

Modulhandbuch für den Bachelor-Studiengang

Mobile und Eingebettete Systeme (MES)

nach Genehmigung durch den Prüfungsausschuss der Fakultät für Informatik und Mathematik am 22. Oktober 2014 und inklusive aller Beschlüsse des Prüfungsausschusses bis einschließlich 30. November 2016

Stand: 31. Januar 2017

Inhaltsübersicht:

442000 Modulgruppe Basistechnologien	Empf. Semester, Format, ECTS
442040 Grundlagen der Mensch-Maschine-Interaktion	3. Sem., 2V+2Ü, 6 ECTS
405530 Grundlagen der Elektrotechnik	2. Sem., 3V+2Ü, 7 ECTS
442020 Sensorik und Aktuatorik	3. Sem., 3V+2Ü, 7 ECTS
442030 Mustererkennung und Zeitreihenanalyse	4. Sem., 3V+1Ü, 6 ECTS
442010 Bildverarbeitung	4. Sem., 3V+2Ü, 7 ECTS

444000 Modulgruppe Mathematische Grundlagen	Empf. Semester, Format, ECTS
441010 Mathematik in Technischen Systemen I	1. Sem., 3V+2Ü, 7 ECTS
441020 Mathematik in Technischen Systemen II	2. Sem., 3V+2Ü, 7 ECTS
441030 Mathematik in Technischen Systemen III	3. Sem., 3V+2Ü, 7 ECTS
441040 Grundlagen der Dynamischen Systeme	5. Sem., 3V+2Ü, 7 ECTS

441000 Modulgruppe Praktische Informatik	Empf. Semester, Format, ECTS
443010 Grundlagen der Informatik für Mobile und Eingebettete Systeme	1. Sem., 4V+2Ü, 10 ECTS
443020 Softwaretechnik für Eingebettete Systeme	4. Sem., 4V+2Ü, 8 ECTS
443030 Programmierung in Java	2. Sem., 3Ü, 5 ECTS
405127 Algorithmen und Datenstrukturen	2. Sem., 3V+2Ü, 7 ECTS

443000 Modulgruppe Technische Informatik	Empf. Semester, Format, ECTS
413151 Technische Informatik	1. Sem., 3V+2Ü, 7 ECTS
405062 Rechnerarchitektur	2. Sem., 2V+1Ü, 5 ECTS
433755 Software-Hardware Codesign	3. Sem., 4Ü, 6 ECTS
405058 Rechnernetze	6. Sem., 3V+2Ü, 7 ECTS

445000 Modulgruppe Systems Engineering	Empf. Semester, Format, ECTS
445020 Complex Systems Engineering	5. Sem., 3V+2Ü, 7 ECTS
445010 Spezifikation und Verifikation von Eingebetteten Systemen	4. Sem., 3V+2Ü, 7 ECTS

446000 Module ohne Gruppenzuordnung	Empf. Semester, Format, ECTS
446100 Seminar zu Mobilien und Eingebetteten Systemen	4. Sem., 2S, 4 ECTS
446010 MES-Praktikum	5. Sem., 8P, 12 ECTS
448999 Präsentation der Bachelorarbeit	6. Sem., 3 ECTS

447000 Wahlpflichtmodule (Empfohlenes Semester: 5./6.Sem.)	Format, ECTS
454102 Einführung in die Kontextererkennung	3V+1Ü, 7 ECTS
479610 Embedded Systems Programming	4P, 7 ECTS
405166 Intelligent Audio Analysis	2V+1Ü, 5 ECTS
405211 Intelligent Systems (Blockseminar)	(4 Tage), 3 ECTS
405165 Intelligent Systems (Übung)	3Ü, 5 ECTS
407606 Numerische Methoden der Linearen Algebra	4V+2Ü, 9 ECTS
Praktikum für Mobile und Eingebettete Systeme (Externes, wissenschaftl. begleitetes Praktikum)	240h, 8 ECTS
405399 Praktikum Regelung und Robotik	1V+1Ü+2P, 7 ECTS
405281 Praktische Parallelprogrammierung	3V+2Ü, 7 ECTS
405026 Programming Applications for Mobile Interaction	3V+2Ü, 7 ECTS
405198 Software Product-Line Engineering	2V+2Ü, 6 ECTS

405206	Software Verification	2V+1Ü+2P, 7 ECTS
405156	Stochastische Simulation	3V+1Ü, 7 ECTS
405025	Text Mining Project	3V+3Ü, 8 ECTS
405002	Verteilte Systeme	2V+1Ü, 5 ECTS
405222	Wavelet-basierte Methoden in der Bildverarbeitung	2V+2Ü, 6 ECTS
431700	Web-Engineering	2V+2Ü, 6 ECTS

Schlüsselqualifikationen und Fremdsprache (Pflicht, Auswahl Wahlpflicht)

	Empf. Semester, Format, ECTS
Pflichtmodul FFA Aufbaustufenmodul 1	3. Sem., 2V, 3 ECTS
oder	
448100 Pflichtmodul FFA Aufbaustufenmodul 2 (542002 + 542003)	3. Sem., 2V, 3 ECTS
542001 Softskills für Informatiker (Wahlpflicht)	4 Tage Blockseminar, 3 ECTS
407450 Gewerblicher Rechtsschutz einschließlich Softwareschutz mit Fallstudien zu Patentrecht und Patentrecherche (Wahlpflicht)	2V, 3 ECTS

Weitere anrechenbare Schlüsselqualifikationen unter

www.fim.uni-passau.de/studium/pruefungen-und-modulkataloge/anrechenbarkeit-und-modulkataloge/

Muster-Studienplan

Semester	Modul	ECTS
1	Grundlagen der Informatik für Mobile und Eingebettete Systeme	10
	Technische Informatik	7
	Mathematik in Technischen Systemen I	7
	Schlüsselqualifikation	3
		27
2	Algorithmen und Datenstrukturen	7
	Grundlagen der Elektrotechnik	7
	Mathematik in Technischen Systemen II	7
	Rechnerarchitektur	5
	Programmierung in Java	5
		31
3	Sensorik und Aktuatorik	7
	FFA	3
	Mathematik in Technischen Systemen III	7
	Software-Hardware Codesign	6
	Grundlagen der Mensch-Maschine Interaktion	6
		29
4	Mustererkennung und Zeitreihenanalyse	6
	Softwaretechnik für Eingebettete Systeme	8
	Spezifikation und Verifikation von Eingebetteten Systemen	7
	Seminar	4
	Bildverarbeitung	7
		32
5	Grundlagen der Dynamischen Systeme	7
	Complex Systems Engineering	7
	MES-Praktikum	12
	Wahlpflichtmodul	6
		32
6	Bachelorarbeit und Präsentation der Bachelorarbeit	15
	Rechnernetze	7
	Wahlpflichtmodul	7
		29

Modulgruppe Basistechnologien

Modulbezeichnung	Grundlagen der Mensch-Maschine-Interaktion
ggf. Kürzel	MMI
Empfohlenes Studiensemester	3
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r)	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Dozent(in)	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Sprache	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Basistechnologien
Lehrform/SWS	2V+2Ü
Arbeitsaufwand	60 Std. Präsenz + 60 Std. Übungsaufgaben + 60 Std. Nachbearbeitung des Vorlesungsstoffes
ECTS-Leistungspunkte	6
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen	keine
Angestrebte Lernergebnisse	<p>Kenntnisse: Die Studierenden kennen grundlegende Interaktionskonzepte und -modelle mit Schwerpunkt auf grafischen Benutzungsoberflächen. Die Studierenden kennen die Grundzüge der menschlichen Wahrnehmung und Informationsverarbeitung mit Bezug zur Mensch-Maschine-Interaktion. Die Studierenden kennen Entwurfsmethodiken und Bewertungsansätze für Benutzungsschnittstellen.</p> <p>Fähigkeiten: Die Studierenden können Benutzungsoberflächen beschreiben, analysieren und diskutieren. Sie sind in der Lage einfache Studien zur Evaluation von Benutzungsoberflächen zu entwerfen, durchzuführen und deren Ergebnisse zu bewerten. Die Studierenden können mit Hilfe einfacher und schneller Verfahren (Prototyping) Entwürfe von Benutzungsoberflächen erstellen und diese mit geeigneten Methoden und Benutzern untersuchen.</p> <p>Kompetenzen: Die Studierenden können einfache grafische Benutzungsoberflächen erstellen, vorhandene Benutzungsoberflächen analysieren und</p>

	<p>diskutieren. Die Studierenden können die Qualität von Benutzungsoberflächen und Interaktionsmodellen in einfach gelagerten Fällen evaluieren und die Ergebnisse kritisch beurteilen. Die Studierenden kennen grundlegende Ansätze zum Entwurf von Benutzungsoberflächen und -schnittstellen.</p>
Inhalt	<p>Grundlagen der menschlichen Wahrnehmung und Informationsverarbeitung Grundlagen der Interaktionsmodelle Grundlegende Konzepte grafischer Benutzungsoberflächen Heuristiken, Richtlinien und formale Modelle für die Entwicklung von Benutzungsschnittstellen Grundlegende theoretische Modelle der Mensch-Maschine-Interaktion (z.B. Fitts's Law, Hick's Law, GOMS, KLM) Prototyping-Methoden und Vorgehensweisen der Prototypenerstellung und -bewertung Entwurf, Durchführung und Auswertung von Benutzerstudien Evaluation von Benutzungsschnittstellen Statistische Grundlagen für die Auswertung von Benutzerstudien und der Beurteilung der Ergebnisse Praktische Anwendung der Inhalte in den Präsenzübungen und sowie selbständige Vertiefung durch Bearbeitung der Übungsaufgaben</p>
Studien-/Prüfungsleistungen	<p>90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten) oder Portfolio-Prüfung; die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang auf den Internetseiten der Fakultät bzw. in der Vorlesung bekannt gegeben Mögliche Portfoliobestandteile sind</p> <ul style="list-style-type: none"> - Eigenständige Zusammenfassung von relevanten wissenschaftlichen Arbeiten zu den Themen der Lehrveranstaltung - Eigenständige Beschreibung von Benutzungsschnittstellen, insbesondere deren Benutzungskontext, z.B. durch Handzeichnungen (Sketches), Storyboards oder Videoprototypen, und vergleichbaren Ansätzen von Abläufen im Kontext der Mensch-Maschine Interaktion - Erstellung von Personas, Use Cases und Szenarios - Anfertigung von geeigneten Prototypen und Entwürfen für Benutzungsschnittstellen für vorgegebene Anwendungskontexte und deren Dokumentation - Entwurf, Durchführung, Dokumentation und Auswertung kleiner Benutzerstudien - Präsentation der erstellten Materialien unter Einsatz geeigneter Präsentationstechniken, z.B. PowerPoint, Desktoppreviews, Postern, Flipchart, Whiteboard, Tafel - Abgabe der einzelnen Bestandteile des Portfolios in einer einzelnen Mappe bzw. Arbeit

	Die genauen Anforderungen werden vom Dozenten zu Beginn der Veranstaltung bekanntgegeben.
Modulnote	Modulnote entspricht der Note der Prüfung
Medienformen	Präsentation mit Projektor, Tafelanschrieb, Gruppenarbeit
Literatur	Wird vom Dozent / von der Dozentin bekannt gegeben

Modulbezeichnung:	Sensorik und Aktuatorik
ggf. Kürzel	
Empfohlenes Studiensemester:	3
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Dozent(in):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Basistechnologien
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagen der Elektrotechnik
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die Basistypen der Sensoren und der Aktuatoren</p> <p>Fähigkeiten: Die Studierenden können die einzelnen Bausteine zu Typen zuordnen und ihre Wirkprinzipien sowie relevante Fehlerquellen erklären</p> <p>Kompetenzen: Die Studierenden verfügen über Grundverständnis der physikalischen Wirkprinzipien einzelner Sensoren, Aktuatoren und Wandlungsbausteine. Sie können diese modellieren, relevante Fehlerquellen identifizieren, klassifizieren und mittels Fehlermodellen analysieren</p>
Inhalt:	<p>Physikalische Grundlagen wichtigster Sensoren und Aktuatoren (Beschleunigungssensor, Gyroskop, Ultraschallsensor, LIDAR)</p> <p>Kapazitive und induktive, aktive und passive Bausteine</p> <p>Sensormodellierung, Fehlerarten und Fehlermodelle</p> <p>Analog-digital und digital-analog Wandlung</p>
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung

Medienformen:	Projektor und Wandtafel
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Mustererkennung und Zeitreihenanalyse
ggf. Kürzel	
Empfohlenes Studiensemester:	4
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Dozent(in):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Basistechnologien
Lehrform/SWS:	3V+1Ü
Arbeitsaufwand:	60 Std. Präsenz + 40 Std. Übungsaufgaben + 80 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	6
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen III, Grundlagen der Informatik für Mobile und Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die grundlegenden überwachten und unüberwachten Lernverfahren und die wesentlichen Modelle und Methoden zur Zeitreihenanalyse</p> <p>Fähigkeiten: Die Studierenden können Algorithmen zur Analyse von Zeitreihen und zur Mustererkennung in Software implementieren.</p> <p>Kompetenzen: Die Studierenden können für gegebene Fragestellungen adäquate Analysemethoden aus einem Portfolio von überwachten und unüberwachten Verfahren identifizieren und auf der Basis geeigneter Modellierung programmtechnisch umsetzen</p>
Inhalt:	<p>Überwachte statistische Klassifikation: Bayes-Klassifikatoren, lineare Diskriminanten, Support Vector Machines, Neuronale Netze, Baumklassifikatoren</p> <p>Unüberwachtes Lernen: Expectation Maximization, Clustering</p> <p>Zeitreihenanalyse: Markov-Modelle, Dynamic Time Warping, polynomielle Approximation</p>
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt

	gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Projektor und Wandtafel
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Bildverarbeitung
ggf. Kürzel	
Empfohlenes Studiensemester:	4
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Tomas Sauer (LST für Mathematik mit SP Digitale Bildverarbeitung)
Dozent(in):	Prof. Dr. Tomas Sauer (LST für Mathematik mit SP Digitale Bildverarbeitung)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Basistechnologien
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen I, III und (event. parallel zur Veranstaltung) Mustererkennung und Zeitreihenanalyse
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die grundlegenden Konzepte der Signalverarbeitung und Verfahren zur Bildverarbeitung und wissen, wie diese hergeleitet werden und wie ihre Korrektheit bewiesen wird.</p> <p>Fähigkeiten: Die Studierenden können Verfahren zur Bildverarbeitung implementieren, modifizieren und in gewissem Rahmen auch neu entwickeln. Außerdem können sie verschiedene Algorithmen vergleichen, bewerten und auf Korrektheit untersuchen.</p> <p>Kompetenzen: Die Studierenden haben die Kompetenz, mit Signal- und Bildverarbeitungsalgorithmen theoretisch und praktisch umzugehen.</p>
Inhalt:	Mathematische Grundlagen: Signalverarbeitung, FFT, Transformationen und Optimierung. Entrauschen, Kompression, Feature Detection, Bildregistrierung, Impainting.
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben

Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Grundlagen der Elektrotechnik
ggf. Kürzel	
Empfohlenes Studiensemester:	2
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Dozent(in):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Basistechnologien
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Technische Informatik, Mathematik in Technischen Systemen I
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die physikalischen Grundlagen der Elektro- und Informationstechnik. Sie verfügen über Basiskenntnisse von Gleich- und Wechselstromkreisen und von den wesentlichen elektronischen Bauelementen.</p> <p>Fähigkeiten: Die Studierenden können ausgewählte Bauelemente und einfache Schaltungen auf ihrer Basis mathematisch beschreiben und analysieren. Sie sind in der Lage, einfach gelagerte Entwurfsaufgaben (z.B. Dimensionierung in einfachen elektronischen Netzwerken) durchzuführen. Sie können Kenngrößen periodischer Signale berechnen.</p> <p>Kompetenzen: Die Studierenden durchdringen die Prinzipien der analogen und digitalen Informationsdarstellung. Sie verstehen die wesentlichen physikalischen Wirkprinzipien von elektronischen Schaltungen und haben die Kompetenz, einfache Schaltungen selbständig zu entwickeln, mathematisch zu beschreiben und zu analysieren.</p>
Inhalt:	<p>Physikalische Grundlagen: Strom, Spannung, Arbeit, Leistung, Ohmsches Gesetz</p> <p>Kirchhoffsche Gesetze, Netzwerktheoreme, Basisschaltungen</p> <p>Grundlagen der elektronischen Bauelemente</p>

Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Projektor und Wandtafel
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben
Sonstiges:	ersetzt im B.Sc. MES PO 2012 gleichwertig das Modul Grundlagen der Informationstechnik, 3V+2UE, 7 ECTS

Modulgruppe Mathematische Grundlagen

Modulbezeichnung:	Mathematik in Technischen Systemen I
ggf. Kürzel	
Empfohlenes Studiensemester:	1
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme)
Dozent(in):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme), Dr. Cornelia Schwarz
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Mathematische Grundlagen
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	
Angestrebte Lernergebnisse:	<p>Kenntnisse: die Studierenden sind vertraut mit elementarer Aussagenlogik, naiver Mengenlehre und Grundkonzepten der Kombinatorik. Sie kennen die wesentlichen Ergebnisse und Methoden der Analysis von Funktionen einer Veränderlichen sowie die Basiskonzepte der Linearen Algebra.</p> <p>Kompetenzen: Die Studierenden beherrschen die grundlegenden Rechen- und Beweisverfahren der Analysis und der Linearen Algebra und können diese selbständig in Standardsituationen anwenden.</p>
Inhalt:	<p>Grundlagen: natürliche Zahlen, vollständige Induktion, reelle Zahlen, Ungleichungen, komplexe Zahlen. Kombinatorische Abzählregeln. Folgen und Reihen: Konvergenzkriterien. Reelle Funktionen: Abbildungseigenschaften, Polynome, rationale Funktionen, Grenzwerte, Stetigkeit, Zwischenwertsatz, Exponentialfunktion, trigonometrische Funktionen, Wurzelfunktionen. Differentialrechnung: Rechenregeln, Extrema, Umkehrfunktionen, Logarithmus, Potenzfunktion. Integralrechnung: Integrale, Hauptsatz der Differential- und Integralrechnung, Mittelwertsätze, Partielle Integration,</p>

	Substitutionsregel, Uneigentliche Integrale. Der Vektorraum \mathbb{R}^n , Skalarprodukt, Norm, Vektorprodukt, Gerade und Ebenen. Vektorräume: Lineare (Un-)Abhängigkeit, Dimension, Basis. Lineare Abbildungen, Matrizen, Lineare Gleichungssysteme, Gauß-Algorithmus, Rang einer Matrix.
Studien-/Prüfungsleistungen:	120-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Mathematik in Technischen Systemen II
ggf. Kürzel	
Empfohlenes Studiensemester:	2
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme)
Dozent(in):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme), Dr. Cornelia Schwarz
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Mathematische Grundlagen
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen I
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden besitzen vertiefte Kenntnisse der Matrizenrechnung und kennen die wesentlichen Begriffe und Methoden der Analysis von Funktionen mehrerer reeller Veränderlicher und der mehrdimensionalen Integration.</p> <p>Kompetenzen: Die Studierenden sind in der Lage, grundlegende Aufgabenstellungen, bei denen lineare Abbildungen oder Differentiation und Integration von Funktionen mehrerer reeller Veränderlicher eine Rolle spielen, zu bearbeiten. Insbesondere beherrschen sie die Grundkonzepte für die Lösung von Optimierungsaufgaben und Eigenwertproblemen.</p>
Inhalt:	Lineare Algebra: Determinanten, Eigenwerte. Funktionenfolgen und -reihen: Taylorreihen, Fourierreihen, Differentialrechnung im \mathbb{R}^n : partielle Ableitungen, Richtungsableitungen Taylorscher Satz, Kettenregel, Extrema, inverse und implizite Funktionen, Extrema unter Nebenbedingungen. Kurvenintegrale: Kurven im \mathbb{R}^n , Gradientenfelder. Integration im \mathbb{R}^n : Substitutionsregel, Satz von Fubini, Oberflächenintegrale, Flächen, Integralsätze.

Studien-/Prüfungsleistungen:	120-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Mathematik in Technischen Systemen III
ggf. Kürzel	
Empfohlenes Studiensemester:	3
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme)
Dozent(in):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme), Dr. Cornelia Schwarz
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Mathematische Grundlagen
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen I, II
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden besitzen Grundkenntnisse der Theorie gewöhnlicher Differentialgleichungen. Sie verstehen die Basiskonzepte und zentralen Ergebnisse der Stochastik im Rahmen einfacher Modelle und kennen Standardverfahren aus der Numerik.</p> <p>Kompetenzen: Sie sind in der Lage, zeitabhängige Phänomene mit gewöhnlichen Differentialgleichungen zu modellieren. Sie können einfache zufällige Phänomene modellieren und statistisch analysieren. Sie besitzen die Fähigkeit, numerische Verfahren zur approximativen Lösung von Standardproblemen in der Linearen Algebra und in der Analysis einzusetzen.</p>
Inhalt:	<p>Gewöhnliche Differentialgleichungen: Elementare Lösungstechniken, Systeme linearer Differentialgleichungen mit konstanten Koeffizienten .</p> <p>Statistik: Wahrscheinlichkeitsraum und Zufallsvariable, Erwartungswert und Varianz, Grenzwertsätze, Schätzverfahren, Konfidenzintervalle und Tests.</p>

	Numerik: Verfahren für Lineare Gleichungssysteme, Eigenwerte und Eigenvektoren. Interpolation und Approximation, numerische Integration, Numerische Verfahren für gewöhnliche Differentialgleichungen.
Studien-/Prüfungsleistungen:	120-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Grundlagen der Dynamischen Systeme
ggf. Kürzel	
Empfohlenes Studiensemester:	5
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester (nach neuem Studienplan)
Modulverantwortliche(r):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme)
Dozent(in):	Prof. Dr. Fabian Wirth (LST für Mathematik mit SP Dynamische Systeme)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Mathematische Grundlagen
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen I, II, III
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden besitzen vertiefte Kenntnisse in der Modellierung und mathematischen Analyse dynamischer Prozesse.</p> <p>Kompetenzen: Die Studierenden sind in der Lage, zeitdiskrete und zeitstetige dynamische Phänomene aus naturwissenschaftlichen und technischen Bereichen mathematisch zu modellieren, zu simulieren und auf ihre wesentlichen Eigenschaften wie Langzeitverhalten, Stabilität, etc. zu untersuchen.</p>
Inhalt:	Definition und wichtige Beispiele dynamischer Systeme, asymptotisches Verhalten und Stabilität von Orbits, Fixpunkte, periodische Lösungen, Limesmengen, Invarianzprinzipien, Bifurkationen, Chaos, Anwendungen (z.B. lineare und nichtlineare Zeitreihenanalyse)
Studien-/Prüfungsleistungen:	120-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben

Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulgruppe Praktische Informatik

Modulbezeichnung:	Grundlagen der Informatik für Mobile und Eingebettete Systeme
ggf. Kürzel	Gdl für MES
Empfohlenes Studiensemester:	1
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Dozent(in):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Praktische Informatik
Lehrform/SWS:	4V+2Ü
Arbeitsaufwand:	60 Std. Präsenz in der Vorlesung + 60 Std. Nachbearbeitung des Vorlesungsstoffes + 180 Std. Übungsaufgaben (30 Std. Präsenz + 150 Stunden Nachbearbeitung)
ECTS-Leistungspunkte:	10
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen:	Keine
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen grundlegende Begriffe und Konzepte der allgemeinen Informatik. Sie lernen elementare Datentypen, Datenstrukturen und Algorithmen kennen. Die Studierenden erlernen die Grundkonzepte imperativer Programmiersprachen. Die Studierenden kennen grundlegende Konzepte von Betriebssystemen. Die Studierenden haben erste Grundkenntnisse von Objektorientierung.</p> <p>Fähigkeiten: Die Studierenden können einfache Programme in der Programmiersprache C/C++ erstellen. Die Studierenden können elementare Algorithmen und Datenstrukturen implementieren und einfache Anwendungsbeispiele zu Such- und Sortierverfahren umsetzen.</p> <p>Kompetenzen: Die Studierenden erkennen in algorithmischen Fragestellungen Konzepte und Strukturen und können diese bei der</p>

	Programmierung geeignet umsetzen.
Inhalt:	<p>Grundbegriffe: Algorithmus, Datentyp, Datenstruktur, Syntax, Semantik, Zahldarstellung, Codierung</p> <p>Elementare Algorithmen und Datenstrukturen, insbesondere zur Suche und zur Sortierung</p> <p>Grundkonzepte: Induktion, Rekursion, imperative und objektorientierte Programmiersprachen, Komplexität und Laufzeit von Algorithmen</p> <p>Grundprinzipien der Programmierung</p> <p>Grundlagen von Betriebssystemen</p> <p>Vertiefte Einführung in die Programmiersprache C/C++, gemeinsame Erarbeitung der Grundlagen in den Präsenzübungen sowie selbständige Vertiefung durch Bearbeitung der Übungsaufgaben</p>
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation mit Projektor, Tafelanschrieb
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Softwaretechnik für Eingebettete Systeme
ggf. Kürzel	
Empfohlenes Studiensemester:	4
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) oder Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Dozent(in):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) oder Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Praktische Informatik
Lehrform/SWS:	4V+2Ü
Arbeitsaufwand:	90 Std. Präsenz + 90 Std. Übungsaufgaben + 60 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	8
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagen der Informatik für Mobile und Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen weiterführende Programmier Techniken, insbesondere imperative und objektorientierte Techniken. Sie kennen ferner Grundbegriffe der Softwaretechnik und des Projektmanagements, können Phasen eines Softwareprojekts nach unterschiedlichen Softwareprozessmodellen benennen und einordnen.</p> <p>Fähigkeiten: Die Studierenden haben das programmiertechnische Rüstzeug, welches für die Entwicklung größerer Softwaresysteme unerlässlich ist. Sie sind mit den wesentlichen Aspekten des Programmierstils, der Analyse, Modellierung, Dokumentation, Wartung und der Qualitätssicherung vertraut.</p> <p>Kompetenzen: Die Studierenden sind in der Lage, an größeren Softwareprojekten teilzunehmen. Sie können die Komplexität solcher Projekte erkennen und durch die Anwendung geeigneter strukturierter Instrumente beherrschbar machen.</p>

Inhalt:	<p>Weiterführende Themen der Programmierung: Abstraktion, Divide & Conquer, Objektorientierung, Generisches Programmieren</p> <p>Fortgeschrittene Programmieretechniken: Fehlerbehandlung, Ein-/Ausgabe, Hardware-nahe Programmierung</p> <p>Programmierstil</p> <p>Softwaretechnik: Anforderungsanalyse, Modellierung, Softwarearchitektur, Softwareprozessmodelle, Dokumentation, Agile Development, Projektmanagement</p> <p>Qualitätssicherung, Automatisches Testen</p>
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Programmierung in Java
ggf. Kürzel	
Empfohlenes Studiensemester:	2
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems)
Dozent(in):	Shahin Amiriparian, M.Sc.
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Modulgruppe Praktische Informatik
Lehrform/SWS:	3Ü
Arbeitsaufwand:	45 Std. betreute Übungsaufgaben, 105 Stunden selbständige Vor- und Nachbereitung
ECTS-Leistungspunkte:	5
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagen der Informatik für Mobile und Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die Syntax und Semantik von Java, etablierte Programmiermuster, wichtige Bibliotheksfunktionen und die Bedienung der relevanten Entwicklungswerkzeuge</p> <p>Fähigkeiten: Die Studierenden können einfache Software in der Programmiersprache Java entwickeln</p> <p>Kompetenzen: Die Studierenden haben die Kompetenz, die beiden Programmiersprachen C/C++ und Java miteinander zu vergleichen und die geeignete Programmiersprache für die Lösung eines gegebenen algorithmischen Problems auszuwählen.</p>
Inhalt:	Der Kurs richtet sich an Studierende, die das imperative und objektorientierte Programmieren mit C/C++ bereits beherrschen. Im Kurs soll auf der Basis der C/C++-Programmierung Java als die zweite Programmiersprache erarbeitet werden. Außerdem sollen die Studierenden die relevanten Entwicklungswerkzeuge (Editor, integrierter Debugger, Profiler etc.) praktisch einsetzen.
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben.

Modulnote:	Die Modulnote entspricht der Note der Präsentation.
Medienformen:	Präsentation und Beamer
Literatur:	C. Ullenboom, Java ist auch eine Insel - Galileo Computing Verlag, 2014 E. Jung, Java 8 - Das Übungsbuch, mitp Verlag, 2014 D. Logofatu, Grundlegende Algorithmen mit Java, Springer Vieweg Verlag, 2014 J. Goll, C. Heinisch, Java als erste Programmiersprache - Springer Vieweg Verlag, 2014

Modulbezeichnung:	Algorithmen und Datenstrukturen
ggf. Kürzel	5200
Empfohlenes Studiensemester:	2
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	N.N. (LST für Informatik mit SP Theoretische Informatik)
Dozent(in):	N.N. (LST für Informatik mit SP Theoretische Informatik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Praktische Informatik
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagen der Informatik für Mobile und Eingebettete Systeme, Programmierung in Java
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen anschließend die grundlegenden Datenstrukturen und elementare Algorithmen.</p> <p>Fähigkeiten: Die Studierenden können die erlernten Datenstrukturen und Algorithmen anwenden und in Programme umsetzen.</p> <p>Kompetenzen: Sie sind in der Lage, Algorithmen zur Lösung von für die Informatik typischen Problemen zu entwickeln und diese zu bewerten.</p>
Inhalt:	<p>Algorithmen sind die Grundlage zur Lösung von Problemen mit Programmen. Sie sind exakt formulierte Verfahren zur Bearbeitung von Daten. In der Informatik nehmen Algorithmen die zentrale Rolle ein.</p> <p>Inhalte der Vorlesung sind Sortierverfahren, lineare Datenstrukturen, wie Arrays, Listen, sowie Suchbäume, Verfahren für die Verwaltung von Mengen und grundlegende Graphenalgorithmen. Darüber hinaus werden Prinzipien zur Konstruktion von Algorithmen vorgestellt, wie Greedy Verfahren, Divide & Conquer und systematisches Suchen. Ein wesentlicher Bestandteil ist die Bewertung der Algorithmen nach ihrer Effizienz und die daraus abgeleitete Komplexität von Problemen einschließlich NP-harter Probleme.</p>

Studien-/Prüfungsleistungen:	90 min Klausur
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation mit Beamer und Tafel
Literatur:	<p>Cormen, Leiserson, Rivest, Stein: Introduction to Algorithms, 2001</p> <p>T. Ottmann P. Widmayer: Algorithmen und Datenstrukturen, Spektrum Verlag 2000</p> <p>Vorlesungsunterlagen</p>

Modulgruppe Technische Informatik

Modulbezeichnung:	Technische Informatik
ggf. Kürzel	5105
Empfohlenes Studiensemester:	1
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Ilia Polian (LST für Technische Informatik)
Dozent(in):	Prof. Dr. Ilia Polian (LST für Technische Informatik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Technische Informatik
Lehrform/SWS:	3V + 2Ü
Arbeitsaufwand:	75 Std. Präsenz + 70 Std. Übungsaufgaben + 65 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	keine
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen Komponenten von einfachen und komplexeren Digitalschaltungen und die grundlegenden Synthese- und Qualitätssicherungsverfahren kennen.</p> <p>Fertigkeiten: Sie können mit den grundlegenden Formalismen wie Boolesche Algebra, endliche Zustandsautomaten, Binäre Entscheidungsdiagramme, Zeichenströme umgehen und Schaltnetze und Schaltwerke aus entsprechenden formalen Beschreibungen synthetisieren und optimieren.</p> <p>Kompetenzen: Sie können unterschiedliche Arten der Information digital darstellen und zur Informationsverarbeitung geeignete digitale Schaltungen spezifizieren und entwerfen, testen und ihr Zeitverhalten analysieren.</p>
Inhalt:	<p>Informationsdarstellung: Zahlendarstellung (Ganzzahlen, Festkommazahlen, Zweierkomplement), Zeichendarstellung, fehlererkennende und -korrigierende Kodierungen.</p> <p>Boolesche Funktionen: Grundbegriffe, Normalformen, Umsetzung durch programmierbare logische Felder, Berechnung des Minimalpolynoms durch Verfahren von Quine-McCluskey, Binäre Entscheidungsdiagramme.</p> <p>Kombinatorische Schaltkreise (Schaltnetze): Logikgatter, Hierarchie, arithmetische Schaltkreise, ALU, Einführung in</p>

