

Software Engineering – Architektur

Code	SWEA		
Fachbereich	Software-Entwicklung		
Studiengänge	BSc Informatik, BSc Wirtschaftsinformatik, MAS Web4Business		
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 type="checkbox"/> Intermediate	<input checked="" type="checkbox"/> Advanced <input checked="" type="checkbox"/> Specialised
Typus**	<input checked="" type="checkbox"/> Core course	<input checked="" 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>Sie können eine Software-Lösung nachvollziehbar konzipieren: „Die Lösung ist entworfen und verstanden.“</p> <p>Sie können die Schnittstellen zum Bau der Software bedienen: „Die Lösung kann gebaut werden.“</p> <p>Sie können ein geeignetes Vorgehen zur Produktionseinführung umsetzen: „Die Lösung ist zur Nutzung bereit.“</p>		
Lerninhalte	<ul style="list-style-type: none"> Die Rolle der Qualitätsanforderungen (nicht-funktionale Anforderungen) Architektur: Software-Architektur und -Design Software-Implementierung und Software-Einführung 		
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	Zertifikatsprüfung iSAQB (50%), WBQ-6-Übung (30%), Kurzreferat (20%)		
Lehrmittel	Ludewig, J.; Lichter, H.: Software Engineering: Grundlagen, Menschen, Prozesse, Techniken; ISBN 978-3-864-90092-1 Starke, G.: Effektive Software-Architekturen – ein praktischer Leitfaden, ISBN 978-3-446-44361-7		
Vorkenntnisse: Modul(e)	SWEM, WBQ-6		
Anschlussmodul(e)	FTOOP, PVA&NV, SWQ		
Bemerkungen	Dieses Modul vertieft und ergänzt das Modul SWE – Modellierung (SWEM).		

*Studienniveau	B Basic level course: Modul zur Einführung in das Basiswissen eines Gebiets. I Intermediate level course: Modul zur Vertiefung der Basiskenntnisse. 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

Leitidee

Software als ein zentrales Produktionsmittel vieler Industrien ist in ihrer Erstellung immer noch herausfordernd. Die Studierenden lernen daher die heute verfügbaren Mittel zur „industriellen“ – d.h. ingenieurmässigen – Erstellung von Software kennen. Dabei liegt der Fokus dieses Moduls SWEA auf dem Erarbeiten der Software-Architektur, der Schnittstellen zur Implementierung und der Produktionseinführung.

Die Studierenden sind damit in der Lage, systematisch und nachvollziehbar Lösungen basierend auf den Anforderungen und den gegebenen Randbedingungen herzuleiten. Insbesondere sind Ziele und Vorgehensweisen des Entwurfs der Software-Architektur inklusive der relevanten Rollen und Prozesse bekannt.

Die Querbeziehungen und Abhängigkeiten zu und von anderen Aktivitäten der Software-Entwicklung – vor allem dem Requirements Engineering – sind ihnen bekannt.

Inhalt

Dieses Modul dient als Vertiefungsmodul und legt den Schwerpunkt auf die „Entwurfsphasen“ der Software-Entwicklung: Architekturentwurf, Schnittstellen zur Implementierung und Produktionseinführung, sowie Betrieb.

Die „frühen Phasen“ setzen wir als bekannt voraus – sie wurden im Modul SWEM erarbeitet.

Die Studierenden eignen sich Kenntnisse über ein methodisches Vorgehen zum Entwurf einer Software-Architektur an. Sie können Methoden zum Implementieren und Deployen von Software in Software-Projekten anwenden.

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 Studierenden können ihre Erfahrungen – z.B. aus OOP und aus ihrem 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.

Architekturen und Muster

– „Die Lösung entwerfen und verstehen“

- Begriff und Bedeutung von Software-Architekturen
- Aufgaben und Verantwortung von Software-Architekten
- Methoden und Techniken zur Entwicklung von Software-Architekturen
- Muster, Frameworks und Werkzeuge

Implementieren

– „Die Lösung bauen“

- Schnittstellen zur Implementierung
- Modelle
- Vorgehensweisen, Rollen und Verantwortlichkeiten

Einführung der Software-Lösung

– „Die Lösung einsetzen“

- Vorgehen
- Outsourcing

Dieses Modul bereitet auch auf die Prüfung zur offiziellen iSAQB-Zertifizierung als *Certified Professional for Software-Architecture, Foundation Level (CPSA-F)* vor (<http://www.isaqb.org>).

Die Studierenden legen diese Zertifizierungsprüfung als Teil des Moduls ab.