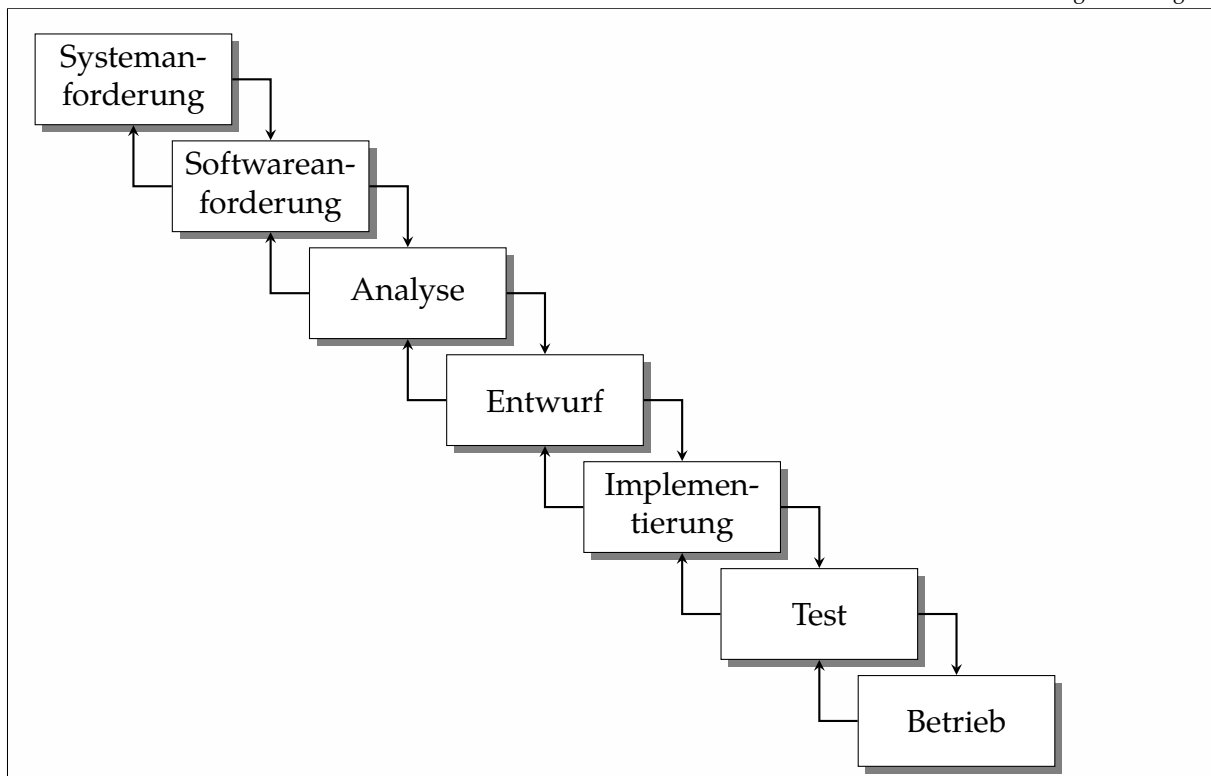
(b) Was sind die wesentlichen Bestandteile eines Vorgehensmodells und in welcher Beziehung stehen diese zueinander?

Bestandteile: Anforderungsanalyse, Modellierung, Implementierung, Test, Auslieferung, Wartung

Die einzelnen Phasen bauen immer aufeinander aus. Je nach Vorgehensmodell können sich Phasen auch wiederholen (Prototyping, Scrum...).

Ein frühes Vorgehensmodell ist das von Dr. Winston Royce 1970 formalisierte Wasserfallmodell.

(a) Geben Sie eine schematische Darstellung des Wasserfallmodells an.



(b) Nennen Sie zwei Probleme des Modells und erläutern Sie diese kurz.

Fehler werden ggf. erst am Ende des Entwicklungsprozesses erkannt, da erst dort das Testen stattfindet. Dadurch kann die Behebung eines Fehlers sehr aufwändig und somit teuer werden.

