

Warum strukturierte Softwareentwicklung notwendig?

- Strukturierung der wichtigsten Aktivitäten
- Früherkennung von Fehlern
- Minimierung von Risiken

Begriffe:

- Prozess
 - Beschreibung Aktivitäten, Rollen und Artefakte(Informationen)
 - Software-Entwicklung und Wartung
- Prozessmodell
 - Präskriptives Modell (Vorgehensmodell und Organisationsstrukturen)
 - Planung und Lenkung
 - Unified Process, V-Modell, Scrum,...

Klassifizierung Software Probleme:

- 1.Dimension:** Requirements (y-Achse Agreement)
- 2.Dimension:** Technology (x-Achse Certainty)
- 3.Dimension:** Skills,Intelligence, Experience, Attitudes, Prejudices

0.1 Vorgehensmodelle

1. Code and Fix
2. Wasserfallmodell
3. Iterative und inkrementelle Modelle

Code and Fix

- **Definition**
 - Codierung / Korrektur im Wechsel mit Ad-hoc Tests
- **Vorteile**
 - Schnell vorankommen
 - Schnelle Ergebnisse
 - Einfache Tätigkeiten (Codieren, Testen, Fixen)
- **Nachteile**
 - Schlecht planbar und keine Unterstützung im Team
 - Aufwand hoch für Korrekturen
 - Schlecht wartbare Software

Wassefallmodell

- **Definition**

- Folge von Aktivitäten/Phasen, gekoppelt durch Teilergebnisse (Dokumente). Reihenfolge ist fest definiert.

- **Vorteile**

- hohe Planbarkeit
- Klare Aufteilung der SWE (Analyse, Design, Test,...)

- **Nachteile**

- Schlechtes Risikomanagement (Lösungskonzept nur auf Papier validiert)
- Anforderungen zu Beginn nie alle bekannt

Iterativ-inkrementelle Modelle

- **Definition**

- Geplante und kontrollierte Iterationen inkrementell entwickelt

- **Vorteile**

- Flexibles Modell bei unklaren Anforderungen
- Gutes Risikomanagement (Mitarbeiter und Technologie)
- Frühe Einsetzbarkeit der Software und Feedback

- **Nachteile**

- Upfront Planbarkeit hat Grenzen (Funktionalität, Zeit und Kosten)
- Braucht Involvierung und Steuerung durch den Kunden über ganze Projektdauer

0.2 Agile SWE

- Basiert auf iterativ-inkrementellen Prozessmodell
- Fokussiert auf gut dokumentierten und getesteten Code statt auf ausführlicher Dokumentation
- Sammlung von Ideen SWE Prozess flexibler und schlanker zu machen
- Adressiert bekannten Probleme bei klassischen Software-Prozessmodellen

Strategie

- Definierte Prozesskontrolle

- Planung am Anfang, Prozess gesteuert und überwacht
- Geeignet für gut planbare Problemstellungen
- Strategie: Steuerung

- Empirische Prozesskontrolle (Agil)

- Nur Grobplanung am Anfang
- Prozess fortlaufend überwacht
- Rollende Planung
- Geeignet für komplexe Problemstellungen
- Strategie: Regelung, Deming-Cycle (Plan-Do-Check-Act)

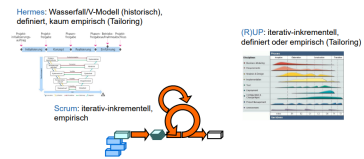


Abbildung 1: Charakterisierung Prozessmodellen