

Software Engineering 1

Code	SWE 1		
Fachbereich	Software-Entwicklung		
Studiengänge	BSc Informatik, BSc Wirtschaftsinformatik, DAS Applikationsentwicklung		
Vertiefungsrichtungen	-		
Art des Studiengangs	<input checked="" type="checkbox"/> Bachelor	<input type="checkbox"/> Master	<input checked="" type="checkbox"/> CAS/MAS/EMBA
Studienniveau*	<input type="checkbox"/> Basic	<input checked="" type="checkbox"/> Intermediate	<input type="checkbox"/> Advanced <input type="checkbox"/> Specialised
Typus**	<input checked="" type="checkbox"/> Core course	<input type="checkbox"/> Related course	<input checked="" type="checkbox"/> Minor course
ECTS-Credits	5		
Präsenzverpflichtung	100%		
Arbeitsaufwand in Std	150		
Verantwortliche Ansprechperson	Fachbereichsleiter: Peter Böhnlein	Autor: Peter Böhnlein	
Zu entwickelnde Kompetenzen	<p>Die Studierenden kennen Ziele und Vorgehensweisen des Software Engineering. Sie können</p> <ul style="list-style-type: none"> • die Anforderungen an eine Software-Lösung formulieren: „Das Problem ist verstanden.“ • die Software-Lösung konzipieren und begründen: „Die Lösung ist verstanden.“ • die wichtigsten Ansätze zum Bau, zur Einführung und zur Wartung von Software anwenden: „Die Lösung ist im Einsatz.“ 		
Lerninhalte	<p>Übersicht und Einführung, Software-Projektmanagement, Analyse, Entwurf und Software-Architekturen, Software Life Cycles.</p> <p>Der Inhalt des Moduls WBQ (Block 3) ist auch Teil dieses Moduls. Beachten Sie die Angaben dazu im Modulplan WBQ (WBQ-3).</p>		
Lehr- und Lernmethoden (Fernstudium nach dem Blended-Learning-Konzept)	Selbststudium <ul style="list-style-type: none"> • Lektüre und Erarbeiten des Stoffes • Lösen von Aufgaben und Fallstudien • Gruppenarbeit 	Online-Studium <ul style="list-style-type: none"> • Forumsdiskussionen • Einreichen von Aufgaben • Repetitionsaufgaben • Online-Feedback etc. 	Präsenzstudium <ul style="list-style-type: none"> • Lehrgespräch • Gruppenarbeiten • Gruppendiskussionen • Fallbeispiele
Unterrichtssprache	Deutsch		
Leistungsbewertung	Modularbeit (60%), Kurzreferat (20%), Zertifikatsprüfung IREB (20%)		
Lehrmittel	<p>Ludewig, J.; Lichter, H.: Software Engineering: Grundlagen, Menschen, Prozesse, Techniken; ISBN 978-3-864-90092-1</p> <p>Seidl, M.; Brandsteidl, M.; Huemer, Ch.; Kappel, G.: UML@Classroom; ISBN 978-3-898-64776-2</p> <p>Pohl, K.; Rupp, Ch.: Basiswissen Requirements Engineering: Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level, ISBN 978-3-86490-283-3</p>		
Vorkenntnisse: Modul(e)	PM-G, OOP		
Anschlussmodul(e)	SWE 2		
Bemerkungen	Dieses Modul wird durch die Module SWE 2 und SWQ vertieft und ergänzt.		

*Studienniveau	B	Basic level course: Modul zur Einführung in das Basiswissen eines Gebiets.
	I	Intermediate level course: Modul zur Vertiefung der Basiskonntenisse.
	A	Advanced level course: Modul zur Förderung und Verstärkung der Fachkompetenz.
	S	Specialised level course: Modul zum Aufbau von Kenntnissen und Erfahrungen in einem Spezialgebiet.
**Typus	C	Core course: Modul des Kerngebiets eines Studienprogramms.
	R	Related course: Unterstützungsmodul zum Kerngebiet (z.B. Vermittlung von Vor- oder Zusatzkenntnissen).
	M	Minor course: Wahl- oder Ergänzungsmodul.

1 Stoffplan

1.1 Leitidee

Software als ein zentrales Produktionsmittel vieler Industrien ist in ihrer Erstellung immer noch problematisch. Die Studierenden erlernen daher die heute verfügbaren Mittel zur „industriellen“, d.h. ingenieurmässigen Erstellung von Software.

Sie sind damit in der Lage, Ziele und Vorgehensweisen des Software Engineering begründet zu vertreten sowie die relevanten Rollen und Prozesse in Software-Projekten zu definieren. Sie kennen die verschiedenen Arbeitsschritte zur Software-Erstellung – Analyse, Entwurf und Implementierung – sowie die Artefakte als Resultate dieser Arbeitsschritte.