Der Kunde / Endanwender wird erst nach der Implementierung wieder eingebunden. Das bedeutet, dass er nach der Stellung der Anforderungen keinen Einblick mehr in den Prozess hat und somit auch nicht gegensteuern kann, falls ihm etwas nicht gefällt oder er etwas nicht bedacht hat.

- (c) In welchen Situationen lässt sich das Wasserfallmodell gut einsetzen?

Das Wasserfallmodell ist geeignet, wenn es sich um ein von Anfang an klar definiertes Projekt ohne große Komplexität handelt, bei dem alle Anforderungen, Aufwand und Kosten schon zu Beginn des Projekts feststehen bzw. abgeschätzt werden können.

Barry Boehm erweiterte das Wasserfallmodell 1979 zum so genannten V-Modell.

- (a) Geben Sie eine schematische Darstellung des V-Modells an.  
(b) Nennen Sie zwei Probleme des Modells und erläutern Sie diese kurz.

- Die Nachteile des Wasserfallmodells bestehen weiterhin!
- Nicht für kleine Projekte geeignet, da aufwändige Tests vorgesehen sind, die im Kleinen detailliert meist nicht stattfinden (können).

- (c) Welchen Vorteil hat das V-Modell gegenüber dem Wasserfallmodell?

Für jedes Dokument besteht ein entsprechender Test (Validierung / Verifikation). Dabei kann die Planung der Tests schon vor der eigentlichen Durchführung geschehen, so dass Aktivitäten im Projektteam parallelisiert werden können. So kann zum Beispiel der Tester die Testfälle für den Akzeptanztest (=Test des Systementwurfs) entwickeln, auch wenn noch keine Implementierung existiert.

In neuerer Zeit finden immer häufiger iterative und inkrementelle Vorgehensweisen Anwendung.

- (a) Erklären Sie den Begriff iterative Softwareentwicklung.

Iterativ heißt, dass der Entwicklungsprozess mehrfach wiederholt wird: statt den „Wasserfall“ einmal zu durchlaufen, werden „kleine Wasserfälle“ hintereinander gesetzt.

- (b) Erklären Sie den Begriff inkrementelle Softwareentwicklung und grenzen Sie ihn von iterativer Softwareentwicklung ab.

Lösungsvorschlag

Bei der inkrementellen Entwicklung wird das System Schritt für Schritt fertig gestellt. D. h., dass ein Prototyp immer etwas mehr kann als der Prototyp davor. Dies wird durch die iterative Entwicklung unterstützt, da bei jeder Wiederholung des Entwicklungsprozesses ein neues Inkrement entsteht, d. h. ein neuer Prototyp, der mehr Funktionalitäten benutzt als der vorangegangene.

- (c) Nennen Sie jeweils zwei Vor- und Nachteile eines iterativen und inkrementellen Vorgehens im Vergleich zum Wasserfallmodell.

Lösungsvorschlag

Vorteile:

- Risiken können früher erkannt werden.
- volatile Anforderungen können besser berücksichtigt werden.
- inkrementelle Auslieferung wird erleichtert.

Nachteile:

- komplexeres Projektmanagement
- schwerer messbar
- (Mehrarbeit)<sup>a</sup>

<sup>a</sup>Quelle: <https://www.pst.ifi.lmu.de/Lehre/WS0607/pm/vorlesung/PM-02-Prozess.pdf>

### Aufgabe 3: UML Diagramme in der Anwendung [Bestellsystem]

Gegeben sei folgender Sachverhalt:

Für eine Verwaltungssoftware einer Behörde soll ein Bestellsystem entwickelt werden. Dabei sollen die Nutzer ihre Raummaße eingeben können. Anschließend können die Nutzer über ein Web-Interface das Büro gestalten und Möbel (wie zum Beispiel Wandschränke) und andere Einrichtungsgegenstände in einem virtuellen Büro platzieren. Aus dem Web-Interface kann die Einrichtung dann direkt bestellt werden. Dazu müssen die Nutzer ihre Büro-Nummer und den Namen und die Adresse der Behörde eingeben und die Bestellung bestätigen.

