

Modulbezeichnung (Kürzel)	Softwaretechnik (SWT)
Modulbezeichnung (eng.)	Software Engineering
Semester (Häufigkeit)	4 (jedes Sommersemester)
ECTS-Punkte (Dauer)	5 (1 Semester)
Art	Pflichtfach
Studentische Arbeitsbelastung	31 h Kontaktzeit + 119 h Selbststudium
Voraussetzungen (laut BPO)	
Empf. Voraussetzungen	Grundlagen der Programmierung 1
Verwendbarkeit	BIPV, BOMI, BOWI
Prüfungsform und -dauer	Klausur 2h
Lehr- und Lernmethoden	Online-Modul
Modulverantwortliche(r) (HSEL/VFH)	C. Wunck / S. Edlich (Beuth Hochschule für Technik Berlin)

Qualifikationsziele

Nach dem erfolgreichen Abschluss des Studienmoduls, sind die Studierenden in der Lage:

- softwaretechnische Kenntnisse in Projekte zu übertragen und anzuwenden
- Anforderungsermittlung und Verwaltung eigenständig durchzuführen
- informationstechnische Sachverhalte grafisch darzustellen
- tragfähige IT-Architekturen zu entwerfen und zu gestalten
- zu entscheiden und abzuwägen, wann welches Vorgehensmodell besser geeignet ist als ein anderes
- Requirements Engineering im Rahmen der Projektarbeit einzusetzen und zu erklären.
- die Hauptprobleme der Softwareentwicklung durch Analyse und Berücksichtigung der wichtigsten Anforderungsmerkmale zu identifizieren.
- in der Analyse - Pflichten- und Lastenheft, Use-Cases und Requirements einzuordnen und zu erstellen.
- den geeigneten Einsatz von UML zu beurteilen und UML praktisch an einem eigenen Projekt anzuwenden und die kritische Nutzung zu berücksichtigen.
- zu beurteilen welche UML-Diagramme in welcher Reihenfolge anzuwenden sind, um ein Modellierungsziel zu erreichen
- die Bedeutung der Architektur im Designprozess zu erklären, auf Projekte anzuwenden und zu begründen
- Werkzeuge für das systematische und objektorientierte Testen einzusetzen und selber Tests zu entwerfen
- die Möglichkeiten und Grenzen des Refactoring zu erklären und unter einer IDE anzuwenden
- die Funktionen des Buildmanagements mit ANT praktisch einzusetzen
- die Konzepte des Versions- und Fehlermanagements zu erklären und praxisnah zu verwenden
- die Bedeutung von Metriken als Qualitätsmaß praktisch zu beurteilen und Basismetriken zu berechnen
- Codemetriken und deren Werkzeuge zu gebrauchen, bspw. Architekturmetriken und deren Visualisierung
- das Entwurfsmuster Dependency Injection unter Verwendung unterschiedlicher Frameworks zu nutzen

Lehrinhalte

Einführung in die Softwaretechnik, Vorgehensmodelle / agile Modelle, Requirements Engineering, Analyse, Unified Modeling Language, Objektorientiertes Design, Objektorientierte Architekturen, Objektorientiertes Testen und Test-Driven Development, Refactoring, Buildmanagement, Versions- und Fehlermanagement, Software- und Architekturmetriken, Dependency Injection

Literatur

Balzert, Lehrbuch der Softwaretechnik
Oesterreich, Analyse und Design mit UML 2.1
Balzert, Lehrbuch der Objektmodellierung

Lehrveranstaltungen

Dozenten/-innen	Titel der Lehrveranstaltung	SWS
C. Wunck	Softwaretechnik	4