	<p>kombinatorische Synthese und Verifikationsverfahren.</p> <p>Sequentielle Schaltkreise (Schaltwerke): Speicherelemente, Zustandsautomaten und ihre Äquivalenz zu sequentiellen Schaltkreisen, Zustandsminimierung, Einführung in sequentielle Synthese, Speicherfelder und Busse.</p> <p>Analyse des Zeitverhaltens von kombinatorischen und sequentiellen Bausteinen.</p> <p>Entwurf und Programmierung eines einfachen Mikroprozessors, Analyse und Optimierung seines Zeitverhaltens.</p> <p>Qualitätssicherung und Testverfahren: Fehlermodellierung, Fehlersimulation, Grundlagen der Automatischen Testmuster-generierung, prüfgerechter Entwurf.</p>
Studien-/Prüfungsleistungen:	90 min Klausur
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation mit Beamer, Simulationswerkzeuge.
Literatur:	<p>Becker, Molitor, "Technische Informatik: Eine einführende Darstellung" (Oldenbourg, 2008).</p> <p>Keller, Paul, "Hardware Design: Formaler Entwurf digitaler Schaltungen" (Teubner, 2005).</p> <p>Eggersglüß, Fey, Polian, "Test digitaler Schaltkreise" (De Gruyter Oldenbourg, 2014).</p> <p>Folienkopien.</p>

Modulbezeichnung:	Rechnerarchitektur
ggf. Kürzel	5204
Empfohlenes Studiensemester:	2
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Ilia Polian (LST für Technische Informatik)
Dozent(in):	Prof. Dr. Ilia Polian (LST für Technische Informatik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Technische Informatik
Lehrform/SWS:	2V+1Ü
Arbeitsaufwand:	45 Std. Präsenz + 45 Std. Übungsaufgaben + 60 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	5
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	keine
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen Komponenten von Rechnern, den internen Aufbau eines Prozessors, sein Zusammenwirken mit der Anwendungssoftware und mit Betriebssystemkomponenten mittels Befehlssatz und seine Interaktion mit Speicherbausteinen.</p> <p>Fertigkeiten: Sie können Informationsverarbeitung durch programmierbare Rechner am Beispiel des Mikroprozessors MIPS nachvollziehen, die Performanz der Rechner und ihrer Komponenten systematisch bewerten, haben Grundkenntnisse über Programmierung in Maschinensprache und ihren Zusammenhang mit Hochsprachen-Konstrukten sowie die Hierarchie unterschiedlicher Typen von Speichern</p> <p>Kompetenzen: Sie sind in der Lage, für eine gegebene algorithmische Aufgabe ihre Abarbeitung durch die Prozessor-Hardware in Interaktion mit einer Speicherhierarchie einzuordnen, einen adäquaten Rechnertyp und die benötigte Leistungsfähigkeit seiner Komponenten mit den dafür geeigneten Metriken zu spezifizieren.</p>
Inhalt:	<p>Einführender Überblick über Hardwareentwurf und Fertigung</p> <p>Metriken zur Performanzbewertung</p> <p>Befehlssatz und Schnittstelle mit der Software</p>

	<p>Interner Aufbau eines Prozessors, Maßnahmen zur Leistungssteigerung</p> <p>Speicher, Speicherhierarchie</p> <p>Multiprozessoren, spezielle Architekturen</p>
Studien-/Prüfungsleistungen:	90 min Klausur
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation mit Beamer, Simulationsprogramme
Literatur:	<p>J. L. Hennessy, D. A. Patterson, "Computer Organization and Design: The Hardware/Software Interface", Morgan Kaufmann, 5. Ausgabe, 2014.</p> <p>Folienkopien.</p>

Modulbezeichnung:	Software-Hardware Codesign
ggf. Kürzel	
Empfohlenes Studiensemester:	3
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Ilia Polian (LST für Technische Informatik)
Dozent(in):	Prof. Dr. Ilia Polian (LST für Technische Informatik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Technische Informatik
Lehrform/SWS:	4Ü
Arbeitsaufwand:	60 Std. Präsenz + 120 Std. Vor- und Nachbearbeitung der Versuche
ECTS-Leistungspunkte:	6
Voraussetzungen nach Prüfungsordnung	<p>Erfolgreiche Absolvierung der Pflichtmodule „Technische Informatik“, „Rechnerarchitektur“</p> <p>Begründung: In der Übung sollen die Studierenden in mehreren größeren (über mehrere Sitzungen hinweg durchgeführten) Versuchen in Teams die theoretisch erlangten Kenntnisse praktisch anwenden und die Abläufe und die einzelnen Arbeitsphasen eines größeren, zusammenhängenden Projekts kennen lernen. Zur Durchführung dieser Übung werden spezielle Labore mit aufwändiger Technik und mit begrenzter Kapazität verwendet, was eine natürliche Begrenzung der Teilnehmerzahl erfordert. Die Anforderung von bestimmten fachlichen Mindestkompetenzen, die durch bestandene Prüfungen der angegebenen Grundlagenmodule nachzuweisen sind, ist für eine erfolgreiche Übungsteilnahme eine Basisvoraussetzung. Die Zulassung von Studierenden ohne diese Mindestvoraussetzungen würde sowohl ihren Studienerfolg als auch den Studienerfolg ihrer Teammitglieder, die sich auf eine erfolgreiche Bearbeitung aller Versuchsteile verlassen müssen, in Frage stellen.</p>
Empfohlene Voraussetzungen:	Keine
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen die Synthese der (insbesondere programmierbaren) Hardware auf FPGA kennen, bekommen Programmierung auf den so entworfenen eingebetteten Prozessoren vermittelt und führen Entwurfsraumexploration durch.</p> <p>Fertigkeiten: Sie üben den Gebrauch der industriellen</p>

	<p>Werkzeuge zur FPGA-Synthese ein. Sie lernen, die so entstehenden Mikroprozessoren zu benutzen und zu programmieren.</p> <p>Kompetenzen: Sie lernen eine konkrete Ausprägung eines komplexen Entwurfsablaufs im Detail kennen, der sowohl bei der Produktentwicklung als auch beim Prototyping eine Schlüsselrolle einnimmt. Sie können einfache Hardwareblöcke durch gängige Beschreibungssprachen spezifizieren und unter Verwendung von aktuellen FPGA-Synthesewerkzeugen auf FPGA-Plattformen abbilden. Sie können auf FPGA eingebettete Mikroprozessoren synthetisieren und hardwarenah programmieren. Sie sind in der Lage, externe Geräte (Sensoren und Aktuatoren) anzusteuern. Sie können die Charakteristika der entstehenden Systeme bewerten und gegeneinander abwägen (Entwurfsraumexploration).</p>
Inhalt:	<p>Mehrere Versuche zur selbständigen Durchführung in kleinen Gruppen. Versuche können sich über mehrere Sitzungen erstrecken</p> <p>Spezifikation von digitalen Schaltungen durch Hardwarebeschreibungssprachen (z.B. VHDL, Verilog, SystemC); Simulation und Synthese der Schaltungen auf einem FPGA, Ansteuerung von Sensoren und Aktuatoren</p> <p>Synthese eines Mikroprozessors auf dem FPGA und hardwarenahe Programmierung des Mikroprozessors mit C/C++; Ansteuerung von Schnittstellen, Sensoren und Aktuatoren durch den Mikroprozessor</p> <p>Entwurfsraumexploration: Aufteilung der Systemfunktionalität auf Software (Mikroprozessor) und Hardware (direkt auf den FPGA-Baustein synthetisiert); Kommunikation zwischen den synthetisierten Hardwareblöcken und dem Mikroprozessor</p>
Studien-/Prüfungsleistungen:	Praktische Leistung bei der selbständigen Erarbeitung und Vorführung der Versuche.
Modulnote:	Die Modulnote entspricht der Note der Prüfung.
Medienformen:	Präsentation mit Beamer, Vorführung von unterschiedlichen Werkzeugen, Simulation
Literatur:	Folienkopien, Dokumentation der Werkzeuge
Anwesenheitspflicht:	<p>Für die Übung besteht Anwesenheitspflicht.</p> <p>Begründung: Die Übung findet im wöchentlichen Rhythmus in einem dafür vorgesehenen Labor mit der nötigen Ausstattung statt. Die Studierenden führen in Teams mehrere Versuche durch, die sich jeweils über mehrere Sitzungen erstrecken. Die Versuche sind teilweise so geartet, dass sie nur im Labor unter</p>

	<p>der Verwendung von aufwändiger proprietärer Hardware und/oder kommerziellen Software-Entwurfswerkzeugen durchgeführt werden können. Es ist unrealistisch, einen völlig flexiblen Zugang der Studierenden ins Labor zu ermöglichen, da dies eine 24-Stunden-Aufsicht erfordern würde. In begründeten und belegten Einzelfällen (z.B. Krankheit von Teammitgliedern) wäre es hingegen möglich, vereinzelte Ersatztermine zu vereinbaren und Aufsicht dafür zu organisieren.</p> <p>Die Kompetenzen werden bei der Übung durch selbständige praktische Erarbeitung der Versuche in Zusammenarbeit mit weiteren Teammitgliedern erworben. Bei Versuchen, die über mehrere Wochen hinweg durchgeführt werden, ist die Auslassung einzelner Versuchsteile grundsätzlich nicht möglich, da das Ergebnis des Versuchs ein funktionierendes und getestetes System sein soll.</p>
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Modulbezeichnung:	Rechnernetze
ggf. Kürzel	5305
Empfohlenes Studiensemester:	6
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Hermann de Meer (LST für Informatik mit SP Rechnernetze und Rechnerkommunikation)
Dozent(in):	Prof. Dr. Hermann de Meer (LST für Informatik mit SP Rechnernetze und Rechnerkommunikation)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Technische Informatik
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz, 65 Std. Übungsaufgaben + 70 Std. Nachbearbeitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Rechnerarchitektur, Technische Informatik
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden erlangen Kenntnisse über die wichtigsten Protokollelemente und die Architektur des Internets. Anhand der Internet-Architektur können fundamentale Problemstellungen der Rechnerkommunikation eingeordnet und verstanden werden. Diese Problemstellungen beziehen sich auf funkbasierter Kommunikation, Fragen des Netzmanagements, der Sicherheit in der Kommunikation, der Mobilität in Netzen und der Multimediakommunikation.</p> <p>Fähigkeit: Die Studierenden können praktische Netzprogrammierung prinzipiell realisieren</p> <p>Kompetenzen: Die Studierenden erhalten die Kompetenz, elementare Zusammenhänge im Bereich von Rechnernetzen zu verstehen, einzuordnen und geeignete Methoden und Protokolle problemabhängig auszuwählen und angepasst zu implementieren.</p>
Inhalt:	Diese Vorlesung umfasst zentrale Algorithmen und Konzepte des TCP/IP Protokoll Stacks. In einem Top-Down-Ansatz wird

	ein allgemeines Verständnis für Schichtenmodelle, Schnittstellen, Protokolle und Services vermittelt. Unter anderem werden folgende Protokolle (in verschiedenen Schichten) behandelt: DNS, HTTP, SMTP, TCP, UDP, IP, Ethernet, WLAN, MiWAX, GSM, UMTS, LTE. Weitere Inhalte umfassen Prinzipien der funkbasierten Kommunikation, des Mobilitätsmanagements, der Netzsicherheit und des Netzwerkmanagements.
Studien-/Prüfungsleistungen:	120-minütige Klausur
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Vorlesung: Präsentation und Beamer, Übung: Beamer, Tafel, Rechnerlabor
Literatur:	J.F. Kurose / K.W. Ross, Computer Networking, PEARSON Addison Wesley (jeweils neueste Ausgabe, z.Zt. 6th Ed.)

Modulgruppe Systems Engineering

Modulbezeichnung:	Complex Systems Engineering
ggf. Kürzel	
Empfohlenes Studiensemester:	5
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems)
Dozent(in):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Systems Engineering
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen I, Softwaretechnik für Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen Instanzen von komplexen Systemen, die mit dem Instrumentarium einer einzelnen Disziplin nicht adäquat behandelt werden können. Sie lernen aktuelle Ansätze, um die Komplexität zu beherrschen und solche Systeme dennoch entwerfen und analysieren zu können.</p> <p>Fähigkeiten: Die Studierenden können die Komplexität von Anwendungen erkennen und systematisiert einordnen. Sie können komplexe Systeme formal beschreiben und prinzipielle Anwendbarkeit von gängigen Entwurfsmethoden bewerten.</p> <p>Kompetenzen: Die Studierenden sind in der Lage, mit aktuellen Entwurfs- und Analysemethoden komplexe Systeme zu behandeln. Sie können für Teilaspekte solcher Systeme geeignete Entwurfsabläufe identifizieren und prinzipiell anwenden. Sie können Eigenschaften wie Emergenz oder Phasenübergänge beschreiben und sich durch geeignete Entwurfsprinzipien zu Nutze machen.</p>
Inhalt:	Beispiele von komplexen Systemen

	<p>Beschreibungssprachen und –formalismen</p> <p>Methoden zur Analyse, Entwurf und Implementierung von komplexen Systemen</p> <p>Theorie komplexer verteilter Systeme</p> <p>Selbstorganisation, selbstadaptive Systeme, Phasenübergänge, Emergenz</p>
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Spezifikation und Verifikation von Eingebetteten Systemen
ggf. Kürzel	
Empfohlenes Studiensemester:	4
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) Prof. Dr. Dirk Beyer (LST für für Informatik mit SP Softwaresysteme)
Dozent(in):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) Prof. Dr. Dirk Beyer (LST für für Informatik mit SP Softwaresysteme)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Modulgruppe Systems Engineering
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 60 Std. Übungsaufgaben + 75 Std. Nachbearbeitung des Vorlesungsstoffs
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Technische Informatik, Grundlagen der Informatik für Mobile und Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die wesentlichen (insb. formalen) Spezifikationssprachen, die im Embedded-Bereich relevant sind. Sie lernen die grundlegenden Verifikationsverfahren kennen. Sie kennen die wesentlichen Begriffe im Kontext von Echtzeitsystemen und die grundlegenden Scheduling-Verfahren.</p> <p>Fähigkeiten: Die Studierenden können Anforderungen an ein eingebettetes System erkennen, eine geeignete Spezifikationssprache dafür auswählen und eine formale Spezifikation in dieser Sprache erstellen. Sie sind insbesondere in der Lage, Echtzeitanforderungen genau zu spezifizieren und die dafür bestimmten Scheduling-Verfahren festzulegen. Sie können aus einem Portfolio von Verifikationsverfahren das adäquate aussuchen und anwenden. Ein vertieftes Verständnis der prinzipiellen Herangehensweise ermöglicht hierbei eine</p>

	<p>fundierte Interpretation des Verifikationsergebnisses.</p> <p>Kompetenzen: Die Studierenden können eingebettete Systeme kleiner bis mittlerer Komplexität spezifizieren und verifizieren und dabei insbesondere die Echtzeitfähigkeit geeignet ausdrücken und garantieren. Außerdem verfügen die Studierenden über die Kompetenz, Verifikationswerkzeuge zu benutzen.</p>
Inhalt:	<p>Spezifikationssprachen: Petri-Netze, erweiterte Automatenformalismen/Statecharts, Transitionssysteme, Timed Automata, Simulink, VHDL/Verilog, Grundlagen der temporalen Logik</p> <p>Verifikationsmethoden: Simulation, Model Checking, Assertions, Equivalence Checking, Property Checking, Program Analysis</p> <p>Basistechnologien der Verifikation: BDDs, SAT, AIGs, SMT</p> <p>Grundbegriffe von Echtzeitsystemen</p> <p>Scheduling-Verfahren</p>
Studien-/Prüfungsleistungen:	<p>90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben</p>
Modulnote:	<p>Modulnote entspricht der Note der Prüfung</p>
Medienformen:	<p>Präsentation und Beamer</p>
Literatur:	<p>Wird vom Dozenten/ von der Dozentin bekannt gegeben</p>

