

[Contact](#)[People](#)[Projects](#)[Teaching](#)[Papers](#)[Intranet](#)

---

## Vorlesung: Software-Engineering im Sommersemester 2004

---

### [Klausurergebnisse](#)

---

**Dozent:** [Ralf Möller](#), **Uebungsbetreuer:** [Miguel Garcia](#)

**Ort:** [TUHH](#), Eißendorfer Straße 40, R. 0007

**Zeit:** Do 13.45 - 15.15 Uhr

---

Eine Teilnahme an der schriftlichen Prüfung ist nur möglich, wenn fristgerecht eine Anmeldung beim jeweiligen für den betreffenden Studiengang zuständigen Prüfungsamt erfolgt ist. Eine Onsite-Anmeldung (mit Proviso) besteht bei dieser Vorlesung nicht. Grund: Es kann sonst nicht sinnvoll abgeschätzt werden, wieviele Plätze und Aufgabenzettel benötigt werden.

---

### Voraussetzungen:

---

### Inhalt:

1. [Einführung](#)  
Begriffsbestimmung, Systeme, Modelle, Qualitätskriterien
2. [Phasen und Vorgehensmodelle](#)  
Projekttypen, Prozeßmodelle (Vorgehensmodelle)  
[Uebung 1](#)
3. [Problemanalyse und Planung](#)  
[Uebung 2](#)
  - Ist-Analyse, Anforderungsanalyse
  - Lastenheft (Inhalt und Struktur)
  - Durchführbarkeitsuntersuchung: Einfache Verfahren zur Aufwandsschätzung
4. [Aufwandsabschätzung](#)  
[Uebung 3](#)
5. [Definition](#)  
Pflichtenheft (Inhalt und Struktur)  
[Uebung 4](#)
6. [Qualitätsmanagement, Analytische Qualitätssicherung, Software-Metriken](#) (z.B. Halstead, McCabe, u.a.), [Testverfahren](#) (z.B. JUnit, Teil 1)  
[Uebung 5](#)
7. [Testverfahren](#) (z.B. JUnit, Teil 2) [Konstruktive Qualitätssicherung](#) (z.B. ISO-9000)  
[Uebung 6](#)  
[JUnitDemo](#)
8. [TQM, Projektmanagement, Versionsverwaltung, Konfiguration](#) (z.B. CVS, WebDAV)
9. [Modellbasierte Softwareerstellung, Entwicklungsumgebungen](#) (von Miguel Garcia)  
(z.B. Model-Driven Architecture, Eclipse, Ant)  
[Uebung 7](#)

10. [Agile Software-Engineering-Methoden für Individualsoftware](#)  
(u.a.: Extreme Programming, Feature-Driven Development)
11. [Standardsoftware, Workflow-Spezifikationen und Referenzmodelle](#)  
(z.B. modelliert in ARIS)  
[Uebung 8](#)
12. [Enterprise Application Integration](#)  
(opportunistische Wiederverwendung)  
[Uebung 9](#)
13. [Software Product Lines](#)  
(organisierte bzw. strategische Wiederverwendung)
14. [Software-Reengineering](#)  
[Uebung 10](#)

Parts (1-6, 8) of the lecture are partially based on a previous lecture given by Prof. Florian Matthes and Prof. Joachim Schmidt. Lecture 9 was given by Miguel Garcia. For acknowledgments for other parts, see the presentations.

---

## Literatur:

(siehe Folien)

---

## Musterlösungen:

---

## Klausuren aus früheren Semestern:

---

[Ralf Möller](#)

Last modified: 01.04.04