



Extreme Programming

Gegenvortrag

Zuverlässige Software
SS2009

Friedrich Gensicke

Gliederung

1. Das Alles-Oder-Nichts-Prinzip
2. Der ideale Kunde
3. Der ideale Programmierer
4. Weitere Probleme
 1. Skalierung
 2. Wartung
 3. Wechselnde Anforderungen

Das Alles-Oder-Nichts-Prinzip

- Extreme Programming ist Ansammlung bewährter Verfahren
- Nur in Kombination erfolgreich
- Scheitern schon durch Vernachlässigen nur einer Komponente

→ XP wird Schuld gegeben

Der ideale Kunde

- Hat weit reichende Aufgaben
 - Fachliche Projektleitung
 - Entscheidung über Zeitplan
 - ...

→ Erhebliche Anforderungen
 - Schwer einen solchen Mitarbeiter zu finden

→ Auftraggeber erhält „Lücke“
- Kunde-vor-Ort-Prinzip schwierig umsetzbar

Der ideale Kunde

Aspekte auf die der Kunde ungern verzichten will:

- Fehlende Dokumentation
 - Integration in bestehende Systemlandschaft
- Vertragsdetails und –bestandteile
 - Welche Leistung kann ich erwarten?

Der ideale Programmierer

- Vielfältige Erfahrung und Disziplin notwendig
 - Ständige Refactorings
 - Überrefactoring
 - Test-First

Problem von XP:

- Kopplung (bzgl. Arbeitsdisziplin) gegensätzlicher Aktivitäten

Weitere Probleme

- **Wartung**
 - Probleme durch fehlende Dokumentation
- **Skalierung**
 - Bei zu vielen Mitarbeiter und zu großen Projekten:
 - Gemeinsame Verantwortlichkeit zerbricht schnell
 - Keiner hat mehr Überblick und sieht sich für etwas verantwortlich
 - ...
- **Wechselnde Anforderungen / Änderungen**
 - Was passiert wenn die Änderungs-Kosten-Kurve anders verläuft (exponentiell)?

Literatur

- **eXtreme Programming**

Eine Einführung mit Empfehlungen und Erfahrungen aus der Praxis

Henning Wolf et al., 2. Auflage 2005

- **Extreme Programming**

Das Manifest

Kent Beck, 2003



Vielen Dank für die
Aufmerksamkeit!