Module ohne Gruppenzuordnung

Seminar, MES Praktikum, Bachelorarbeit-Präsentation

Modulbezeichnung:	Seminar zu Mobilien und Eingebetteten Systemen (Geeignete Seminare werden zu Beginn des Semesters durch Aushang sowie auf der Webseite der Fakultät bekannt gegeben)
ggf. Kürzel	
Empfohlenes Studiensemester:	4
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Semester
Modulverantwortliche(r):	Vorsitzender oder Vorsitzende des Prüfungsausschusses
Dozent(in):	Alle Dozenten des Studiengangs
Sprache:	Deutsch oder Englisch nach Wahl des oder der Vortragenden (Literatur wird am Anfang des Seminars bekannt gegeben)
Zuordnung zum Curriculum	MES
Lehrform/SWS:	2S
Arbeitsaufwand:	30 Std Präsenz und 90 Std. Vor- und Nachbereitung
ECTS-Leistungspunkte:	4
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen:	Weiterführende Vorlesungen aus den 3./4. Semester und/oder event. Wahlpflichtmodule
Angestrebte Lernergebnisse:	Die Studierenden lernen sich in das gestellte Seminarthema einzuarbeiten und dieses zu präsentieren. Die Erarbeitung erfolgt teils unter Anleitung teils selbständig. Sie erlernen die Präsentation fachbezogener Inhalte. Kompetenzen: Selbständige Einarbeitung in ein Thema, schriftliche Erörterung, mündliche Ausdrucks- und Präsentationskompetenz
Inhalt:	Erarbeitung des gestellten Themas anhand von wissenschaftlicher Literatur und dessen Präsentation
Studien-/Prüfungsleistungen:	Bewertung der schriftlichen Ausarbeitung (ca. 8-10 Seiten) und deren Präsentation (ca. 20 Min.). Dabei wird jeweils die mündliche Ausdrucks- und Präsentationskompetenz bzw. die schriftliche Erörterungskompetenz geprüft; für beide Leistungen wird eine gemeinsame Note vergeben.
Modulnote:	Modulnote entspricht der Note der Prüfung

Medienformen:	Präsentation
Literatur:	Wird vom Dozenten oder der Dozentin bekannt gegeben

Modulbezeichnung:	MES Praktikum
ggf. Kürzel	MESP
Empfohlenes Studiensemester:	5. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Dozent(in):	Lehrstuhl für Technische Informatik Lehrstuhl für Complex Systems Engineering Lehrstuhl für Digitale Bildverarbeitung Lehrstuhl für Sensorik Lehrstuhl für IT-Sicherheit Lehrstuhl für Rechnernetze und Rechnerkommunikation
Sprache:	Deutsch
Zuordnung zum Curriculum	MES
Lehrform/SWS:	8P
Arbeitsaufwand:	40 Std. Einarbeitung in Entwurfswerkzeuge + 45 Std. Teammeetings + 25 Std. Projektmanagement + 70 Std. Analyse und Spezifikation + 110 Std. Entwurf und Implementierung + 40 Std. Validierung + 10 Std. Berichterstellung + 10 Std. Kolloquien + deren Vorbereitung + 10 Std. Präsentation und deren Vorbereitung Gesamt: 360 Std.
ECTS-Leistungspunkte:	12
Voraussetzungen nach Prüfungsordnung	Erfolgreiche Absolvierung der Pflichtmodule „Softwaretechnik für Eingebettete Systeme“, „Spezifikation und Verifikation von Eingebetteten Systemen“ <i>Hinweis:</i> <i>Die Voraussetzung der erfolgreichen Absolvierung des Pflichtmoduls „Spezifikation und Verifikation von Eingebetteten Systemen“ ist für das WS 2016/17 und das SS 2017 ausgesetzt.</i> Begründung: In dem Praktikum sollen die Studierenden in mehreren größeren (über mehrere Sitzungen hinweg durchgeführten) Versuchen in Teams die theoretisch erlangten Kenntnisse praktisch anwenden und die Abläufe und die einzelnen Arbeitsphasen eines größeren, zusammenhängenden Projekts kennen lernen. Zur Durchführung dieses Praktikums werden spezielle Labore mit aufwändiger Technik und mit begrenzter Kapazität verwendet, was eine natürliche Begrenzung der Teilnehmerzahl erfordert. Die Anforderung von bestimmten fachlichen Mindestkompetenzen, die durch bestandene Prüfungen der angegebenen Grundlagenmodule nachzuweisen sind, ist für eine erfolgreiche Praktikumsteilnahme eine Basisvoraussetzung. Die Zulassung von Studierenden ohne

	<p>diese Mindestvoraussetzungen würde sowohl ihren Studienerfolg als auch den Studienerfolg ihrer Teammitglieder, die sich auf eine erfolgreiche Bearbeitung aller Versuchsteile verlassen müssen, in Frage stellen.</p>
Empfohlene Voraussetzungen:	<p>Complex Systems Engineering, Programmierung in Java, Grundlagen der Informatik für Mobile und Eingebettete Systeme, Software-Hardware Codesign</p>
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen die Realisierung von umfangreichen Engineering-Projekten aus dem Kontext des Studienganges und die dazu notwendigen Vorgehensweisen, Methoden und Werkzeuge kennen. Die vorhandenen Kenntnisse in der Softwareentwicklung und Codeerstellung werden vertieft. Die Kenntnisse in Projektmanagement werden vertieft. Die Studierenden sammeln Erfahrungen mit der Teamarbeit.</p> <p>Fähigkeiten: Die Studierenden beherrschen die praktischen Fragestellungen der Entwicklung und Umsetzung von Systemen, sowie des dazu nötigen Projektmanagements, können im Team effektive Lösungen erarbeiten und durchführen und erfolgreich ein großes Projekt im Team realisieren. Die Studierenden vertiefen die Fähigkeit im Team Probleme zu lösen und ggf. konstruktive Konfliktlösungsstrategien zu finden.</p> <p>Kompetenzen: Die Teilnehmer erlernen soziale Kompetenz durch die Teamarbeit und die notwendigen organisatorischen und fachlichen Kompetenzen zur Durchführung größerer Projekte aus dem Kontext des Studienganges im Team und unter Zeitdruck erfolgreich zu bearbeiten, da der Umfang der Aufgabe nur mit durchdachter Arbeitsteilung erfolgreich erfüllt werden kann.</p> <p>Teil des Lernziels besteht in der Abschätzung und Kontrolle des Arbeitsaufwandes, sowie der Entwicklung und Anwendung von Strategien zum erfolgreichen Projektmanagement. Dazu werden Stundenzettel geführt.</p>
Inhalt:	<p>Im Rahmen des Praktikums wird eine realitäts- bzw. forschungsnahe Problemstellung aus dem Kontext des Studienganges mittels der Problemstellung angemessener Methoden und Werkzeuge im Team bearbeitet. Dabei werden geeignete Vorgehensweisen zur Projekt- und Arbeitsorganisation in der Gruppe erlernt und angewendet. Das jeweils entsprechend der Aufgaben- und Problemstellung angemessene Vorgehen entspricht dabei so weit wie möglich der bestehenden Praxis in Industrie und Forschung. Der Fokus liegt dabei weniger auf den Methoden des Software Engineerings, sondern auf dem Systems Engineering und der Methodenanwendung aus dem jeweiligen Kontext der Problemstellung. Für die Entwicklung der Software</p>

als Bestandteil eines größeren Systems, bestehend aus Software und Hardware, finden jeweils geeignete Vorgehensmodelle Anwendung, z.B. agile Vorgehensmodelle.

Es wird ein komplexes Projekt systematisch bearbeitet. Die Bearbeitung des Projektes erfolgt in einer der Aufgabenstellung angemessener Teamgröße von in der Regel 3 bis 5 Studierenden. Die durchzuführende Projektarbeit wird geeignet in bearbeitbare Arbeitspakete unterteilt. Die Umfänge und Gewichtungen der einzelnen Aktivitäten eines jeden Arbeitspakets sind dabei von der konkreten Problemstellung abhängig. Dies gilt ebenfalls für die konkret anzuwendenden Methoden bzw. einzusetzenden Werkzeuge. Diese können auf Grund der Vielfältigkeit des Gebiets der mobilen und eingebetteten Systeme nur im Kontext der konkreten Problemstellung ausgewählt werden. Wiederkehrende Aktivitäten bei der Bearbeitung der einzelnen Arbeitspakete sind dabei:

1. Analyse

Detaillierte Festlegung der Anforderungen an das System. Beachtung der Grundprinzipien Präzision, Vollständigkeit und Konsistenz. Der Inhalt umfasst das Systemmodell als Übersicht, die geeignete Beschreibung der Systemumgebung mittels geeigneter Werkzeuge, sowie die Erfassung und Dokumentation funktionaler und nicht-funktionaler Anforderungen.

2. Entwurf

Hauptbestandteil ist ein systematischer Grobentwurf eines Systems, das die in der Analyse ermittelten Anforderungen bestmöglich erfüllt. Auf dieser Basis wird ein detaillierter Entwurf ausgearbeitet, der mit der Problemstellung angemessenen, domänenspezifischen Werkzeugen und Vorgehensweisen das umzusetzende System spezifiziert und dokumentiert.

3. Umsetzung

Im Rahmen der Umsetzung erfolgt die tatsächliche Realisierung des entworfenen Systems. Das System besteht in der Regel aus Software- und Hardware-Komponenten. Es werden Arbeitseinheiten definiert, deren Aufwand abgeschätzt und deren Realisierung zugeordnet. Zusätzlich zum fertigen System wird am Ende ein Bericht erstellt, aus dem evtl. Aufwandsabweichungen oder Modellkorrekturen ersichtlich sind.

4. Validierung

Die Validierung der Umsetzung auf Basis der in einer Analyse bestimmten Anforderungen dient zum Abgleich des Ist-Zustandes mit dem Soll-Zustand. es wird eine detaillierte Systemdokumentation erstellt und von den Aufgabenstellern zusammen mit dem System abgenommen.

Allgemein gilt dabei:

Jedes Arbeitspaket kann eine oder mehrere dieser Aktivitäten umfassen und jede Aktivität kann Gegenstand von einem oder mehrerer Arbeitspaketen sein. Dabei müssen alle Inhalte der Aufgabenstellung durch Arbeitspakete adäquat abgedeckt sein. In den einzelnen Arbeitspaketen kommen projekt- und domänenspezifische Werkzeuge, Methoden und Beschreibungssprachen zum Einsatz. Das Ergebnis eines jeden Arbeitspaketes ist ein eigenes Dokument, ggf. begleitet von Software, ggf. begleitet von Anhängen mit z.B. Software oder Beschreibungen von Hardwareblöcken in geeigneten

	<p>Spezifikationssprachen.</p> <p>Jedes Arbeitspaket schließt mit einem kurzen Kolloquium ab, in dem die Ergebnisse den Betreuern präsentiert und verteidigt werden. Das Kolloquium kann auch die Präsentation zusammen mit anderen Teams umfassen um eine reflektive Diskussion über die Ergebnisse und Vorgehensweisen zu ermöglichen.</p> <p>Vortragender ist der zu Beginn festgelegte Verantwortliche des Arbeitspakets. Jeder Teilnehmer übernimmt mind. einmal die Rolle des Arbeitspaket-Verantwortlichen. Dieser ist auch für den Erfolg seines Arbeitspakets verantwortlich und regelt deshalb die Aufgabenteilung im Team. Die Teams werden durch regelmäßige Treffen mit dem Betreuer unterstützt, deren Häufigkeit der aktuellen Phase bzw. dem Bearbeitungsfortschritt angemessen ist.</p> <p>Das Praktikum schließt mit einem Abschlusskolloquium ab, in dem das fertig entwickelte System präsentiert und abgenommen wird.</p> <p>Programmiersprachen sind hauptsächlich: C/C++/Java</p> <p>Darüber hinaus werden der Problemstellung angemessene spezifische Werkzeuge und Methoden eingesetzt.</p>
Studien-/Prüfungsleistungen:	Schriftliche Dokumentation in Form eines technischen Berichts, mündliche Kolloquien zu den Arbeitspaketen die durch geeignete Medien (z.B. Folien) unterstützt werden, dokumentierter und funktionsfähiger Quelltext inkl. aller zur Demonstration notwendigen Informationen, sowie einer Systemdemonstration und Präsentation im Rahmen einer Abschlussveranstaltung.
Modulnote:	Für das Portfolio wird eine Gesamtnote vergeben, welcher die Modulnote entspricht.
Medienformen:	Präsentation mit Projektor, Gruppenarbeit, Wiki
Literatur:	Wird vom Dozent / von der Dozentin bekannt gegeben Die Literatur wird in Abhängigkeit der konkreten Aufgabenstellung ausgewählt und bekanntgegeben.
Anwesenheitspflicht:	<p>Für ausgewählte Kolloquien, sowie für spezifische Laborarbeiten die im Rahmen einer spezifischen Themenstellung im Rahmen des MES-Praktikums notwendig sind, besteht Anwesenheitspflicht.</p> <p>Begründung der Anwesenheitspflicht bei den Kolloquien: Im MES-Praktikum arbeiten die Teams von Studierenden an einem größeren Projekt über das ganze Semester hindurch. Es findet zu jedem Arbeitspaket ein Kolloquium statt, in dem über die Fortschritte berichtet, aufgetretene Probleme ausgetauscht und ihre Lösungen diskutiert werden; am Ende findet ein Abschlusskolloquium statt. Neben der Kompetenz vor Studierenden und Projektbetreuern zu präsentieren und fachliche Inhalte zu diskutieren, sind die Rückmeldungen der Dozierenden zum erreichten Stand für die Studierenden von besonderer Bedeutung, z.B. um frühzeitig korrektive Maßnahmen zur Erreichung der Praktikumsziele bzw. Aufgabenstellung zu ergreifen.</p> <p>Wird keine umfassende Anwesenheit bei den Kolloquien gefordert, wird die Kompetenz nicht geübt, vor anderen</p>

	<p>Studierenden zu präsentieren und auf ihre Fragen und Anmerkungen (und nicht nur die des Dozenten) einzugehen und diese zu diskutieren. Präsentation und Diskussion dienen zusätzlich zur Verifikation, dass die Projektbearbeitung durch alle Teammitglieder erfolgt und die Aufgabenstellung von allen verinnerlicht wurde. Die vereinzelte Abwesenheit aus nicht vom Studierenden zu vertretenden und nachgewiesenen Gründen ist möglich.</p> <p>Begründung der Anwesenheitspflicht bei Laborarbeiten: Die Aufgabenstellung kann besondere Ausstattung (Geräte, Arbeitsplätze, etc.) erfordern, die nur in den Laboren und Räumen der Universität in geeigneter Weise zur Verfügung steht. Ferner ist ggf. eine direkte Betreuung und Unterweisung an speziellen Geräten notwendig bzw. eine Aufsicht der Nutzung der Labore. Daher ist in der Regel eine Bearbeitung außerhalb dieses Kontextes nicht möglich und die Anwesenheit Voraussetzung für die erfolgreiche Bearbeitung.</p>
<p>Sonstiges:</p>	<p>Auf Beschluss der Prüfungsausschüsse der FIM vom 24.11.14 wird die Voraussetzung von 'Spezifikation und Verifikation Eingebetteter Systeme' für das MES Praktikum (StuPO MES 2012, 18 (1) bzw. FStuPO MES 2014, 5 (2)), im WS 15/16 ausgesetzt.</p>

Modulbezeichnung:	Präsentation der Bachelorarbeit MES
ggf. Kürzel	
Empfohlenes Studiensemester:	6
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Semester
Modulverantwortliche(r):	Vorsitzender oder Vorsitzende des Prüfungsausschusses
Dozent(in):	Alle Dozenten des Studiengangs
Sprache:	Deutsch oder eine Fremdsprache auf Antrag an und bei Genehmigung durch den Prüfungsausschuss
Zuordnung zum Curriculum	MES
Lehrform/SWS:	
Arbeitsaufwand:	75 Std. Nachbearbeitung und Prüfungsvorbereitung
ECTS-Leistungspunkte:	3
Voraussetzungen nach Prüfungsordnung	Abgabe der Bachelorarbeit gemäß § 20 Abs. 5
Empfohlene Voraussetzungen:	
Angestrebte Lernergebnisse:	Kompetenz, die Ergebnisse eigener wissenschaftlicher Arbeit kurz und verständlich mündlich (ggf. unter Verwendung weiterer Medien wie Folien oder Vorführungen) darzustellen und eine fachliche Diskussion über eigene Ergebnisse zu führen.
Inhalt:	Darstellung der in der Arbeit erworbenen Erkenntnisse sowie kurze Diskussion
Studien-/Prüfungsleistungen:	Vortrag von ca. 20 Minuten Dauer und Diskussion
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Beamer, Tafel, Overheadprojektor
Literatur:	Je nach Thema

