



Universität  
Zürich<sup>UZH</sup>

Institut für Informatik

Martin Glinz   Thomas Fritz

# Software Engineering

Herbstsemester 2016

## Einleitung zur Vorlesung



# Ziele der Vorlesung

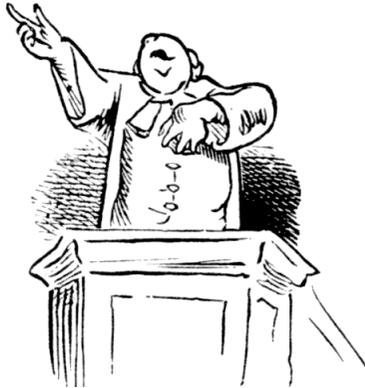
---

- Die Studierenden verfügen über ein **Grundwissen** in **Software Engineering** und kennen die Mittel **zur wirtschaftlichen Herstellung und Pflege guter Software**.
- Auf dieser Grundlage können sie
  - den **Stellenwert** und Bedeutung von Software und Software Engineering in Wirtschaft und Alltag **beurteilen**
  - ihr **Wissen** über Software-Entwicklung **systematisch einordnen**
  - die **Ursachen** für Schwierigkeiten und Probleme in Software-Projekten **erkennen**
  - **eigene** Software-Entwicklungsvorhaben **systematisch** und **zielgerichtet** **angehen**
  - als **Anwender** bzw. als **Manager** **erfolgreich** mit Software-entwicklern **zusammenarbeiten**

# Worum es geht

---

Professor, Software Engineering lehrend  
(frei nach Wilhelm Busch)



Die traur'ge Wahrheit lautet schlicht  
in der Praxis tun's die meisten nicht.



Man glaubt es kaum, es ist ein Graus,  
sie werfen das Geld zum Fenster hinaus.



Vom Software Engineering sind sie weit entfernt –  
der Grund ist der: sie haben's nie gelernt.



Drum, liebe Leute kommt herbei,  
ihr braucht SE, ich bring's euch bei.

# Inhalt

---

## Teil I: Grundlagen und Einführung

1. Software-Entwicklung und -Pflege als Problem
2. Ziele und Qualität
3. Modelle

## Teil II: Technische Aspekte

4. Spezifikation von Anforderungen
5. Entwurf von Software
6. Systematisches Programmieren
7. Validierung und Verifikation
8. Test
9. Review
10. Messen
11. Statische Analyse
12. Software-Evolution, Reengineering

## Teil III: Managementaspekte

13. Prozesse und Prozessmodelle
14. Software-Projektmanagement
15. Aufwandschätzung
16. Software-Qualitätsmanagement
17. Bewertung und Verbesserung von Prozessen
18. Produktivitätsfaktoren
19. Die Rolle der Menschen im SE

## Teil IV: Unterstützungsaspekte

20. Konfigurationsmanagement
21. Dokumentation
22. Werkzeuge

# Reihenfolge der Kapitel

---

- Die **Reihenfolge der Kapitel** ist auf eine Systematik des Stoffs (analog zu einem Lehrbuch) angelegt.
- Aus didaktischen Gründen werden die Kapitel **nicht in dieser Reihenfolge** behandelt.
- Den Plan mit der vorgesehenen Reihenfolge der Präsentation finden Sie auf den Webseiten zur Vorlesung.
- Beginn: **Kapitel 6: Systematisches Programmieren**

# Unterlagen, Literatur

---

- Vorlesungsfolien
- Begleitliteratur
  - J. Ludewig, H. Lichter: *Software Engineering: Grundlagen, Menschen, Prozesse, Techniken*. 3. Auflage. Heidelberg: dPunkt, 2013.
  - S. L. Pfleeger, J. Atlee: *Software Engineering: Theory and Practice*, 4th edition, Upper Saddle River, N.J.: Pearson Education International, 2010.
  - Martin Glinz: *Software Engineering*. Vorlesungsskript Universität Zürich, 2005. [Deckt nicht alle Kapitel der Vorlesung ab]
- Verweise auf weiterführende und vertiefende Literatur finden sich am Ende der jeweiligen Kapitel sowie im Skript

# Zielpublikum

---

- Studierende im **Studiengang BSc in Informatik**
  - Software Engineering ist ein **Pflichtmodul** im ersten Studienjahr der Bachelorstufe
  - Voraussetzung: Assessmentstufe (mindestens bedingt) bestanden
- Studierende **anderer Fakultäten** mit **Nebenfach Informatik**
  - Software Engineering ist ein **Wahlpflichtmodul**
  - Voraussetzung: Assessmentstufe des Nebenfachstudiums bestanden

# Buchung für Teilnahme am Leistungsnachweis

---

- An- und Abmeldung gemäss den publizierten Fristen
- Buchen unter  
<http://www.students.uzh.ch/booking.html>

# Bedingungen für Leistungsnachweis

---

- Erfolgreiche Teilnahme an den Übungen  
und
- Bestehen der Schlussklausur
- Gesamtnote: 2/3 Schlussklausur, 1/3 Übungspunkte unter den Voraussetzungen
  - mindestens 60 von 120 Übungspunkten erreicht
  - Note der Schlussklausur  $\geq 4.0$

# Übungsbetrieb

---

- Es werden **sechs Übungen** ausgegeben (mit je 20 Punkten)
- Für eine erfolgreiche Übungsteilnahme müssen insgesamt **mindestens 60 Übungspunkte** erreicht werden
- Übung 1: Verstehen und Ändern bestehender Software sowie Einarbeitung in das Google Web Toolkit GWT und die Google App Engine (Einzelarbeit)
- Übungen 2-6: Elemente der Applikationsentwicklung in einem kleinen, geführten Projekt (in Gruppen)
- Ausgabe Übung 1: **heute (!)**
- Unterlagen: Webseiten der Vorlesung und OLAT
- Weitere Details: **siehe Webseiten und OLAT**

# Zeiten

---

8.10	Beginn
8.10 - 9.45	Vorlesungsblock 1
9.45 - 10.15	Pause
10.15 - 11.50	Vorlesungsblock 2

In der Mitte der Vorlesungsblöcke je eine Kurzpause

# Weitere Informationen

---

## Webseite

- Alle Infos zur Vorlesung
- Link zu OLAT

<http://www.ifi.uzh.ch/rerg/courses/hs16/se>

## OLAT

- Übungsbetrieb
- Einschreibung in die Übungsgruppen
- Forum
- Blog

<https://lms.uzh.ch/url/RepositoryEntry/16110059650>

# Dieses Modul wird Ihnen präsentiert von...

---

- Professoren
  - Martin Glinz
  - Thomas Fritz



- Übungsleiterin
  - Sofija Hotomski
- Unterrichtsassistenten
  - Christoph Vogel
  - Livio Sgier