Weiterhin können Nutzer auch Büromaterialien über das Web-Interface bestellen. Dazu ist anstatt der Eingabe der Raummaße nur das Eingeben von Büro-Nummer und des Namens und der Adresse der Behörde erforderlich.

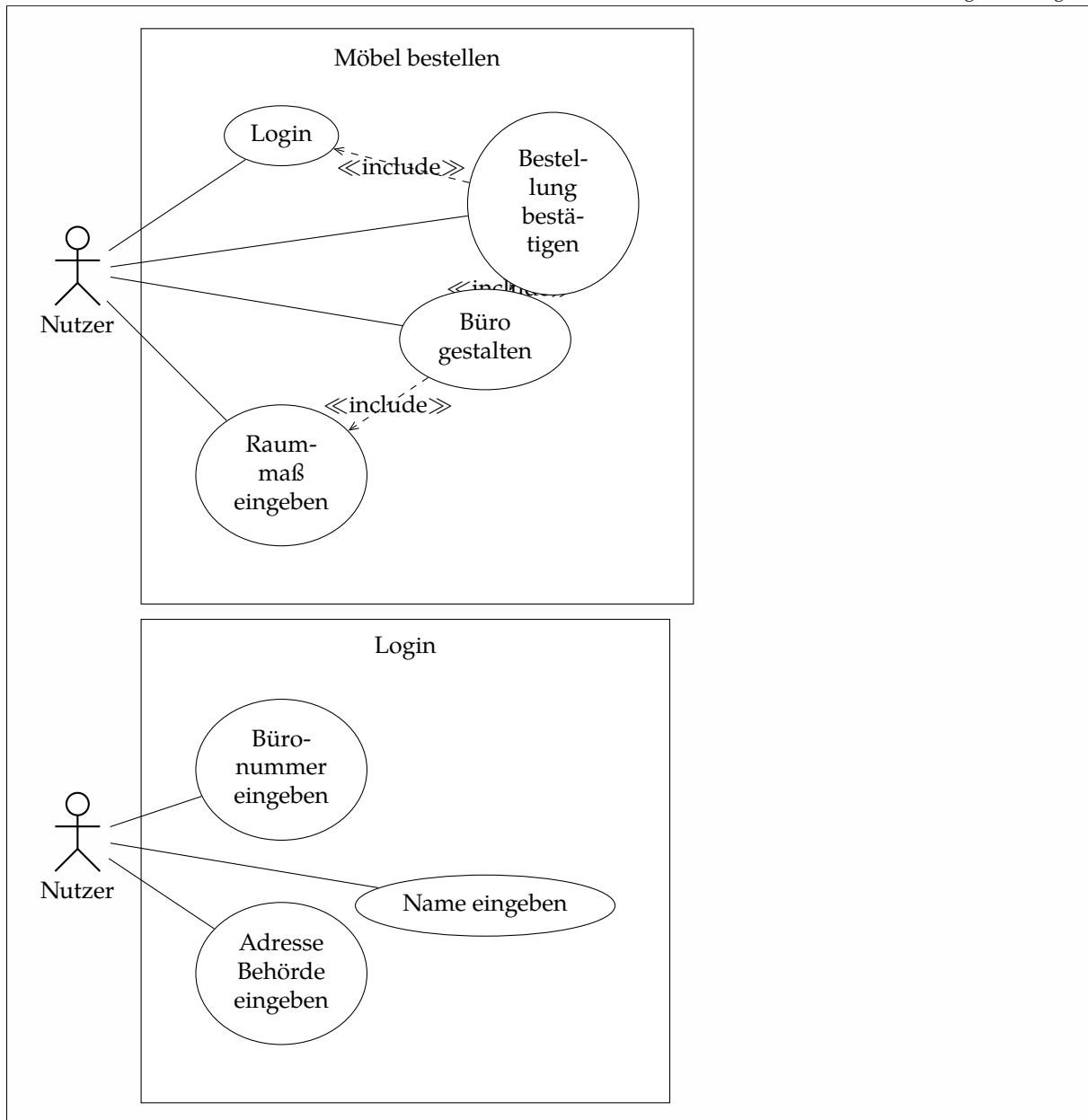
Zusätzlich zum Standard-Nutzer können sich auch Administratoren im System anmelden und Möbel zur Kollektion hinzufügen und aus der Kollektion entfernen. Die Möbel können eindeutig durch ihre Inventurnummer identifiziert werden.