Wahlpflichtmodule

Modulbezeichnung:	Einführung in die Kontexterkenkung
ggf. Kürzel	5976
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Dozent(in):	Prof. Dr. Oliver Amft (LST für Informatik mit SP Sensorik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	3V+1Ü (Blockveranstaltung)
Arbeitsaufwand:	60 Std. Präsenz + 105 Std. Projektarbeit + 45 Std. Nachbearbeitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Kenntnisse von Matlab oder Python
Angestrebte Lernergebnisse:	<p>Kenntnisse: Grundlegende Sensormodalitäten und deren Einsatzgebiete in der Kontexterkenkung, einfache Mustererkennungsalgorithmen.</p> <p>Kompetenzen: Teilnehmende können das wissenschaftliche Vorgehen bei der Entwicklung von Kontextanwendungen anwenden. Applikationen der Kontexterkenkung beschreiben, Lösungen für ausgewählte Erkennungsaufgaben spezifizieren, kleine Messstudien durchführen und Datenanalysen durchführen. Mit Hilfe geeigneter Verarbeitungssoftware (Python) können Studierende Musterklassifikation anwenden, Ergebnisse präsentieren und Anwendungsprototypen umsetzen.</p>
Inhalt:	<p>Die Veranstaltung umfasst wesentliche Themen der Aktivitäts- und Kontexterkenkung, bietet Überblick zu Kontext-Technologien, wichtigen Prinzipien des Ubiquitous Computing, und begleitender Aspekte. Neben den Grundlagen zur ubiquitären Sensortechnologie, Signalanalyse und Mustererkennung, werden Realitätskonzepte, menschliche Verhaltensmodellierung, Physiologie, Material und Textiltechnologie, wissenschaftliches Programmieren, Sicherheit und Ethik betrachtet.</p> <p>Das Arbeitsprogramm besteht aus Lehrveranstaltungen, Seminaren mit Präsentationen der Teilnehmer, sowie einem Kursprojekt. Kursprojekte werden von einer jeweils individuell betreuten Gruppe von Teilnehmern umgesetzt und dienen der Vertiefung und praktischen Anwendung/Erweiterung der vorher theoretisch diskutierten Veranstaltungsinhalte.</p> <p>Teilnehmende präsentieren die Kursprojekte am Beginn und</p>

	Ende der Arbeit. In einem kurzen Bericht werden Zielsetzung, Vorgehen, und Ergebnisse zusammengefasst.
Studien-/Prüfungsleistungen:	Präsentation (ca. 20 Min.), Bericht (ca. 15 Seiten)
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Beamer, Tafel
Literatur:	Wird vom Dozenten oder der Dozentin bekannt gegeben

Modulbezeichnung:	Embedded Systems Programming
ggf. Kürzel	
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Dozent(in):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	4P
Arbeitsaufwand:	60 Std. Präsenz + 150 Std. Vor- und Nachbearbeitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Programmierung in Java oder Grundlagen der Programmierung 1 und 2, MES Praktikum oder SEP
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen die Realisierung von umfangreichen Engineering-Projekten aus dem Kontext Eingebetteter Systeme und die dazu notwendigen Vorgehensweisen, Methoden und Werkzeuge kennen. Theoretische Kenntnisse vom Entwurf hardwarenaher Systeme, der Entwicklung spezialisierter Anwendungen und allgemeiner Software-Systeme werden praktisch angewendet und durch die Systementwicklung eines komplexen Gesamtsystems vertieft.</p> <p>Fähigkeiten: Die Studierenden beherrschen die praktischen Fragestellungen der Entwicklung und Umsetzung von Systemen bestehend aus eingebetteten Systemen in technischen Kontexten. Die Studierenden beherrschen die relevanten Werkzeuge und Systeme für die Entwicklung und Testung eingebetteter Systeme und Entwicklungsparadigmen. Die Studierenden können in einem kleinen Team effektive Lösungen erarbeiten und durchführen und erfolgreich ein vorlesungsbegleitendes Projekt im Team realisieren.</p> <p>Kompetenzen: Die Teilnehmer erlernen soziale Kompetenz durch die Teamarbeit und die notwendigen organisatorischen und fachlichen Kompetenzen zur Durchführung von Projekten aus dem Kontext der eingebetteten Anwendungsentwicklung erfolgreich zu bearbeiten. Teil des Lernziels besteht in der Abschätzung und Kontrolle des Arbeitsaufwandes, sowie der Entwicklung von Strate-</p>

	gien zum erfolgreichen Projektmanagement. Dazu werden Stundenzettel geführt.
Inhalt:	<p>Im Rahmen der Lehrveranstaltung wird ein dem Umfang der Lehrveranstaltung angepasstes eingebettetes System realitätsnah entwickelt mittels der Problemstellung angemessener Methoden und Werkzeuge im Team bearbeitet unter Anwendung geeigneter Vorgehensweisen zur Projekt- und Arbeitsorganisation. Insbesondere werden Vorgehensweisen aus den Bereichen Software Entwicklung (Prototyping, Entwicklung, Test-Driven Development, Entwicklungsprozesse, Continuous Integration Server) und hardwarenahem Systems Engineering (hardware in the loop (HIL), in-circuit debugging (ICD), Simulationssysteme) eingesetzt. Das Vorgehen deckt sich soweit möglich mit bestehender Praxis aus Industrie und Forschung.</p> <p>Teams von in der Regel 2-3 Studierenden bearbeiten in der Übung gemeinsam und systematisch ein kleineres Projekt, das in mehrere Arbeitspakete strukturiert ist. Die genaue Aufgabenstellung wird zu Beginn der Lehrveranstaltung schriftlich in Form einer Zielvorgabe mit minimalen Eigenschaften als Bestehenskriterien vorgegeben.</p> <p>Bei der Bearbeitung des vorlesungsbegleitenden Projekts werden folgende Engineering-Aktivitäten abgedeckt:</p> <p>1. Analyse Detaillierte Festlegung der Anforderungen an das System. Beachtung der Grundprinzipien Präzision, Vollständigkeit und Konsistenz. Der Inhalt umfasst das Systemmodell als Übersicht, die geeignete Beschreibung der Systemumgebung mittels geeigneter Werkzeuge, sowie die Erfassung und Dokumentation funktionaler und nicht-funktionaler Anforderungen.</p> <p>2. Entwurf Hauptbestandteil ist ein systematischer Grobentwurf eines Systems, das die in der Analyse ermittelten Anforderungen bestmöglich erfüllt. Auf dieser Basis wird ein detaillierter Entwurf ausgearbeitet, der mit der Problemstellung angemessenen, domänenspezifischen Werkzeugen und Vorgehensweisen das umzusetzende System spezifiziert und dokumentiert.</p> <p>3. Umsetzung Im Rahmen der Umsetzung erfolgt die tatsächliche Realisierung des entworfenen Systems. Das System besteht in der Regel aus Software- und Hardware-Komponenten. Zur Realisierung sind bestehende, konfigurierbare Softwarebausteine mit eigener Software zu ergänzen und zu einem lauffähigen Gesamtsystem zu integrieren. Hierzu werden Methoden aus dem Bereich der verteilten Systeme, z.B. Architekturentwurf, oder der vernetzten Systeme, z.B. Socket-Programmierung, verwendet.</p> <p>4. Validierung Validierung und Verifikation der Ergebnisse von Entwurf und Umsetzung auf Grundlage der durch Analyse bestimmten Anforderungen.</p> <p>Jedes Arbeitspaket kann eine oder mehrere dieser Aktivitäten umfassen und jede Aktivität kann Gegenstand eines oder mehrerer Arbeitspakete sein. Dabei müssen alle Aktivitäten durch Arbeitspakete adäquat abgedeckt sein. In den einzelnen Arbeitspaketen kommen projekt- und domänenspezifische</p>

	<p>Werkzeuge und Methoden zum Einsatz.</p> <p>Zu allen Arbeitspaketen werden Arbeitseinheiten definiert, deren Aufwand abgeschätzt und deren Realisierung z.B. anhand einer Gantt-Chart organisiert und durchgeführt. Das Ergebnis jedes Arbeitspakets wird durch einen kurzen Bericht dokumentiert, ggf. begleitet von Software. Aus dem Bericht sind auch Aufwandsabweichungen und Korrekturen vorangegangener Arbeitspakete ersichtlich.</p> <p>Jedes Arbeitspaket schließt mit einem Kurzvortrag in der nächsten Einheit ab. Die Teams werden durch ein festes wöchentliches Treffen mit dem Betreuer unterstützt.</p> <p>Die Lehrveranstaltung schließt mit einem Abschlusskolloquium ab, in dem das fertig entwickelte System präsentiert und abgenommen wird.</p>
Studien-/Prüfungsleistungen:	Schriftliche Dokumentation in Form eines technischen Berichts, mündliche Präsentationen zu den Arbeitspaketen die durch geeignete Medien (z.B. Folien) unterstützt werden, dokumentierter und funktionsfähiger Quelltext inkl. aller zur Demonstration notwendigen Informationen, sowie einer Systemdemonstration im Rahmen des Abschlusskolloquiums.
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation mit Projektor, Gruppenarbeit, Wiki
Literatur:	<p>Wird vom Dozent / von der Dozentin bekannt gegeben</p> <p>Die Literatur wird in Abhängigkeit der konkreten Aufgabenstellung ausgewählt und bekanntgegeben.</p>
Anwesenheitspflicht:	<p>Für die Lehrveranstaltung besteht Anwesenheitspflicht. Es findet eine wissenschaftlich-technische Einführung zu den Themen der Lehrveranstaltung statt, diese werden in den anschließenden Übungen direkt mit der Übungsleitung praktisch umgesetzt.</p> <p>Begründung: In der Lehrveranstaltung arbeiten die Teams von Studierenden an einem größeren Projekt über das ganze Semester hindurch. Es zu jedem Arbeitspaket bzw. Themengebiet ein Kolloquium statt, in denen über die Fortschritte berichtet, aufgetretene Probleme ausgetauscht und ihre Lösungen diskutiert werden; am Ende findet ein Abschlusskolloquium statt.</p> <p>Wird keine umfassende Anwesenheit bei den Kolloquien gefordert, wird die Kompetenz nicht geübt, vor anderen Studierenden zu präsentieren und auf ihre Fragen und Anmerkungen (und nicht nur die des Dozenten) einzugehen und diese zu diskutieren. Die Kompetenz, die präsentierten Inhalte zu analysieren, bewerten und kritisch zu diskutieren ist eine wesentliche Anwendung der Lehrveranstaltungsinhalte die nur bei Präsenz eingeübt werden kann. Die vereinzelte Abwesenheit aus nicht vom Studierenden zu vertretenden und</p>

	<p>nachgewiesenen Gründen ist möglich.</p> <p>Darüber hinaus kann die spezifische Aufgabenstellung besondere Ausstattung erfordern, die nur in den Laboren und Räumen der Universität in geeigneter Weise zur Verfügung steht. Ferner ist ggf. eine direkte Betreuung und Unterweisung an speziellen Geräten notwendig. Daher ist bei spezifischer Aufgabenstellung eine Bearbeitung außerhalb dieses Kontextes nicht möglich und die Anwesenheit dann zwingende Voraussetzung für die erfolgreiche Bearbeitung. Andernfalls ist die erfolgreiche Teilnahme an der Lehrveranstaltung gefährdet.</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Modulbezeichnung:	Intelligent Audio Analysis
ggf. Kürzel	
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems)
Dozent(in):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) u.a.
Sprache:	Englisch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	2V + 1Ü
Arbeitsaufwand:	45 Std. Präsenz, 105 Stunden selbständige Vor- und Nachbereitung
ECTS-Leistungspunkte:	5
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagenvorlesungen der Mathematik
Angestrebte Lernergebnisse:	<p>Knowledge: The students learn the principal concepts of sequential signal processing, signal source separation, and feature extraction and information reduction exemplified by audio signals. They further gain insight into machine learning principles such as learning dynamics and context as is needed for intelligent analysis of audio. They will learn about different problems and solutions in the analysis of speech, music, and general sound.</p> <p>Skills: The students will be able to choose appropriate algorithms of signal processing and machine intelligence, further develop these, design new solutions, and apply these to a broad range of audio and general signal analysis problems.</p> <p>Competences: The students are able to characterise, judge on the quality and suitability, and design suited algorithmic solutions for intelligent signal analysis with a focus on audio signals. They are further able to extract meaningful and relevant features and process these with modern approaches of machine intelligence.</p>
Inhalt:	Basics of Signal Processing, Signal Source Separation, Audio Features, Speech Features, Automatic Speech Recognition, Computational Paralinguistics, Music Information Retrieval, Music Transcription, Auditory Scene Analysis, Audio Structure Analysis, Audio Fingerprinting
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 25 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt

	gegeben.
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Björn Schuller: „ <i>Intelligent Audio Analysis</i> “, Signals and Communication Technology, Springer, ISBN: 978-3642368059, 2013.

Modulbezeichnung:	Intelligent Systems (Blockseminar)
ggf. Kürzel	5954
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems)
Dozent(in):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) u.a.
Sprache:	Englisch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	Blockveranstaltung (4 Tage)
Arbeitsaufwand:	30 Std. Präsenz + 60 Std. Nachbearbeitung der gestellten Aufgaben
ECTS-Leistungspunkte:	3
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	keine
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen die generelle Funktionsweise von Systemen zur allgemeinen Mustererkennung zu verstehen und erwerben grundlegendes Wissen zu geeigneten maschinellen Lernverfahren.</p> <p>Fertigkeiten: Sie können intelligente Systeme in Bezug auf die algorithmische Lösung bewerten. Sie sind weiterhin mit Verfahren zur Leistungsevaluierung eines komplexen Systems vertraut. Darüber hinaus können grundlegende Probleme der Mustererkennung analysiert und Verhaltensweisen komplexer Systeme interpretiert werden.</p> <p>Kompetenzen: Die Studierenden können unterschiedliche Arten der Informationsverarbeitung und –analyse spezifizieren und im Rahmen von praxisnahen Programmieraufgaben algorithmisch realisieren. Sie können ferner kritisch Fehlverhalten erkennen und bewerten und Lösungen zur Reduktion finden.</p>
Inhalt:	<p>Im Gebiet der komplexen Systeme werden vielschichtige Systeme behandelt, deren Verhalten nicht einfach vorhergesagt werden kann und welche ständigen Veränderungen unterliegen. Die Systeme weisen dabei bestimmte Eigenschaften auf, u.A. Emergenz, d.h. die Fähigkeit, neue Eigenschaften zu entwickeln und Selbstorganisation. Beispiele komplexer Systeme sind das menschliche Gehirn oder soziale und Kommunikationssysteme.</p> <p>Anhand von praktischen Aufgaben werden in erster Linie stochastische Modelle für die Informationsverarbeitung betrachtet. Beispiele beinhalten die intelligente Verarbeitung von Audio-</p>

	und Videosignalen.
Studien-/Prüfungsleistungen:	Präsenz und Protokollierung der Arbeitsergebnisse
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	Wird vom Dozenten/ von der Dozentin bekannt gegeben

Modulbezeichnung:	Intelligent Systems (Übung)
ggf. Kürzel	5953
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems)
Dozent(in):	Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) u.a.
Sprache:	Englisch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	3Ü
Arbeitsaufwand:	45 Std. Übungsaufgaben + 75 Std. Nachbearbeitung der Übungen und Prüfungsvorbereitung
ECTS-Leistungspunkte:	5
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagenvorlesungen der Mathematik, Besuch des Moduls Complex Systems Engineering ist von Vorteil
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen die generelle Funktionsweise von Systemen zur allgemeinen Mustererkennung zu verstehen und erwerben grundlegendes Wissen zu geeigneten maschinellen Lernverfahren.</p> <p>Fertigkeiten: Sie können intelligente Systeme in Bezug auf die algorithmische Lösung bewerten. Sie sind weiterhin mit Verfahren zur Leistungsevaluierung eines intelligenten Systems vertraut. Darüber hinaus können grundlegende Probleme der Mustererkennung analysiert und Verhaltensweisen intelligenter Systeme interpretiert werden.</p> <p>Kompetenzen: Die Studierenden können unterschiedliche Arten der intelligenten Informationsverarbeitung und –analyse spezifizieren und algorithmisch realisieren. Sie können ferner kritisch Fehlverhalten erkennen und bewerten und Lösungen zur Reduktion finden.</p>
Inhalt:	Einführung zu intelligenten Systemen. Symbolische und signalbasierte Merkmale. Grundlagen der maschinellen Intelligenz: Lineare Entscheidungsfunktionen, Abstandsklassifikatoren, Nächster-Nachbar-Regel, Kernelmaschinen, Bayes'scher Klassifikator, regelbasierte Verfahren, Entscheidungsbäume, Ensemblelernen, neuronale Netze, dynamische Klassifikation. Klassifikation und Regression. Lernverfahren. Merkmalsreduktion und Merkmalsselektion. Verfahren der Clusteranalyse, teilüberwachtes Lernen. Evaluierung.

Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben.
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer begleitet durch freie Software zur Veranschaulichung/Übung
Literatur:	I.H. Witten, F. Eibe, M.A. Hall: Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, 2011. B. Schuller: Intelligent Audio Analysis, Springer, 2013. K. Kroschel, G. Rigoll, B. Schuller: Statistische Informationstechnik, 5. Neuauflage, Springer, 2011.

Modulbezeichnung:	Numerische Methoden der Linearen Algebra
ggf. Kürzel	5751
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Brigitte Forster (Prof. für Angewandte Mathematik)
Dozent(in):	Prof. Dr. Brigitte Forster (Prof. für Angewandte Mathematik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	4V+2Ü
Arbeitsaufwand:	90 Std. Präsenz + 90 Std. Projektarbeit + 90 Std. Nachbearbeitung
ECTS-Leistungspunkte:	9
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Mathematik in technischen Systemen I-III
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden sollen grundlegende Verfahren der numerischen linearen Algebra kennen und deren Grenzen beurteilen können. Darüber hinaus sollen sie für Genauigkeitsfragen und den Einfluss von Rundungsfehlern der Methoden sensibilisiert sein.</p> <p>Fähigkeiten: Implementierung grundlegender Algorithmen, Untersuchung der Algorithmen auf Stabilität. Nutzung bestehender Software zur Lösung von numerischen Problemen der linearen Algebra.</p> <p>Kompetenzen: Lösen von realitätsnahen Problemen der numerischen linearen Algebra, zum Teil mit Computerunterstützung. Bewertung der Lösungsmethoden.</p>
Inhalt:	Singulärwert-Zerlegung, QR-Faktorisierung, Konditionierung und Stabilität, Numerische Lösung von linearen Gleichungssystemen, Numerische Bestimmung von Eigenwerten und Eigenvektoren
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben.
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer oder Tafel
Literatur:	<p>James W. Demmel: Applied Numerical Linear Algebra, SIAM, 1997</p> <p>Llyod N. Trefethen, Davod Bau III: Numerical Linear Algebra, SIAM, 1997</p>

	N. Higham, Accuracy and stability of numerical algorithms, SIAM, 1996
--	-----------------------------------------------------------------------

Modulbezeichnung:	Praktikum für Mobile und Eingebettete Systeme
ggf. Kürzel	
Empfohlenes Studiensemester:	5. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Kranz (Studiengangsverantwortlicher)
Dozent(in):	Alle Dozenten und Dozentinnen
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	Praktikum
Arbeitsaufwand:	Mindestens 6 Wochen in Vollzeit (40h/Woche) = 240 Stunden
ECTS-Leistungspunkte:	8
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagen der Informatik für Mobile und Eingebettete Systeme, Programmierung in Java, Softwaretechnik für Eingebettete Systeme
Angestrebte Lernergebnisse:	<p><u>Kenntnisse:</u> Die Studierenden lernen unter der wissenschaftlichen Betreuung durch einen Hochschullehrer oder eine Hochschullehrerin den beruflichen Alltag in einem typischen Berufsfeld ihres Studienfachs kennen und erwerben Kenntnisse über die Tätigkeiten und Anforderungen. Darüber hinaus sollen auch betriebliche Zusammenhänge und Aspekte von Mitarbeiterführung und Management kennen gelernt werden.</p> <p><u>Fähigkeiten:</u> Die Studierenden können im beruflichen Umfeld die im Studium erworbenen Kenntnisse, Fähigkeiten und Kompetenzen anwenden.</p> <p><u>Kompetenzen:</u> Die Studierenden erwerben die Kompetenz, im beruflichen Umfeld zielgerichtet und im Team tätig zu sein. Sie kennen den Unterschied zwischen Studium und Praxis.</p>
Inhalt:	<p>Eine Praktikumsstätigkeit in einem Wirtschaftsunternehmen, einer außeruniversitären öffentlichen Verwaltungseinrichtung oder einer gemeinnützigen Organisation, die in einem engen Bezug zum späteren Berufsfeld und den Tätigkeitsanforderungen für Absolventen des Studiengangs steht.</p> <p>Das Praktikum wird gemäß den folgenden Richtlinien durchgeführt:</p> <ol style="list-style-type: none"> 1. Das Praktikum umfasst mindestens 240 Stunden (= mind. 6 Wochen in Vollzeitarbeit), längere Praktika sind möglich, die Mehrzeit wird aber nicht als Studienleistung angerechnet. 2. Für Praktika eignen sich alle Betriebe und Einrichtungen im Bereich zukünftiger Berufsfelder für Absolventen des jeweiligen Studiengangs, sowie Tätigkeiten, bei denen die Anwendung von im Studium zu erwerbenden Kom-

	<p>petenzen auf Hochschulniveau nötig ist. Grundsätzlich nicht anerkannt werden Praktika, bei denen Tätigkeiten ausgeübt wurden, in denen Kompetenzen des Studiengangs keine oder nur eine untergeordnete Rolle spielt, etwa reine Büro- oder Verwaltungstätigkeiten.</p> <ol style="list-style-type: none"> 3. Das Praktikum wird von einem Hochschullehrer oder einer Hochschullehrerin des entsprechenden Fachbereichs wissenschaftlich betreut, der als Prüfer oder die als Prüferin im Studiengang bestellt ist. 4. Die Studierenden suchen für sie geeignete Praktika und beteiligen sich an der Organisation des Praktikums. Der betreuende Hochschullehrer oder die betreuende Hochschullehrerin kann die Studierenden bei der Suche unterstützen und berät gegebenenfalls die Studierenden fachlich während der Durchführung des Praktikums. 5. Ein Praktikum kann entweder in einem Block oder in mehreren Abschnitten durchgeführt werden. Jeder Abschnitt des Praktikums ist dem oder der Modulverantwortlichen zur Kenntnis zu bringen. Die Information des oder der Modulverantwortlichen soll rechtzeitig schriftlich unter Angabe des Betreuers oder der Betreuerin, des Betriebs sowie der Art und Dauer der vorgesehenen Tätigkeit erfolgen. 6. Spätestens zwei Monate nach Abschluss des Praktikums sind dem betreuenden Hochschullehrer oder der betreuenden Hochschullehrerin qualifizierende Zeugnisse über die Tätigkeit und ein Praktikumsbericht vorzulegen. Der betreuende Hochschullehrer oder die betreuende Hochschullehrerin beurteilt unter Verwendung dieser Unterlagen und eines Prüfungsgesprächs die erfolgreiche Durchführung des Praktikums.
Studien-/Prüfungsleistungen:	Praktikumsbericht und Prüfungsgespräch (ca. 20 min)
Modulnote:	unbenotet
Medienformen:	-
Literatur:	-
Sonstiges	Formular zum Antrag auf Anerkennung

Modulbezeichnung:	Praktikum Regelung und Robotik
ggf. Kürzel	
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Fabian Wirth (LST für Mathematik mit Schwerpunkt Dynamische Systeme)
Dozent(in):	Prof. Dr. Fabian Wirth (LST für Mathematik mit Schwerpunkt Dynamische Systeme), Dr.-Ing. Cornelia Schwarz
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	1V+1P+2Ü
Arbeitsaufwand:	15+15+30 Std. Präsenz, 90+60 Std. Eigenarbeitszeit
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen I-III und Grundlagen Dynamischer Systeme
Angestrebte Lernergebnisse:	<p><u>Kenntnisse:</u> Die Studierenden kennen Grundlagen der mathematischen Modellierung autonomer Roboter, sie sind mit wichtigen Regelungsprinzipien vertraut und kennen Methoden zur praktischen Umsetzung, Implementierung und Evaluation.</p> <p><u>Fähigkeiten und Kompetenzen:</u> Die Studierenden sind in der Lage einfache Robotermodelle zur Lösung konkreter Aufgaben zu entwickeln. Sie können Regler entwerfen und implementieren. Sie haben Erfahrung in der Evaluierung mittels simulativer Studien und durch praktische Experimente.</p>
Inhalt:	<p>Mathematische Modellierung und Fahrzeugdynamik Regelungskonzepte und –algorithmen Simulationsverfahren Konstruktion von Robotermodellen zur Lösung konkreter Aufgaben</p> <ul style="list-style-type: none"> • Analyse der Aufgabenstellung; • Ermittlung der notwendigen Sensoren und Aktuatoren; • Bau des Roboters und Implementierung; • Inbetriebnahme und Funktionsnachweis;
Studien-/Prüfungsleistungen:	Vollständige schriftliche Dokumentation (10-15 Seiten) und Präsentation mit Diskussion (ca. 30 min) zur gewählten Aufgabenstellung.

Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Tafel, Beamer, Vorlesungsskript, Übungsblätter
Literatur:	Vorlesungsbegleitendes Skript, Herstellerunterlagen

Modulbezeichnung:	Praktische Parallelprogrammierung
ggf. Kürzel	5610
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	in der Regel im Sommersemester
Modulverantwortliche(r):	Prof. Christian Lengauer (LST für Informatik mit SP Programmierung)
Dozent(in):	Prof. Christian Lengauer (LST für Informatik mit SP Programmierung)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 50 Std. Übungsaufgaben + 85 Std. Nachbearbeitung und Prüfungsvorbereitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	keine
Angestrebte Lernergebnisse:	<p><u>Kenntnisse:</u></p> <p>Die Studierenden lernen mehrere parallele Architekturen und eine Reihe von verschiedenen Ansätzen zur Parallelprogrammierung kennen. Sie werden in die Lage versetzt, für eine vorliegende Problemstellung und parallele Plattform den geeigneten Programmieransatz auszuwählen und anzuwenden.</p> <p><u>Fähigkeiten:</u></p> <p>Die Studierenden beherrschen die Konzepte der Parallelisierung konkreter Anwenderprobleme und können diese für eine ausgewählte Programmiersprache umsetzen.</p> <p><u>Kompetenzen:</u></p> <p>Absolventen der Veranstaltung sind sich der Vielfalt paralleler Architekturen und Programmierungsansätze bewusst. Sie kennen die erhöhten Anforderungen an die Korrektheit von parallelen Programmen, verglichen mit sequenziellen Programmen, und würdigen auch den Stellenwert der höheren Performanz paralleler Programme gegenüber äquivalenten sequenziellen Lösungen.</p>
Inhalt:	Es werden etwa ein halbes Dutzend verschiedene Paradigmen zur Parallelprogrammierung vorgestellt. Beispiele sind MPI, OpenMP, BSP, High-Performance Fortran und Java. Mindestens

	drei werden in Programmierprojekten vertieft. Es werden grundlegende Größen und Gesetze paralleler Berechnungen vorgestellt und theoretische Grundkenntnisse in der Spezifikation und Analyse von parallelen Abläufen vermittelt. Die Vor- und Nachteile verschiedener Vernetzungsmuster werden angesprochen.
Studien-/Prüfungsleistungen:	Implementierungen (Quellcode) zu mehreren Programmierprojekten mit je etwa 2 bis 3 Wochen Bearbeitungszeit.
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Beamer, Tafel, Overheadprojektor
Literatur:	<p>Foliensätze, Buchauszüge, Forschungspapiere</p> <p>Ian Foster http://www-fp.mcs.anl.gov/~foster/: /Designing and Building Parallel Programs/ Addison-Wesley, 1994.</p> <p>Michael J. Quinn http://www.cs.orst.edu/~quinn/: /Parallel Programming in C with MPI and OpenMP/ McGraw-Hill, 2004.</p>

Modulbezeichnung:	Programming Applications for Mobile Interaction
ggf. Kürzel	(alter Name: Programmierung mobiler Endgeräte)
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	in der Regel im Sommersemester
Modulverantwortliche(r):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme)
Dozent(in):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme) Prof. Dr. Ilija Polian (LST für Technische Informatik) Prof. Dr. Björn Schuller (LST für Complex and Intelligent Systems) Prof. Dr. Oliver Amft (LST für Informatik mit Schwerpunkt Sensorik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	3V+2Ü
Arbeitsaufwand:	75 Std. Präsenz + 135 Std. Übungsaufgaben, Nachbearbeitung und Prüfungsvorbereitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen:	Programmierung in Java, Mensch-Maschine-Interaktion, MES Praktikum, zusätzlich ggf. Verteilte Systeme
Angestrebte Lernergebnisse:	<p><u>Kenntnisse:</u></p> <p>Die Studierenden lernen die Realisierung von umfangreichen Engineering-Projekten aus dem Kontext Mobiler Anwendungen und Systeme und die dazu notwendigen Vorgehensweisen, Methoden und Werkzeuge kennen. Theoretische Kenntnisse vom Entwurf verteilter Systeme, der Entwicklung mobiler Anwendungen und Rechnernetze werden praktisch angewendet und durch die Systementwicklung eines komplexeren Gesamtsystems vertieft.</p> <p><u>Fähigkeiten:</u></p> <p>Die Studierenden beherrschen die praktischen Fragestellungen der Entwicklung und Umsetzung von Systemen bestehend aus mobilen Endgeräten und zentralen bzw. de-zentralen Infrastrukturen sowie den maßgeblichen Einfluss der Mensch-Maschine-Interaktion mit dem Gesamtsystem. Die Studierenden können in einem kleinen Team effektive Lösungen erarbeiten und durch-</p>

	<p>führen und erfolgreich ein vorlesungsbegleitendes Projekt im Team realisieren.</p> <p><u>Kompetenzen:</u></p> <p>Die Teilnehmer erlernen soziale Kompetenz durch die Teamarbeit und die notwendigen organisatorischen und fachlichen Kompetenzen zur Durchführung von Projekten aus dem Kontext der mobilen Anwendungsentwicklung erfolgreich zu bearbeiten. Teil des Lernziels besteht in der Abschätzung und Kontrolle des Arbeitsaufwandes, sowie der Entwicklung von Strategien zum erfolgreichen Projektmanagement. Dazu werden Stundenzettel geführt.</p>
<p>Inhalt:</p>	<p>Im Rahmen der Lehrveranstaltung wird eine dem Umfang der Lehrveranstaltung angepasste mobile Anwendung (ggf. mit zugehörigem Backend-System) realitätsnah entwickelt mittels der Problemstellung angemessener Methoden und Werkzeuge im Team bearbeitet unter Anwendung geeigneter Vorgehensweisen zur Projekt- und Arbeitsorganisation. Insbesondere werden Vorgehensweisen aus den Bereichen Mensch-Maschine Interaktion (Prototyping, Entwicklung, Durchführung und Auswertung von Benutzerstudien, Human-Centered Software Engineering, Feldtests, Fokusgruppen), verteilte Systeme (Architekturentwurf verteilter Anwendungen, Verteilung von Funktionalität, Protokollentwurf), und Software Engineering (Agile Entwicklungsprozesse) eingesetzt. Das Vorgehen deckt sich soweit möglich mit bestehender Praxis aus Industrie und Forschung.</p> <p>Teams von in der Regel 2-3 Studierenden bearbeiten in der Übung gemeinsam und systematisch ein kleineres Projekt, das in mehrere Arbeitspakete strukturiert ist. Die genaue Aufgabenstellung wird zu Beginn der Lehrveranstaltung schriftlich in Form einer Zielvorgabe mit minimalen Eigenschaften als Bestehenskriterien vorgegeben.</p> <p>Bei der Bearbeitung des vorlesungsbegleitenden Projekts werden folgende Engineering-Aktivitäten für die 1.) Infrastrukturkomponenten bzw. für die 2.) mobile Anwendung abgedeckt:</p> <p>1. Analyse</p> <p>1.) Detaillierte Festlegung der Anforderungen an das System. Beachtung der Grundprinzipien Präzision, Vollständigkeit und Konsistenz. Der Inhalt umfasst das Systemmodell als Übersicht, die geeignete Beschreibung der Systemumgebung mittels geeigneter Werkzeuge, sowie die Erfassung und Dokumentation funktionaler und nicht-funktionaler Anforderungen.</p> <p>2.) Für die Mobile Anwendung sind, zusätzlich zu den genannten Aufgaben, Prototyping-Methoden einzusetzen (z.B. Wizard-of-Oz) bzw. Studien zur Identifikation der Nutzergruppen (z.B. Interviews) durchzuführen.</p> <p>2. Entwurf</p> <p>1.) Hauptbestandteil ist ein systematischer Grobentwurf eines</p>

Systems, das die in der Analyse ermittelten Anforderungen bestmöglich erfüllt. Auf dieser Basis wird ein detaillierter Entwurf ausgearbeitet, der mit der Problemstellung angemessenen, domänenspezifischen Werkzeugen und Vorgehensweisen das umzusetzende System spezifiziert und dokumentiert.

