

Modulnummer	B18
Titel	Software Engineering II / Software Engineering 2
Leistungspunkte	8 LP
Workload	102 Stunden Präsenz (2 SWS SU + 4 SWS Ü), 138 Stunden Selbststudium
Lerngebiet	Fachspezifische Vertiefung
Lernziele / Kompetenzen	Die Studierenden lernen interaktive Software-Anwendungen auf der Grundlage einer Systemanalyse eigenständig zu entwerfen und zu implementieren. Dabei wenden Sie ausgewählte Architekturmuster und Entwurfskonzepte auf ein konkretes Beispiel an und vertiefen ihr Verständnis für das Zusammenspiel bereits bekannter und neuer Konzepte der objektorientierten Programmierung. Sie bauen wichtige Kompetenzen im Bereich Softwareentwurf und Software-Architektur (OOD) auf. Für die Umsetzung ihrer Entwürfe lernen die Studierenden aktuelle Modellierungs- und Programmiersprachen sowie relevante Entwicklungs-Werkzeuge und -Bibliotheken gezielt einzusetzen.
Voraussetzungen	Empfehlung: Software-Engineering I
Niveaustufe	4. Studienplansemester
Lehrform	Seminaristischer Unterricht, Übung
Status	Pflichtmodul
Häufigkeit des Angebotes	jedes Semester
Prüfungsform	Die Prüfungsform wird nach §19(2) RSPO durch die Lehrkraft festgelegt. Sofern die Lehrkraft die Prüfungsform und die Prüfungsmodalitäten nicht am Semesteranfang in der Frist nach §19 (2) RSPO festlegt, gilt folgende Prüfungsform: 50% Klausur (90 Minuten), 50% Projektarbeit (40 Stunden).
Ermittlung der Modulnote	Siehe Studienplan
Anerkannte Module	Module vergleichbaren Inhalts
Inhalte	<p>1. Software-Architekturen und Software-Entwurf (OOD):</p> <ul style="list-style-type: none"> • SW-Architekturen für verteilte und multimediale Anwendungen (Multi-Tier-Arch.) • Frameworks • Ebenen, Komponenten • Entwurfsprinzipien und -methoden, Entwurfsmuster • UML-Einsatz im Entwurf (Klassendiagramm, Interaktionsdiagramm, Zustandsdiagramm etc.) • Entwurfskommunikation <p>2. Implementierung (OOP):</p> <ul style="list-style-type: none"> • Transformation der Entwurfsmodelle • Abbildung der Architektur auf statische und dynamische Code-Bereiche (Pakethierarchien, Klassen, Algorithmen, etc.) • Nutzung von Frameworks und Bibliotheken für die Umsetzung des Entwurfs. z.B. Dependency-Injection-Container, OR-Mapper für die Persistenz <p>3. Versionierung, Konfigurationsmanagement</p> <p>In der Übung entwickeln die Studierenden, geführt durch konkrete Aufgaben, eine komplexe interaktive Anwendung. Definition und Planung der Entwurfsphase, der Entwurf sowie die Implementierung der Software werden hierbei explizit durchlaufen.</p>
Literatur	G. Starke: <i>Effektive Softwarearchitekturen: Ein praktischer Leitfaden</i> , Carl Hanser Verlag E. Freeman, K. Sierra, B. Bates: <i>Entwurfsmuster von Kopf bis Fuß</i> , O'Reilly H. Balzert: <i>Objektorientierte Programmierung mit Java 5</i> , Spektrum Akademischer Verlag
Weitere Hinweise	Dieses Modul wird auf Deutsch angeboten.