

Datenfeld	Erklärung
Modulnummer	B35
Titel	Modellgetriebene Softwareentwicklung (Model Driven Software Development)
Leistungspunkte	5 LP
Workload	3 SWS SU 1 SWS Ü 68 Stunden Präsenz 82 Stunden Selbststudium
Lerngebiet	Fachspezifische Grundlagen
Lernziele / Kompetenzen	Die Studierenden <ul style="list-style-type: none"> • kennen die grundlegenden Techniken der modellgetriebenen Softwareentwicklung (MDSO) für eingebettete Systeme. • kennen die wichtigsten Modellierungsaspekte der eingebetteten Softwaresysteme: Komponenten, Abhängigkeiten, funktionale Zuordnung, Zustände, zeitliche Kritikalität usw. • verstehen die Zuordnung zwischen den modellierten Softwareaspekten und dem aus dem Softwaremodell automatisch generierten Code und sind in der Lage, den generierten Code zu verwenden. • können aufgrund der Anforderungen an ein eingebettetes Softwaresystem die Notwendigkeit/Sinnhaftigkeit eines eventuellen MDSO-Einsatzes einschätzen. • beherrschen eine Auswahl von Werkzeugen und Modellierungssprachen für die Realisierung des MDSO im eingebetteten Softwareparadigma. <p>Fachunabhängig: Teamarbeit, Kommunikationsfähigkeit, Projektmanagement</p>
Voraussetzungen	Empfehlung: Fächer der Studienplansemester 1-5
Niveaustufe	6. Studienplansemester
Lehrform	Seminaristischer Unterricht und Projektlabor
Status	Pflichtmodul
Häufigkeit des Angebotes	Sommersemester
Prüfungsform	Die Prüfungsform wird nach §19 (2) RSPO durch die Lehrkraft festgelegt. Sofern die Lehrkraft die Prüfungsform nicht am Semesteranfang in der Frist nach §19 (2) RSPO festlegt, gilt folgende Prüfungsform: Klausur (90 Minuten)
Ermittlung der Modulnote	Siehe Studienplan
Verwendbarkeit	Module mit vergleichbarem Inhalt
Inhalte	<ul style="list-style-type: none"> • Modellierungssprachen, Grundlagen, Grundlegende Modellierungsparadigmen, Domain Specific Languages, Abstraktionsschichten der Modellierungs-formalismen (Modell, Metamodell, Metametamodell) • Automotive Softwareengineering: Spezifika und deren Auswirkungen auf die Softwareentwicklung • Grundlegende Modellierungssprachen, Beispielmodelle • DSLs, UML, SysML, Simulink, Hardwaremodellierung • Codegenerierung • Generierungstechniken, Werkzeuge, Fallbeispiel • Sicherheit • Funktionale Sicherheit, Risikoanalyse, Fehlertoleranz, statische Codeanalyse.
Literatur	Schäuffele, Zurawka: Automotive Software Engineering - Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen . Springer Vieweg Oliver Alt: Modellbasierte Systementwicklung mit SysML . Carl Hanser Verlag
Weitere Hinweise	Dieses Modul wird in deutscher oder englischer Sprache angeboten.
Raumbedarf	SU-Sem Ü-Lab