Um jegliche Veränderungen im System protokollieren zu können müssen Nutzer und Administratoren zur Bestätigung eingeloggt sein.

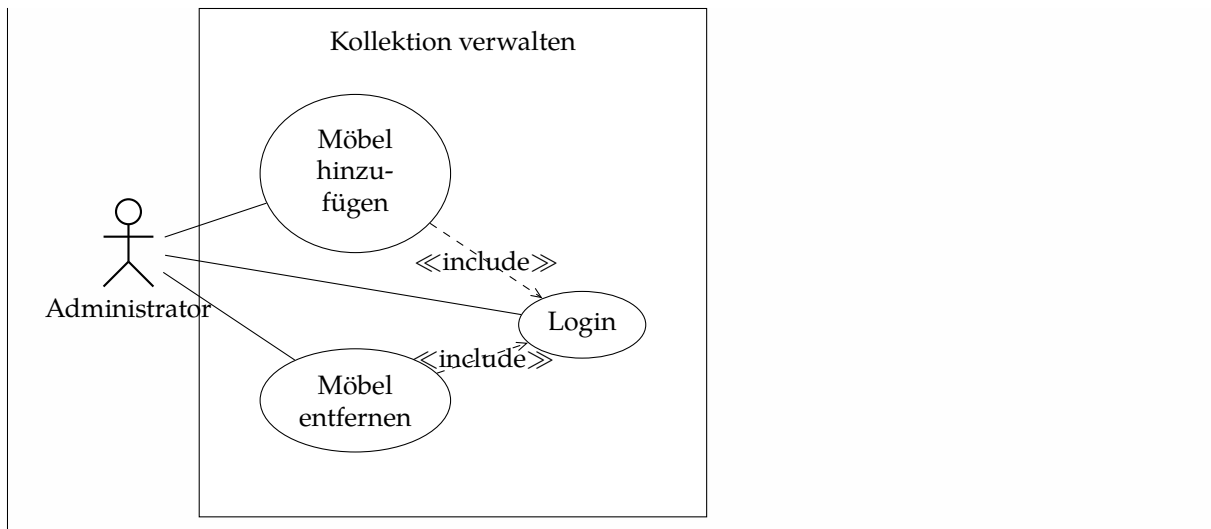
- (a) Erfassen Sie die drei Systemfunktionen *Möbel bestellen*, *Login* und *Kollektion verwalten* in einem UML-konformen Use Case Diagramm.

Login  
Kollektion  
verwalten

Lösungsvorschlag







- (b) Erstellen Sie ein UML-Klassendiagramm, welches die Beziehungen und sinnvolle Attribute der Klassen „Nutzer, Büro, Möbelstück und Wandschrank“ darstellt.
- (c) Instanzieren Sie das Klassendiagramm in einem Objektdiagramm mit den zwei Nutzern mit Namen Ernie und Bernd, einem Büro mit der Nummer CAPITOL2 und zwei Schränken mit den Inventurnummern S1.88 und S1.77.
- (d) Geben Sie für ein Büromöbelstück ein Zustandsdiagramm an. Überlegen Sie dazu, welche möglichen Zustände ein Möbelstück während des Bestellvorgangs haben kann und finden Sie geeignete Zustandsübergänge.