2.) Die mobile Anwendung ist, im Gegensatz zum Hauptsystem, mittels Prototyping-Methoden agil und iterativ zu entwerfen und zu validieren. Dazu sind z.B. Methoden zur Erstellung horizontaler bzw. vertikaler High-Level/Low-Level Prototypen aus dem Bereich der Mensch-Maschine-Interaktion einzusetzen.

3. Umsetzung

1.) Im Rahmen der Umsetzung erfolgt die tatsächliche Realisierung des entworfenen Systems. Das System besteht in der Regel aus Software- und Hardware-Komponenten. Zur Realisierung sind bestehende, konfigurierbare Softwarebausteine mit eigener Software zu ergänzen und zu einem lauffähigen Gesamtsystem zu integrieren. Hierzu werden Methoden aus dem Bereich der verteilten Systeme, z.B. Architekturentwurf, oder der vernetzten Systeme, z.B. Socket-Programmierung, verwendet.

2.) Die Umsetzung der Mobilien Anwendung wird durch spezielle Frameworks und Entwicklungssysteme aus dem Bereich mobiler Anwendungen unterstützt.

4. Validierung

1.) Validierung und Verifikation der Ergebnisse von Entwurf und Umsetzung auf Grundlage der durch Analyse bestimmten Anforderungen.

2.) Die mobile Anwendung ist durch geeignete Methoden aus dem Bereich der Mensch-Maschine-Interaktion zu evaluieren und die Ergebnisse sind kritisch zu diskutieren. Hierzu können z.B. Beobachtung, Fragebögen, Effizienz- und Fehlermessungen bei der Interaktion eingesetzt werden.

Jedes Arbeitspaket kann eine oder mehrere dieser Aktivitäten umfassen und jede Aktivität kann Gegenstand eines oder mehrerer Arbeitspakete sein. Dabei müssen alle Aktivitäten durch Arbeitspakete adäquat abgedeckt sein. In den einzelnen Arbeitspaketen kommen projekt- und domänenspezifische Werkzeuge und Methoden zum Einsatz z.B. zum Test von Client/Server-Systemen, Schnittstellenbeschreibungssprachen, Service Description Languages.

Zu allen Arbeitspaketen werden Arbeitseinheiten definiert, deren Aufwand abgeschätzt und deren Realisierung z.B. anhand einer Gantt-Chart organisiert und durchgeführt. Das Ergebnis jedes Arbeitspakets wird durch einen kurzen Bericht dokumentiert, ggf. begleitet von Software. Aus dem Bericht sind auch Aufwandsabweichungen und Korrekturen vorangegangener Arbeitspakete ersichtlich.

Jedes Arbeitspaket schließt mit einem Kurzvortrag in der nächsten Einheit ab. Die Teams werden durch ein festes wöchentliches Treffen mit dem Betreuer unterstützt.

	Die Lehrveranstaltung schließt mit einem Abschlusskolloquium ab, in dem das fertig entwickelte System präsentiert und abgenommen wird.
Studien-/Prüfungsleistungen:	Schriftliche Dokumentation in Form eines technischen Berichts, mündliche Präsentationen zu den Arbeitspaketen die durch geeignete Medien (z.B. Folien) unterstützt werden, dokumentierter und funktionsfähiger Quelltext inkl. aller zur Demonstration notwendigen Informationen, sowie einer Systemdemonstration im Rahmen des Abschlusskolloquiums.
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation mit Projektor, Gruppenarbeit, Wiki
Literatur:	Wird vom Dozent / von der Dozentin bekannt gegeben Die Literatur wird in Abhängigkeit der konkreten Aufgabenstellung ausgewählt und bekanntgegeben.
Anwesenheitspflicht	<p>Für die Vorlesung und die Übung im Rahmen der Lehrveranstaltung besteht Anwesenheitspflicht. In der Vorlesung findet eine wissenschaftlich-technische Einführung zu den Themen der Lehrveranstaltung statt, diese werden in den anschließenden Übungen direkt mit der Übungsleitung praktisch umgesetzt.</p> <p>Begründung: In der Lehrveranstaltung arbeiten die Teams von Studierenden an einem größeren Projekt über das ganze Semester hindurch. Es zu jedem Arbeitspaket bzw. Themengebiet ein Kolloquium statt, in denen über die Fortschritte berichtet, aufgetretene Probleme ausgetauscht und ihre Lösungen diskutiert werden; am Ende findet ein Abschlusskolloquium statt.</p> <p>Wird keine umfassende Anwesenheit bei den Kolloquien gefordert, wird die Kompetenz nicht geübt, vor anderen Studierenden zu präsentieren und auf ihre Fragen und Anmerkungen (und nicht nur die des Dozenten) einzugehen und diese zu diskutieren. Die Kompetenz, die präsentierten Inhalte zu analysieren, bewerten und kritisch zu diskutieren ist eine wesentliche Anwendung der Lehrveranstaltungsinhalte die nur bei Präsenz eingeübt werden kann. Die vereinzelte Abwesenheit aus nicht vom Studierenden zu vertretenden und nachgewiesenen Gründen ist möglich.</p> <p>Darüber hinaus kann die spezifische Aufgabenstellung besondere Ausstattung erfordern, die nur in den Laboren und Räumen der Universität in geeigneter Weise zur Verfügung steht. Ferner ist ggf. eine direkte Betreuung und Unterweisung an speziellen Geräten notwendig. Daher ist bei spezifischer Aufgabenstellung eine Bearbeitung außerhalb dieses Kontextes nicht möglich und die Anwesenheit dann zwingende Voraussetzung für die erfolgreiche Bearbeitung. Andernfalls ist die erfolgreiche Teilnahme an der Lehrveranstaltung gefährdet.</p>

Modulbezeichnung:	Software Product-Line Engineering
ggf. Kürzel	5612
Studiensemester:	5.-6. Semester
Moduldauer:	1 Semester
Häufigkeit des Modulangebots	In der Regel im Sommersemester
Modulverantwortliche(r):	Prof. Dr. Sven Apel (LST für Informatik mit SP Softwareproduktlinien)
Dozent(in):	Prof. Dr. Sven Apel (LST für Informatik mit SP Softwareproduktlinien)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	2V+2Ü
Arbeitsaufwand:	60 Std. Präsenz + 50 Std. Übungsaufgaben + 70 Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	6
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen:	Softwaretechnik für Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die Vorteile und Nachteile des Produktlinienansatzes sowie klassischer und moderner Programmiermethoden wie z.B. Präprozessoren, Versionsverwaltungssysteme, Komponenten, Frameworks, Feature-Orientierung, Aspekt-Orientierung.</p> <p>Fähigkeiten: Die Studierenden haben die Befähigung zur Bewertung, Auswahl und Anwendung moderner Programmierparadigmen, Techniken, Methoden und Werkzeuge erlangt, insbesondere in Hinblick auf die Entwicklung von Softwareproduktlinien</p> <p>Kompetenzen: Die Studierenden erwerben Urteilsvermögen über den Einsatz von Programmiermethoden für die Entwicklung von Softwareproduktlinien.</p>
Inhalt:	<p>Einführung in die Problematik der Entwicklung komplexer, maßgeschneiderter Software-Systeme am Beispiel von eingebetteten Datenmanagement-Systemen</p> <p>Modellierung und Implementierung von Programmfamilien, Produktlinien und Softwarefabriken</p> <p>Wiederholung von Grundkonzepten der Software-Technik (u. a. Separation of Concerns, Information Hiding, Modularisierung, Strukturierte Programmierung und Entwurf)</p> <p>Einführung in verschiedene klassische und erweiterte Sprachen</p>

	<p>und Werkzeuge zur Entwicklung von Softwareproduktlinien u. a. Präprozessoren, Frameworks, Komponenten, Subjekte, Schichten, Aspekte, Kollaborationen, Rollen, etc.</p> <p>Vergleich grundlegender Konzepte, Methoden, Techniken und Werkzeuge der vorgestellten Ansätze</p> <p>Kritische Diskussion von Vor- und Nachteilen der einzelnen Ansätze sowie ihrer Beziehung untereinander</p> <p>In der Veranstaltung werden aktuelle Forschungsergebnisse des Dozenten sowie anderer Forscher besprochen, angewendet und evaluiert</p>
Studien-/Prüfungsleistungen:	ca. 15 min mündliche Prüfung
Modulnote:	Die Modulnote entspricht der Note der Prüfung.
Medienformen:	Präsentation und Beamer, Tafel
Literatur:	<p>Vorlesungsfolien</p> <p>Generative Programming. Methods, Tools and Applications: Krzysztof Czarnecki, Ulrich Eisenecker, Addison Wesley, 2000</p>

Modulbezeichnung:	Software Verification
ggf. Kürzel	5843
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	In der Regel im Wintersemester
Modulverantwortliche(r):	Prof. Dr. Dirk Beyer (LST für Informatik mit SP Softwaresysteme)
Dozent(in):	Prof. Dr. Dirk Beyer (LST für Informatik mit SP Softwaresysteme)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	2V+ 1Ü + 2 Projekt
Arbeitsaufwand:	75 Std. Präsenz + 30 Std. Übungsaufgaben + 30 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung + 75 Std. Projektarbeit
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Module der Modulgruppe Praktische Informatik
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden erlernen grundlegende Prinzipien und erwerben Kenntnisse über moderne Techniken für die Bewertung und Verbesserung von Methoden zur Verifikation von Softwaresystemen.</p> <p>Fähigkeiten: In den Übungen vertiefen die Studenten das in der Vorlesung behandelte Wissen bei der Lösung von Übungsaufgaben. Im Semesterprojekt entwerfen und implementieren die Studenten eigene Komponenten für ein Software-Verifikationswerkzeug.</p> <p>Kompetenzen: Die Studenten können formale Techniken als praktisches Mittel zur Gestaltung und zur Analyse von Softwaresystemen in der industriellen Praxis einsetzen. Die Anwendungen konzentrieren sich auf die Analyse von Software-Quelltext.</p>
Inhalt:	Die Vorlesung behandelt wichtige Prinzipien und Verfahren der Softwareanalyse, insbesondere Datenflussanalyse, Software Model Checking, testen. Die Studenten lernen formale Techniken als praktisches Mittel zur Analyse von Softwaresystemen kennen. Hervorgehoben wird Werkzeugunterstützung. Die Anwendungen konzentrieren sich auf die Analyse von Quelltext. Im Semesterprojekt entwerfen und

	<p>implementieren die Studenten eigene Komponenten für ein Software-Analysewerkzeug</p> <p>Kurzübersicht zur Vorlesung:</p> <ul style="list-style-type: none"> • Programmanalyse, Datenflussanalyse • Abstract Domains und Abstract Interpretation • Software Model Checking, gegenbeispielbasierte Abstraktionsverfeinerung • Generierung von Programminvarianten • Verifikation endlicher Automaten • Datenstrukturen für die Repräsentierung von endlichen Zustandsmengen • Verifikation unendlicher Zustandsmengen, Echtzeitsysteme • Datenstrukturen für die Repräsentation unendlicher Zustandsmengen • Anwendungen von Theorembeweisern
Studien-/Prüfungsleistungen:	<p>Portfolio:</p> <ul style="list-style-type: none"> • Durchführung des Semesterprojektes, letzteres nachgewiesen durch praktische Leistung bei der selbständigen Erarbeitung, Implementierung und Präsentation der eigenen Softwarekomponente, sowie durch die Abgabe des Projektberichtes mit Erklärung der Konzepte und der Implementierung. • Ca. 30-min. mündliche Prüfung
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Tafel, Beamer
Literatur:	<p>F. Nielson, H. R. Nielson, C. Hankin. Principles of Program Analysis. Springer, 2005.</p> <p>E. M. Clarke, O. Grumberg and D. Peled. Model Checking. MIT Press, 2000.</p> <p>G. J. Holzmann. The SPIN Model Checker: Primer and Reference Manual. Addison-Wesley, 2003.</p>

Modulbezeichnung:	Stochastische Simulation
ggf. Kürzel	5812
Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Thomas Müller-Gronbach (LST für Mathematische Stochastik und ihre Anwendungen)
Dozent(in):	Prof. Dr. Thomas Müller-Gronbach (LST für Mathematische Stochastik und ihre Anwendungen)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	3V + 1Ü
Arbeitsaufwand:	75 Std. Präsenz + 75 Std. Übungsaufgaben + 60 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	7
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Mathematik in Technischen Systemen I-III, Grundlagen der Informatik für Mobile und Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Grundlegende Algorithmen der stochastischen Simulation, ihre Eigenschaften und typische Anwendungen.</p> <p>Kompetenzen: Effiziente Implementierung von Standardverfahren zur direkten Simulation, zur Verteilungssimulation und zur Simulation von Markov-Prozessen. Darstellung und Interpretation von Simulationsergebnissen im Rahmen der Stochastik und Statistik.</p>
Inhalt:	Das Verfahren der direkten Simulation Simulation von Verteilungen Methoden der Varianzreduktion Markov Chain Monte Carlo
Studien-/Prüfungsleistungen:	90-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bekannt gegeben
Modulnote:	Die Modulnote entspricht der Note der Prüfung.

Medienformen:	Präsentation und Beamer, Tafel
Literatur:	Müller-Gronbach, Novak, Ritter: Monte Carlo-Algorithmen, Springer, 2012.

Modulbezeichnung:	Text Mining Project
ggf. Kürzel	5981
Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	unregelmäßig
Modulverantwortliche(r):	Prof. Dr. Siegfried Handschuh (LST für Informatik mit SP Digital Libraries and Web Information Systems)
Dozent(in):	Prof. Dr. Siegfried Handschuh (LST für Informatik mit SP Digital Libraries and Web Information Systems)
Sprache:	Deutsch oder Englisch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	3V + 3Ü
Arbeitsaufwand:	90 Std. Präsenz + 60 Std. Übungsaufgaben + 90 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	8
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Lineare Algebra, Wahrscheinlichkeitsrechnung, Programmierkenntnisse in Python
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden lernen in der praktischen Anwendung grundlegende Konzepte und die wichtigsten Methoden zur Analyse von Textdaten.</p> <p>Kompetenzen: Die Studierenden erwerben die Grundkompetenzen in Python und der NLTK (Natural Language Toolkit) Bibliothek. Diese Kompetenz erlaubt die Extraktion nützlicher Information aus unstrukturierten Texten, um damit eine breite Palette von realen Anwendungen anzugehen.</p>
Inhalt:	<p>Der Kurs bietet eine leicht zugängliche Einführung in das Text Mining und die Verarbeitung natürlicher Sprache (NLP). Das Thema erlaubt eine Vielzahl von Anwendungen, von der automatischen Worterkennung und Email-Filterung bis hin zur automatischen Zusammenfassung und Übersetzung. Die Teilnehmer lernen, wie man Python-Programme erstellt, um große Sammlungen unstrukturierter Texte automatisch zu verarbeiten. Ebenso, wie man Sprach-Ressourcen (reich annotierte Datensätze) mittels einer umfassenden Palette an linguistischer Datenstrukturen verwendet. Die Teilnehmer lernen die wichtigsten Algorithmen für die Analyse des Inhalts und der Struktur schriftlicher Kommunikation kennen. Dies wird vermittelt</p>

