

Modulhandbuch SE

Software Engineering

Bachelor Technische Informatik 2020

Version: 1 | Letzte Änderung: 03.09.2019 11:27 | Entwurf: 0 | Status: vom Modulverantwortlichen freigegeben |
Verantwortlich: Nissen

– Allgemeine Informationen

Anerkannte Lehrveranstaltungen	<u>SE Nissen</u>
---	------------------

Gültig ab	Wintersemester 2021/22
------------------	---------------------------

Fachsemester	3
---------------------	---

Dauer	1 Semester
--------------	------------

ECTS	5
-------------	---

Zeugnistext (de)	Software Engineering
-------------------------	----------------------

Zeugnistext (en)	Software Engineering
-------------------------	----------------------

Unterrichtssprache	deutsch
---------------------------	---------

abschließende Modulprüfung	Ja
---------------------------------------	----

Modulprüfung

Benotet	Ja
----------------	----

Konzept	Schriftliche Klausur. Die Klausur stellt sicher, dass jeder Studierende auch individuell die Ziele des L.O. erreicht hat, durch Aufgaben der folgenden Typen: Fragen zu Grundwissen über Spezifikationstechniken, Entwurfsprinzipien, Methoden zur Software-Prüfung (K.2, K.3), Modellierung verschiedener Perspektiven eines Software-Systems (K.1, K.2, K.4, K.5, K.9), Erstellung geeigneter Testfälle (K.7, K.9), Anwendung einfacher Entwurfsmuster (K.9).
----------------	---

Frequenz	Jedes Semester
-----------------	----------------

– Allgemeine Informationen

Inhaltliche Voraussetzungen

PI2 - Praktische Informatik 2 Fundierte Kenntnisse in der Programmierung.

PP - Programmierpraktikum Praktische Erfahrungen in der Programmierung.

Handlungsfelder

Systeme zur Verarbeitung, Übertragung und Speicherung von Informationen für technische Anwendungen planen, realisieren und integrieren

Anforderungen, Konzepte und Systeme analysieren und bewerten

Informationstechnische Systeme und Prozesse organisieren und betreiben

Learning Outcomes

ID	Learning Outcome
LO1	<p>Was: Das Modul vermittelt Kompetenzen zur Erstellung von Software-Systemen in allen Projektphasen von der Dokumentation von Systemanforderungen (K.1, K.4), der Spezifikation und Modellierung von Systemeigenschaften (K.1, K.2, K.3, K.5, K.9), der Erstellung und Bewertung eines Systementwurfs für vorgegebene Qualitätsziele (K.1, K.2, K.5, K.9, K.10), der Prüfung von Systemeigenschaften (K.7, K.9, K.10), und der Erstellung leserlichen Programmcode (K.6). Im parallel laufenden Praktikum werden die Kompetenzen zur Erstellung umfangreicher technischer Texte (K.2, K.4), zum Entwurf eines Software-Systems unter Berücksichtigung von Qualitätszielen (K.1, K.3, K.5) zur Realisierung von Systemmodellen in Programmcode (K.6), zur Prüfung von Programmcode (K.6, K.7) und zur Prüfung eines erstellten Software-Systems (K.7, K.9, K.10) verstärkt.</p> <p>Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs/Übungsteil unter Einbeziehung einer umfangreichen Fallstudie und betreut parallel dazu ein Praktikum, in dem die Studierenden ein kleines Software-System planen, entwerfen, realisieren und analysieren.</p> <p>Wozu: Kompetenzen in der Entwicklung eines Software-Systems sind essentiell für technische Informatiker, die im HF 1 arbeiten wollen. Durch die</p>

Arbeit
 an einem komplexeren
 Beispielsystem erwerben die
 Studierenden zudem
 Erfahrungen,
 die essentiell für das HF 2 sind,
 u.a. Anforderungen spezifizieren,
 Systeme und
 Software-Architekturen
 entwerfen, realisieren und
 bewerten. Die Entwicklung eines
 Software-Systems
 über mehrere Praktikumstermine
 hinweg, vermittelt den
 Studierenden erste
 Erfahrungen in der Organisation
 umfangreicherer Systeme unter
 Berücksichtigung
 vorgegebener Qualitätsziele (HF
 3).

Kompetenzen

Kompetenz	Ausprägung
In Systemen denken	diese Kompetenz wird vermittelt
Systeme analysieren	diese Kompetenz wird vermittelt
fachliche Probleme abstrahieren und formalisieren	diese Kompetenz wird vermittelt
Konzepte und Methoden der Informatik, Mathematik und Technik kennen und anwenden	diese Kompetenz wird vermittelt
Systeme entwerfen	diese Kompetenz wird vermittelt
Systeme prüfen	diese Kompetenz wird vermittelt
Typische Werkzeuge, Standards und Best Practices der industriellen Praxis kennen und einsetzen	diese Kompetenz wird vermittelt
In vorhandene Systeme einarbeiten und vorhandene Komponenten sinnvoll nutzen	diese Kompetenz wird vermittelt

Systeme realisieren

diese Kompetenz wird
vermittelt

– Vorlesung / Übungen

Typ	Vorlesung / Übungen
------------	---------------------

Separate Prüfung	Nein
-------------------------	------

Exemplarische inhaltliche Operationalisierung	Anhand praxisnaher Beispiele kann die Modellierung eines Systems durchgeführt werden. Es könnten unterschiedliche Architekturvorschläge für ein System im Hinblick auf nicht-funktionale Anforderungen untersucht werden. Die Anwendung von Testmethoden kann für hinreichend komplexe Programmteile geübt werden.
--	---

– Praktikum

Typ	Praktikum
------------	-----------

Separate Prüfung	Ja
-------------------------	----

Exemplarische inhaltliche Operationalisierung	Der Umgang mit Systemmodellen sollte aus zwei Richtungen erfolgen. Zum einen sollten vorgegebene Modelle in korrekten Code übersetzt werden. Zum anderen sollten für vorgegebene Anforderungsspezifikationen entsprechende Modelle für unterschiedliche Systemperspektiven erstellt werden. Für eine selbst erstellte Spezifikation sollen entsprechende Programmteile realisiert werden und die Ableitung und Implementierung geeigneter Testfälle erfolgen.
--	---

Separate Prüfung

Benotet	Nein
----------------	------

Frequenz	Einmal im Jahr
-----------------	----------------

Voraussetzung für Teilnahme an Modulprüfung	Ja
--	----

Konzept	Im Praktikum werden die Studierenden folgende Typen von Aufgaben bearbeiten: Erstellung von Programmcode auf Grundlage einer gegebenen Software-Modellierung (K.6). Umfangreiche Spezifikation eines Systems anhand einer gegebenen Anforderungsdefinition (K.1, K.2, K.3, K.4, K.9). Erstellung eines Software-Entwurfs auf Basis der eigenen Spezifikation (K.3, K.5, K.9), Realisierung des Entwurfs in geeignetem Programm-Code unter Verwendung bereits erstellter Komponenten (K.6, K.9, K.10) und Prüfung des Codes gegen die vorhandene Spezifikation (K.7, K.9).
----------------	--