Sie kennen Architekturen und Prinzipien zum Entwurf von Software als Voraussetzung für die Codierung und die Wartbarkeit. Ausserdem verfügen sie über Grundlagenwissen zum Steigern von Produktivität und Qualität, dem Senken der Fehlerrate und dem leichten Fehlerfinden in einem Software-Projekt sowie dem Schaffen guter Voraussetzungen für die Weiterentwicklung von Software.

Die speziellen Bedingungen bei Offshore-Entwicklung und Outsourcing der Entwicklung sind ihnen bekannt, und sie können ihre Entwicklungsprojekte entsprechend auslegen und steuern. Die Querbeziehungen und Abhängigkeiten zu und von anderen Aktivitäten der Software-Entwicklung – vor allem dem Projektmanagement und der Software-Qualität – sind ihnen bekannt.

1.2 Inhalt

Dieses Modul dient als Basismodul und legt den Schwerpunkt auf die frühen Phasen der Software-Entwicklung: Analyse und Entwurf. Auf die anderen relevanten Phasen gehen wir überblicksmässig ein. Diese werden im Modul Software Engineering 2 vertieft.

Nach einer Übersicht über die Ingenieurdisziplin Software Engineering und die verschiedenen bekannten Vorgehens- und Prozessmodelle werden die einzelnen Arbeitsschritte zur Erstellung einer Software genauer betrachtet.

Diese Arbeitsschritte kommen in (fast) allen Software-Entwicklungsmodellen vor. In modernen Modellen werden diese Schritte wiederholt und zum Teil parallel ausgeführt.

Die Studierenden eignen sich Kenntnisse über ein methodisches Vorgehen zur Software-Entwicklung an, verschiedene Vorgehens- und Prozessmodelle sind ihnen bekannt.

Sie können Methoden zum Gewinnen, Analysieren und Strukturieren von Anforderungen als Basis des Entwurfs einer Software in Software-Projekten umsetzen.

Die Teilnehmer erwerben die Fähigkeit, ingenieurmässig Massnahmen geplant und zielgerichtet bei der Software-Erstellung zu definieren und einzusetzen. Sie kennen die entsprechenden Methoden und haben diese eingeübt. Ausserdem sind sie in der Lage, mit entsprechenden Spezialisten zusammenzuarbeiten: Abstimmung, Planung und Abnahme deren Ergebnisse.

Die Studierenden sind in der Lage, in Gruppen zu arbeiten und sich an Diskussionen zu beteiligen, in denen oft verschiedene Meinungen und Ansichten als zutreffend stehen gelassen werden müssen – da Software Engineering keine exakte Ingenieurdisziplin ist.

Die Teilnehmer sind in der Lage, die mit der Software-Entwicklung verbundenen Aufgaben und Rollen zu definieren, sowie deren Notwendigkeit zu vermitteln. Sie lernen, im Team unterschiedliche Rollen einzunehmen und entsprechende Beiträge zu leisten.

Die Studierenden können ihre Erfahrungen (z.B. aus OOP und aus dem beruflichen Alltag) mit dem gelernten Stoff verknüpfen und neue Erkenntnisse sowie kritische Fragen daraus ableiten.

Die Studierenden erkennen ihren Erklärungs- resp. Vertiefungsbedarf in Bezug auf den Lernstoff und bringen Wünsche zum Setzen von Schwerpunkten in den Präsenzen vorgängig ein.

Dieses Modul bereitet auch auf die Prüfung zur offiziellen iREB-Zertifizierung als Requirements Engineer Foundation Level vor (<http://www.ireb.org/pruefungen/foundation-level.html>). Die Studierenden legen diese Zertifizierungsprüfung als Teil des Moduls ab.

Einführung und Ziele

- Anforderungen und Projektvorgehen
- Begriffseinführungen
- Vorgehens- und Prozessmodelle

– „Die Aufgabe verstehen“

Analyse

- Erfassen, Dokumentieren, Verhandeln und Pflegen von Anforderungen
- Qualitätsmerkmale von Software-Anforderungen
- Requirements Reviews
- Aufbau und Struktur von Anforderungsdokumenten
- Werkzeuge

– „Das Problem verstehen“

Entwurf

- Modelle
- Modellierung statischer Aspekte
- Modellierung dynamischer Aspekte
- Strukturiertes Zerlegen (Dekomposition)
- Methoden des Entwurfs
- Architekturen
- Die Rolle von Entwurfsmustern

– „Die Lösung verstehen“

Realisierung

- Werkzeuge
- Frameworks
- Verteilte Software-Entwicklung

– „Die Lösung bauen“

Qualität

- Testverfahren

– „Die Lösung prüfen“

Einführung der Software-Lösung

- Vorgehen
- Outsourcing

– „Die Lösung einsetzen“