

SE Besprechung

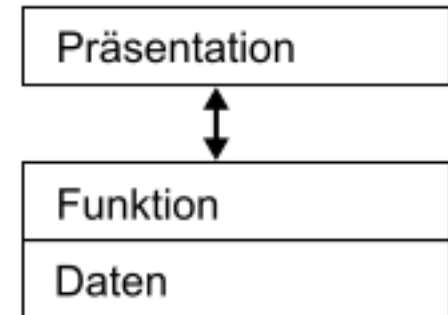
Übung 3 – Softwareprozesse

Analyse der Systemkomponenten (3 Punkte)

Mögliche Ansätze:

3-Schichten-Architektur (tree-tier-architecture)

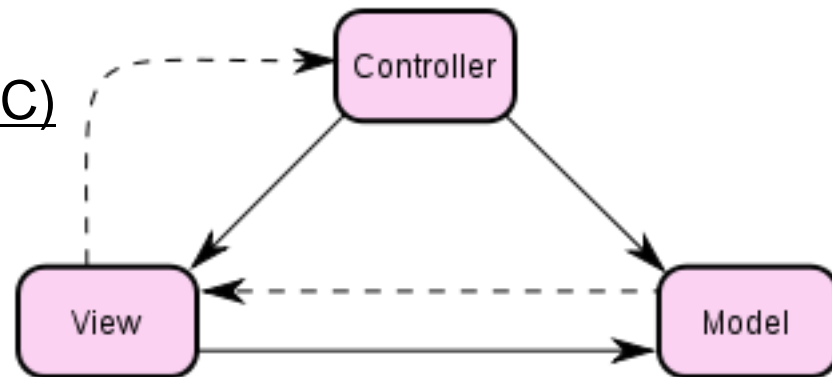
- Präsentation
- Anwendungslogik
- Daten



➤ Präsentationsschicht interagiert nie direkt mit Datenschicht

Model View Controller (MVC)

- Modell
- Präsentation
- Steuerung



Durchgezogene Linie: direkte Assoziation

Gestrichelte Linie: indirekte Assoziation (z.B. durch Beobachter)

Analyse der Systemkomponenten (3 Punkte)

Must-Haves:

Systemkomponente	Beschreibung
Benutzerinterface	<p>Website: Intuitives Interface zur Auswahl von verschiedenen Tickets (Studenten-/Angestelltenrabatt, ..)</p> <p>Automat: gleiche Funktionalität wie Website, zusätzlich zur Kreditkarte Barzahlung möglich</p> <p>Admin-Interface: zum Updaten der Ticket-/Zonendatenbanken & Einspielen von Releases</p>
Interne Datenbank	Zur Speicherung von Kundendaten (Name, Adresse, Zahlungsinfos, ..), Ticketdaten (Rabatte, Reichweite, ..), Zoneninformationen,

Analyse der Systemkomponenten (3 Punkte)

Systemkomponente	Beschreibung
Controller / Basissystem	Enthält die gesamte Logik (Berechnung der Ticketpreise, ...)
Schnittstelle zur Uni Datenbank	Schnittstelle, welche Studentenstatus prüft. Gesendete Informationen: Namen, Matrikelnummer, ...
Kreditkartensystem / Schnittstelle zu Bank/Kreditkarteninstitut	Komponente welche die Zahlung mit Kreditkarte erlaubt. Gesendete Informationen: Name, Adresse, Preis, ...

Analyse der Systemkomponenten (3 Punkte)

Feedback:

- Häufig vergessene Komponenten:
 - Admin-Interface
 - Kreditkartensystem
 - Controller
- Komponenten falsch identifiziert:
 - Angestellte
 - Erweiterbarkeit
- Beschreibungen ungenau:
 - Ticket DB: In DB werden Tickets abgespeichert
- Mehrsprachigkeit
 - Schlecht: Sprachdatenbank
 - Besser: Informationen in DB mehrsprachig erfasst, Controller steuert Zugriff

Prozessmodell a) (3 Punkte)

Kriterien für Modellwahl:

- Anforderungen klar?
- Anforderungen ändern während der Projektdauer?
- Projektgrösse
- Wichtigkeit (Sicherheitskritisches System)?
- Entwicklungsrisiko
- Know How Entwickler / Kunde
- Verfügbarkeit / Kompetenz des Kunden

Prozessmodell a) (3 Punkte)

Phasenmodell:

- Durchführung:
 - Projekt wird erst konzipiert, dann realisiert
- Benötigt:
 - Fixe Anforderungen
 - Kleines Risiko (!= Wichtigkeit)
- Vorteil:
 - Strukturiert durch Planung
 - Fehlererkennung
- Nachteil:
 - Spät lauffähige Version -> Adäquat?
 - Motivation

Prozessmodell a) (3 Punkte)

Wachstumsmodell:

- Durchführung:
 - System wächst
- Benötigt:
 - Einführbarkeit in mehreren Schritten (Lieferung auch intern möglich)
- Vorteil:
 - Basisversion schnell verfügbar
 - Frühes Feedback von Usern -> Fehler
 - Motivation
- Nachteil:
 - Gefahr von schlechter strukturierter SW

Prozessmodell a) (3 Punkte)

Agile SW Entwicklung:

- Durchführung:
 - Schnelle, kundennahe Entwicklung mit wenig Dokumentation
- Benötigt:
 - Erfahrener SW Architekt
 - Kompetenter Kundenvertreter
 - Qualitätssicherung (z.B. Pair-Programming)
- Vorteil:
 - Starker Einbezug des Kunden
 - Schnelles Feedback
- Nachteil:
 - Gefahr von Ad-hoc Programmierung
 - Gefahr von schlecht strukturierter SW

Prozessmodell a) (3 Punkte)

Feedback:

- Erweiterungen nach Projektende möglich unabhängig vom Prozessmodell
 - Phasenmodell hat Probleme mit Anforderungsänderungen **während** der Entwicklung
- Ziel aller Modelle ist es Fehler früh zu finden und Risiko zu minimieren
 - Jedes Modell hat andere Fehler und Risiken
 - Phasenmodell: Spezifikationsfehler
 - Wachstumsmodell: Adäquatheitsfehler
- Kreditkartensystem haben andere schon gemacht
 - = technisches Risiko gering
 - != Know How vorhanden

Prozessmodell b) (2 Punkte)

Keine richtige Lösung, kommt auf Argumentation an!

Beispiel-Lösung:

Agile SW Entwicklung ungeeignet da:

- Anforderungen sind stabil
- Abgrenzung der Komponenten schwierig
- Kompetenz von Kunde / SW Architekt unklar

Phasenmodell eher ungeeignet da:

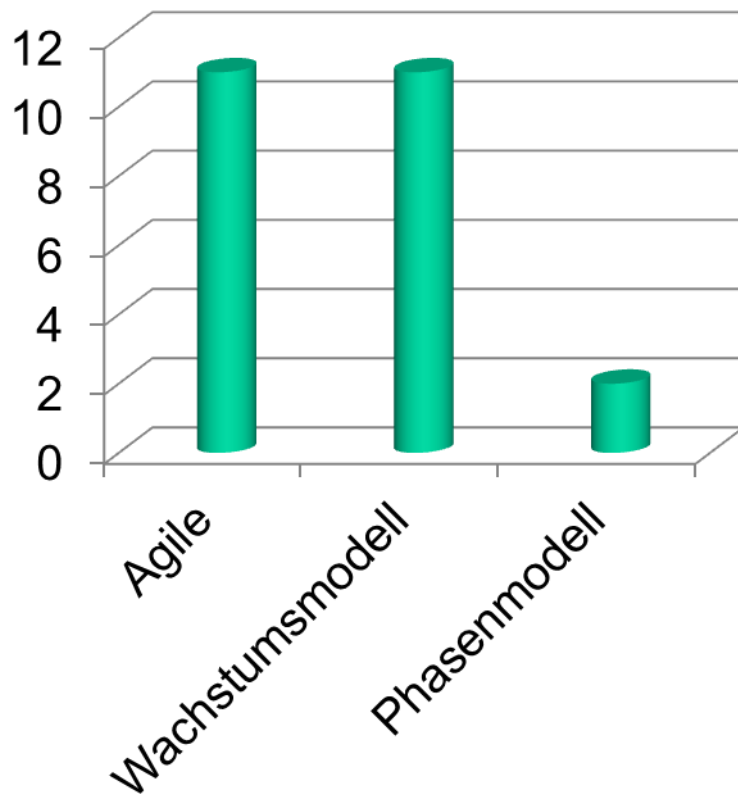
- Unnötig lange Konzeptionsphase
- + Vorteil wäre aber, dass Anforderungen nicht ändern

Wachstumsmodell gewählt da:

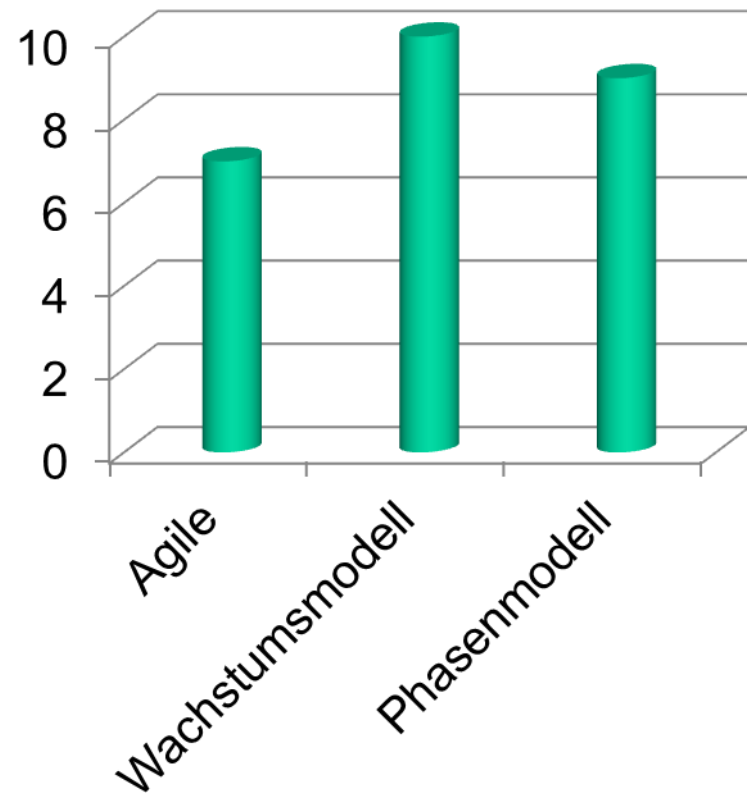
- + Einführung einer Basisversion ohne Mehrsprachigkeit / Möglichkeit von Rabattcodes für Arbeitnehmer
- + Schnelles Feedback der User / des Kunden

Prozessmodell c) (2 Punkte)

Social Media



Versicherungsgesellschaft



Prozessmodell c) (2 Punkte)

Bewertung:

•Social Media:

- Wachstumsmodell / Agile: 0.5 Punkte
- Begründung: 0.5 Punkte

•Versicherungsgesellschaft:

- Agile: 0.5 Punkte
- Begründung: 0.5 Punkte

Prozessmodell c) (2 Punkte)

Social Media: Wachstumsmodell (bevorzugt)

- System *wächst* → schrittweise Einführung
- Schnelle Markteinführung
- Jede Teillieferung ist verwendbar
- Volle Funktionalität nicht bereits zu Beginn benötigt
- Grosses System mit langer Entwicklungszeit
- Kurze Rückkopplungszyklen

Prozessmodell c) (2 Punkte)

- Auch möglich: Agile Development
- NICHT Phasenmodell:
 - Projekt ist gross
 - Nicht viel existierendes Wissen
 - Anforderung sind weder klar noch klar definierbar

Prozessmodell c) (2 Punkte)

Versicherungsgesellschaft: Agile

- Anforderungen können sich stark ändern
 - Expansion in verschiedene Länder: andere politische, rechtliche und kulturelle Randbedingungen, jeder Markt kann spezielle Eigenheiten aufweisen
- Einbezug von Kunden wichtig auf Grund Komplexität
- NICHT Wachstumsmodell
 - Anforderungen ändern
 - Harte Zeitanforderungen an das Projekt
- NICHT Phasenmodell:
 - Grosses Projekt (3 Länder)

Meilensteine (5 Punkte)

Häufige Fehler:

- Ganzes Projekt betrachten, nicht nur Implementierungsphase
 - Überprüfbarkeit der Meilensteine nicht gegeben
 - > Begriffe wie „fertig“, „abgeschlossen“ etc. sind ohne genaue Kriterien **undefiniert** (= wertlos!)
- Überprüfung der Meilensteine:
 - Tests (Unit, Komponenten, Integration, System)
 - Review (benötigt ebenfalls Testergebnisse)
 - Abnahmetest beim Kunden

Prototyping (5 Punkte)

Übungsbeschreibung:

“in welchen Phasen **und** für welche Komponenten” + “welche Art von Prototyp” + “Begründen...” → alles in einer Tabelle darstellen.

