

0101seda010100

software engineering dependability

Software Entwicklung 2

Inhalte und Lernziele

- Inhalte von SE 2
- Lernziele SE 2
- Struktur SE 2
- Zusammenhang mit den anderen Lehrveranstaltungen der SE-Reihe sowie mit der Lehrveranstaltung Grundlagen des Software Engineering

- Prozessmodelle
- Lastenheft / Pflichtenheft
- Projektplanung
- UML in der Analyse
- UML im Entwurf
- Von UML zu Programmcode und zurück (am Beispiel Java)
- Strukturierte Analyse (SA)
- Entwurf
 - Structured Design
 - Entwurf nach Datenabstraktion
- Implementierung (Systematische Erzeugung von Code auf Basis von Modellen am Beispiel Compilerbau)
- Softwaretesten
- (Oberflächengestaltung)

- Übergeordnetes Lernziel in SE 2
- Sichere Beherrschung von grundlegenden Methoden, Techniken und Notationen der Softwareentwicklung
 - Standardprozesse
 - Lastenheft, Glossar, Pflichtenheft
 - Basistechniken der Projektplanung (Netzpläne, Gantt-Diagramme)
 - Funktional dekomponierende Analysetechniken (SA)
 - Entwurf nach funktionaler Abstraktion (SD)
 - Objektorientierte Analyse (UML)
 - Objektorientierter Entwurf (UML)
 - Implementierung (Compiler)
 - Techniken für den Softwaretest
- Sie sollten am Ende dieses Semesters in der Lage sein, die erlernten Techniken anhand von Einflussfaktoren zu bewerten, ihre Vor- und Nachteile zu erläutern und die Techniken selbstständig anzuwenden.

- Einführung in Softwareentwicklungsprozesse als Rahmen für die vorgestellten Techniken
- Durchsprache von wichtigen Techniken in der Reihenfolge ihres Auftretens entlang eines Phasenmodells
- Unterscheidung funktional dekomponierender und objektorientierter Methoden für Analyse und Entwurf
- Schwerpunkte auf Planung, Analyse, Entwurf und Test

- SE 1 führt in elementare Programmierfertigkeiten ein („Handwerkszeug“)
 - Programmieren, Bewertung von Algorithmen, Verifizieren, elementare Strukturprinzipien
 - Motto: „Ein Baumeister, der eine Kathedrale bauen will, sollte wissen, wie man mauert; auch wenn er das nicht jeden Tag tut!“
- SE 2 führt zentrale Techniken und Notationen für die Software-Entwicklung ein
 - UML, Funktionale Analyse- und Entwurfsnotationen, Testen, ...
 - Das Verständnis dieser Inhalte setzt die Inhalte aus SE 1 voraus und ist Grundlage für das Verständnis der Software-Entwicklung im Großen
- Software Engineering befasst sich mit der arbeitsteiligen Entwicklung großer Software-Systeme
 - Zentraler Inhalt sind sogenannte Prozesse, in denen unterschiedliche Personen Teilaufgaben der Entwicklung übernehmen und in definierter Weise zusammenarbeiten
 - Die Lehrveranstaltung Software Engineering nutzt die in SE vermittelten Kenntnisse als Inhalte von Prozessphasen und ergänzt den Prozess als solches, d.h. das Zusammenwirken der Phasen, die Organisation, Rollen, ...