⇒ Tabelle sollte so aussehen:

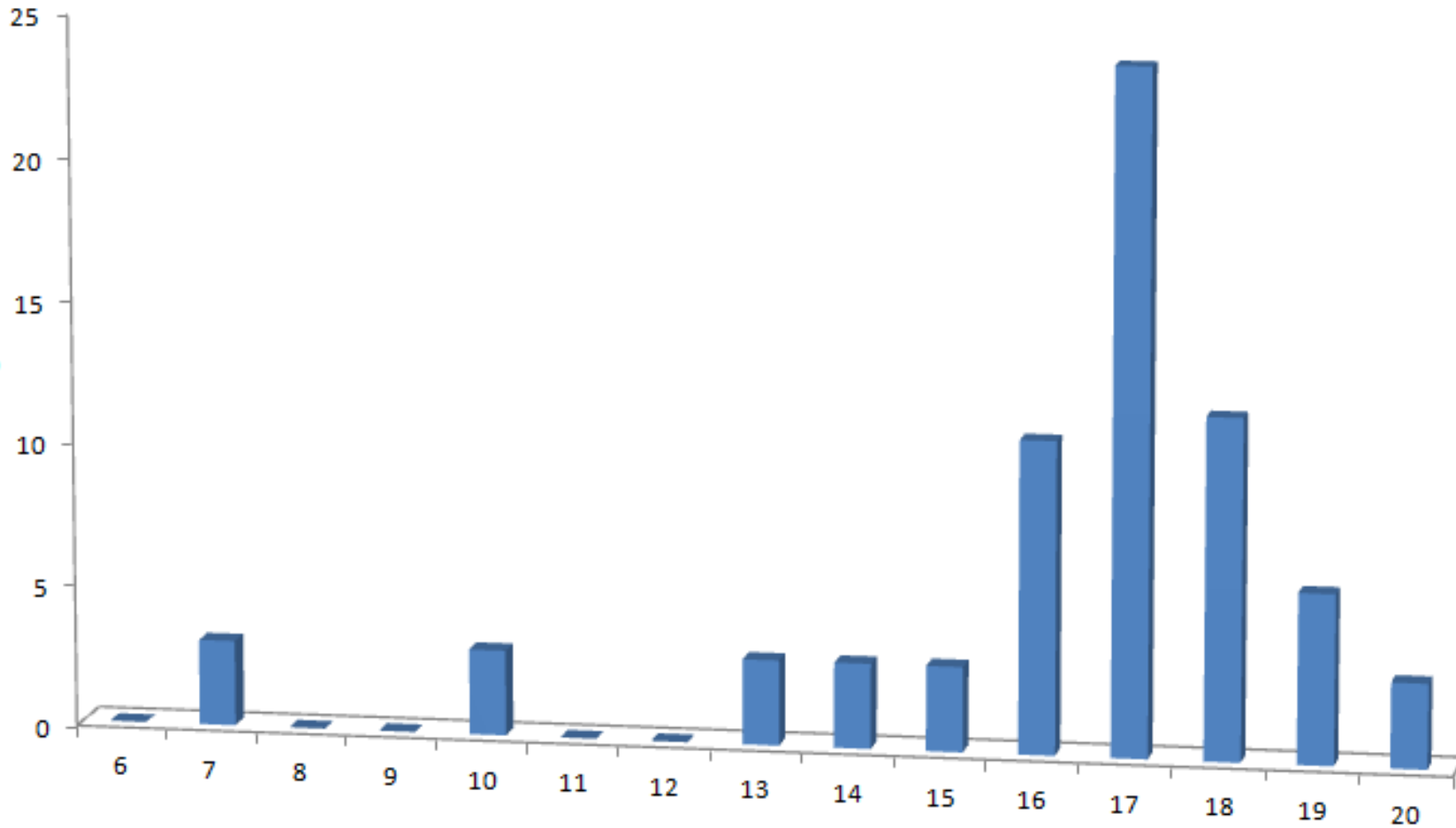
Phase	Komponente	Prototyp	Begründung

Prototyping (5 Punkte)

Kommentare:

- Keine Folgefehler aus Komponentenanalyse Aufgabe 2.1
- Phasenmodell keine evolutionären Prototypen
 - Im Gegensatz zum Wachstumsmodell.
- GUIs: explorative Prototypen empfehlenswert
- Um kritische Aspekte des Systems zu verifizieren: experimentelle Prototypen empfehlenswert
- Bewertung:
 - Korrekte Nennung der Phasen: 1 Punkt
 - Korrekte Nennung der Komponenten: 1 Punkt
 - Sinnvolle Vorschläge für Prototypen (min. 4): 0.5 Punkte x 4 = 2 Punkte
 - Begründung: 1 Punkt

Übung 3 Resultate



Allgemeines

- Fragen/Feedback zu Übung 3?