	<p>anhand umfangreicher Beispiele und Übungen.</p> <p>Beispielsweise lernen die Teilnehmer:</p> <ul style="list-style-type: none"> • die Informationsgewinnung aus unstrukturierten Texten, zur Themen-Erkennung (Topic Detection) oder der Identifikation wichtiger Begriffe (Named Entities) • Die Analyse linguistischer Strukturen im Text; einschließlich Parsing und semantischer Analyse • Zugriff auf linguistische Datenbanken inklusive WordNet und Treebanks • Die Integration von Techniken aus so unterschiedlichen Bereichen wie der Linguistik und der künstlichen Intelligenz <p>Der Kurs vermittelt praktische Fähigkeiten in der Verarbeitung natürlicher Sprache mit Hilfe der Programmiersprache Python und dem Natural Language Toolkit (NLTK).</p> <p>Mögliche Projektarbeiten umfassen die automatische Text-Analyse Sozialer Medien (bspw Twitter), die Analyse multilingualer Nachrichtenquellen, die Erzeugung von Sprachressourcen, oder die Erzeugung eines Wissensgraphs mittels Wikipedia.</p> <p>Die folgenden Inhalte werden im Detail geboten:</p> <ul style="list-style-type: none"> • Language Processing and Python • Accessing Text Corpora and Lexical Resources • Processing Raw Text • Categorizing and Tagging Words • Learning to Classify Text • Extracting Information from Text • Analyzing Sentence Structure • Building Feature-Based Grammars • Analyzing the Meaning of Sentences • Managing Linguistic Data
Studien-/Prüfungsleistungen:	Projektarbeit bestehend aus Source-Code, schriftliche Ausarbeitung in Form eines technischen Berichts und Präsentation der Arbeit
Modulnote:	Die Modulnote entspricht der Note der Prüfung.
Medienformen:	Tafel, Projektor, Rechner
Literatur:	<ul style="list-style-type: none"> • Steven Bird, Ewan Klein and Edward Loper (2009), <i>Natural Language Processing with Python</i>, O'Reilly Media • Eigenes Skriptum

Modulbezeichnung:	Verteilte Systeme
ggf. Kürzel	5402
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Hermann de Meer (LST für Informatik mit SP Rechnernetze und Rechnerkommunikation)
Dozent(in):	Prof. Dr. Matthias Kranz (LST für Informatik mit SP Eingebettete Systeme) Prof. Dr. Hermann de Meer (LST für Informatik mit SP Rechnernetze und Rechnerkommunikation)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	2V+1Ü
Arbeitsaufwand:	45 Std. Präsenz + 35 Std, Übungen + 70 Std. Nachbearbeitung und Prüfungsvorbereitung
ECTS-Leistungspunkte:	5
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Grundlagen der Informatik für Mobile und Eingebettete Systeme, Rechnerarchitektur
Angestrebte Lernergebnisse:	<p>Kenntnisse: Kenntnis von Grundkonzepten von Betriebssystemen, Grundverständnis von Problemen und Algorithmen im Bereich verteilter Datenverarbeitung, Kenntnis der wichtigsten Werkzeuge.</p> <p>Fähigkeiten: Implementierung verteilter Algorithmen, einfache Systemprogrammierung in Java und C</p> <p>Kompetenzen: Entwurf und Analyse komplexer Verteilter Applikationen</p>
Inhalt:	Grundlegende Modelle verteilter Systeme (synchrone, asynchrone Systeme, Fehlermodelle etc.), logische Zeit und Zeitsynchronisation, Kooperation, Zugriffskonflikte, Deadlocks, Relevante Grundkonzepte von Betriebssystemen wie Prozesse Threads, Schutzmechanismen, Kommunikationsmechanismen, C-Programmiermodell, Middleware und Verfahren zur verteilten Ausführung, Anwendungen (z.B. verteilte Filesysteme)
Studien-/Prüfungsleistungen:	90 min Klausur

Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer, Tafel (oder Labor/Rechner/...)
Literatur:	G. Coulouris, J. Dollimore, T. Kindberg, Distributed Systems, Concepts and Design Andrew S. Tanenbaum, Modern Operating Systems, 2/E, Prentice Hall

Modulbezeichnung:	Wavelet-basierte Methoden in der Bildverarbeitung
ggf. Kürzel	
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Wintersemester
Modulverantwortliche(r):	Prof. Dr. Brigitte Forster-Heinlein (Professur für Angewandte Mathematik)
Dozent(in):	Dr. Johannes Nagler (Professur für Angewandte Mathematik)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	2V+2Ü
Arbeitsaufwand:	30+30 Std. Präsenz, 60+60 Std. Eigenarbeitszeit (Vor- und Nachbearbeitung sowie Teilnahme an der Übung)
ECTS-Leistungspunkte:	6
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Bildverarbeitung
Angestrebte Lernergebnisse:	<p><u>Kenntnisse:</u> Die Studierenden kennen nach dieser Vorlesung die Konzepte der Multiskalenanalyse mit Wavelets. Sie verstehen die diskrete Wavelet-Transformation in 1D und 2D und deren Anwendung auf Bilder. Insbesondere kennen und verstehen sie Verfahren zur Kompression und zum Entrauschen von Bildern.</p> <p><u>Fähigkeiten und Kompetenzen:</u> Die Studierenden können Wavelet-basierte Verfahren implementieren, modifizieren und in gewissem Rahmen auch neu entwickeln. Insbesondere können sie Wavelet-basierte Verfahren zur Kompression und zum Entrauschen von Bildern einsetzen und mit anderen Verfahren vergleichen und bewerten. Die Studierenden haben die Kompetenz, mit Wavelet-basierten Verfahren theoretisch und praktisch umzugehen.</p>
Inhalt:	Mathematische Grundlagen: Fourier-Transformation in L^1 und L^2 , Multiskalenanalyse mit Wavelets in L^2 , Diskrete Wavelet-Transformation, Kompressionsverfahren (JPEG, JPEG2000) und Entrauschen von Bildern (Wiener Filter, Wavelet Shrinkage)
Studien-/Prüfungsleistungen:	60-minütige Klausur oder mündliche Prüfung (ca. 20 Minuten); die genaue Prüfungsart wird zu Beginn des Semesters durch Aushang und auf den Internetseiten der Fakultät bzw. in der

	Vorlesung bekannt gegeben.
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation, Beamer, Übungsblätter
Literatur:	<p>S. Mallat: A Wavelet Tour of Signal Processing, Academic Press, 3rd Edition, 2009</p> <p>T. F. Chan, J. Shen: Image Processing and Analysis. SIAM, 2005</p> <p>K. S. Thygarajan: Still Image and Video Compression with Matlab, Wiley-IEEE Press, 2010</p>

Modulbezeichnung:	Web-Engineering
ggf. Kürzel	5430
Empfohlenes Studiensemester:	5.-6. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Prof. Dr. Harald Kosch (LST für Informatik mit SP Verteilte Informationssysteme)
Dozent(in):	Prof. Dr. Harald Kosch (LST für Informatik mit SP Verteilte Informationssysteme)
Sprache:	Deutsch
Zuordnung zum Curriculum	MES Wahlpflichtmodul
Lehrform/SWS:	2V+2Ü
Arbeitsaufwand:	60 Std. Präsenz + 50 Std. Übungsaufgaben + 70 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	6
Voraussetzungen nach Prüfungsordnung	Keine
Empfohlene Voraussetzungen:	Grundlagen der Informatik für Mobile und Eingebettete Systeme
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden kennen die Techniken und Konzepte moderner Web Anwendungen. Sie kennen die aktuellen Entwicklungsframeworks. Sie kennen die einzelnen Schritte des SW-prozesses für Web-Anwendungen und die Unterschiede zum herkömmlichen SW-Prozess. Sie kennen die Grundkonzepte des Semantic Webs und die Techniken des Web 2.0.</p> <p>Fähigkeiten: Die Studierenden beherrschen den Web-engineering Softwarezyklus und können komplexere Webanwendungen vor allem in die Java-basierten Frameworks (Struts und JSF) nach dem erlernten Softwarezyklus umsetzen.</p> <p>Kompetenzen: Die Studierenden haben die Kompetenz komplexere Web-Anwendungen zu entwerfen und zu implementieren, Sie können den Betrieb und Wartung von Web-Anwendungen durchführen und die Qualität von Anwendungen beurteilen und verbessern.</p>
Inhalt:	Das Modul Web-Engineering konzentriert sich auf die Vermittlung der notwendigen Konzepte, Techniken und

	<p>Architekturen, welche die Umsetzung von komplexen Web Anwendungen gewährleistet. Ein wesentlicher Schwerpunkt ist die Einführung des bekannten Model View Controller (MVC) Konzeptes, welches anhand der Anwendung aktueller Entwicklungsframeworks (Struts, Java Server Faces) demonstriert wird. Aufbauend auf diese Frameworks werden spezifische Themen eines Web-Entwicklungsprozesses (ähnlich zu SW-Projekten) behandelt: Planung, Modellierung, Wartung, etc.</p> <p>Inhaltliche Gliederung: Web-Informationssysteme: Architektur und Implementierungen Funktionsweise des CGI-basierte Ansätze Java Server Pages und Java Servlets : Grundprinzipien und deren Unterscheidung. Behandlung des Konzeptes Session Tracking und dessen Umsetzungsarten. Entwicklungsframeworks für Web-Anwendungen: Struts 1.x, 2.0 und deren Unterschiede, Java Server Faces (JSF) Alternative Technologien für einzelne Bereiche des MVC-Konzeptes: Facelets, Ruby on Rails Modellierung von Web-Anwendungen (Content, Hypertext, Präsentation, Kontext Adaptation) Entwicklungsprozess von Web-Anwendungen (SW- und Datenorientierte Entwürfe, UML vs. ER basierte) Einführung in das Semantic Web Standards und Einsatz (SOAP, WSDL, UDDI etc.) Aktuelle Web 2.0 Technologien Asynchrone Webtechnologien: Ajax</p>
Studien-/Prüfungsleistungen:	90 min schriftliche Klausur
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	<p>Folienorientierte Vorlesung, Tafelbenützung bei Beispielen, zusätzlichen Erläuterungen und zu erklärenden Sachverhalten: Wöchentliche Übungen in kleinen Gruppen. Dabei werden Präsenzaufgaben sowie die Musterlösungen zu den Übungsaufgaben vorgerechnet Erwartete Aktivitäten der Studierenden: Mitarbeit bei Präsenzübungen, Übungsaufgaben, selbständiges Studium von sekundärer Literatur Folienskript ist vorhanden und über StudIP zugänglich.</p>
Literatur:	<p>Gerti Kappel, Birgit Pröll, Siegfried Reich, Werner Retzischzegger : "Web Engineering : Systematische Entwicklung von Web-Anwendungen", dpunkt.verlag, Oktober 2003, ISBN 3-89864-234-8</p>

Schlüsselqualifikationen und Fremdsprache

Pflicht und Wahlpflicht (Auswahl)

Weitere anrechenbare Schlüsselqualifikationen unter

www.fim.uni-passau.de/studium/pruefungen-und-modulkataloge/anrechenbarkeit-und-modulkataloge/

Modulbezeichnung:	FFA Aufbaustufenmodul 1
ggf. Kürzel	9060 (Nummer wird vom Sprachenzentrum vergeben)
Empfohlenes Studiensemester:	3. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	In der Regel jedes Semester
Modulverantwortliche(r):	Sprachenzentrum
Dozent(in):	Lektor/Lektorin des Sprachenzentrums
Sprache:	Englisch
Zuordnung zum Curriculum	MES Fremdsprache
Lehrform/SWS:	2 SWS / Sprachübung
Arbeitsaufwand:	30 Std. Präsenz, Präsentationen, Rollenspiele, 60 Std. Vor- und Nachbereitung
ECTS-Leistungspunkte:	3
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Sprachkenntnisse auf dem Niveau B1 des Gemeinsamen Europäischen Referenzrahmens für Sprachen
Angestrebte Lernergebnisse:	<p>Er/Sie versteht den wesentlichen Inhalt alltagspraktischer und berufsbezogener Hörtexte mit Bezug zu Themen der Informatik, z.B. Reden, Vorträge und Vorlesungen.</p> <p>Er/Sie kann sich mündlich zu einer Vielfalt fachlicher Themen angemessen äußern und an entsprechenden Gesprächen aktiv beteiligen, wobei er/sie auch zu einem gewissen Grad komplexe Satzstrukturen und fachspezifisches Vokabular benutzt.</p> <p>Kompetenzerwerb: Die Studierenden verbessern ihr Englisch durch Konversation, Diskussionen und weitere Sprachübungen über studien- und berufsspezifische Themen, z.B. die Rolle des Internets in China oder den Einsatz von IT-Technologien in Studium und Berufsleben.</p> <p>Fähigkeiten: Die Studierenden lernen, wie man eine gute Präsentation zusammenstellt und überzeugend vorträgt; durch Rollenspiele lernen sie, in wechselnden, auch interkulturellen Kontexten wirksam in der Fremdsprache zu kommunizieren, und durch die Arbeit mit Audio- und Videomaterialien erweitern sie ihr fachspezifisches Vokabular und verbessern ihr Hörverstehen.</p> <p>Kompetenzen: soziale, analytische, sprachlich-kommunikative und interkulturelle Kompetenz.</p>
Inhalt:	Jeder/e Studierende hält eine fachbezogene Präsentation vor

	<p>seinen/ihrer Kommilitonen zum Thema des Tages (Länge 15-20 Min.).</p> <p>Jeder/e Studierende hält eine kleine Präsentation (Reportage) über neueste Entwicklungen im Bereich IT-/Computer-/Medien-Technologie (Länge 5-10 Min.).</p> <p>Fachbezogene Themengebiete werden in Kleingruppen diskutiert, wobei sowohl der Dozent/die Dozentin als auch Studierende die Rolle eines Moderators einnehmen können.</p> <p>Rollenspiele und Audio-/Videomaterialien werden eingesetzt, um das jeweilige Thema besser zu erklären und Diskussionen vorzubereiten und zu begleiten.</p>
Studien-/Prüfungsleistungen:	45-minütige Klausur zum Hörverständnis und zur Sprechfertigkeit am Ende des Semesters
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Multimediaanwendungen, z.B. Internet, Video- und Audiomaterialien.
Literatur:	Keine

Modulbezeichnung:	FFA Aufbaustufenmodul 2
ggf. Kürzel	9060 (Nummer wird vom Sprachenzentrum vergeben)
Empfohlenes Studiensemester:	3. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	In der Regel jedes Semester
Modulverantwortliche(r):	Sprachenzentrum
Dozent(in):	Lektor/Lektorin des Sprachenzentrums
Sprache:	Englisch
Zuordnung zum Curriculum	MES Fremdsprache
Lehrform/SWS:	2 SWS / Sprachübung
Arbeitsaufwand:	30 Std. Präsenz, Präsentationen, Rollenspiele, 60 Std. Vor- und Nachbereitung
ECTS-Leistungspunkte:	3
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	Sprachkenntnisse auf dem Niveau B1 des Gemeinsamen Europäischen Referenzrahmens für Sprachen
Angestrebte Lernergebnisse:	<p>Er/Sie versteht den wesentlichen Inhalt alltagspraktischer und berufsbezogener Hörtexte mit Bezug zu Themen der Informatik, z.B. Reden, Vorträge und Vorlesungen.</p> <p>Er/Sie kann sich mündlich zu einer Vielfalt fachlicher Themen angemessen äußern und an entsprechenden Gesprächen aktiv beteiligen, wobei er/sie auch zu einem gewissen Grad komplexe Satzstrukturen und fachspezifisches Vokabular benutzt.</p> <p>Kompetenzerwerb: Die Studierenden verbessern ihr Englisch durch Konversation, Diskussionen und weitere Sprachübungen über studien- und berufsspezifische Themen, z.B. die Rolle des Internets in China oder den Einsatz von IT-Technologien in Studium und Berufsleben.</p> <p>Fähigkeiten: Die Studierenden lernen, wie man eine gute Präsentation zusammenstellt und überzeugend vorträgt; durch Rollenspiele lernen sie, in wechselnden, auch interkulturellen Kontexten wirksam in der Fremdsprache zu kommunizieren, und durch die Arbeit mit Audio- und Videomaterialien erweitern sie ihr fachspezifisches Vokabular und verbessern ihr Hörverstehen.</p> <p>Kompetenzen: soziale, analytische, sprachlich-kommunikative und interkulturelle Kompetenz.</p>
Inhalt:	Jeder/e Studierende hält eine fachbezogene Präsentation vor

	<p>seinen/ihrer Kommilitonen zum Thema des Tages (Länge 15-20 Min.).</p> <p>Jeder/e Studierende hält eine kleine Präsentation (Reportage) über neueste Entwicklungen im Bereich IT-/Computer-/Medien-Technologie (Länge 5-10 Min.).</p> <p>Fachbezogene Themengebiete werden in Kleingruppen diskutiert, wobei der Dozent/die Dozentin /die Dozentin als auch Studierende die Rolle eines Moderators einnehmen können.</p> <p>Rollenspiele und Audio-/Videomaterialien werden eingesetzt, um das jeweilige Thema besser zu erklären und Diskussionen vorzubereiten und zu begleiten.</p>
Studien-/Prüfungsleistungen:	<p>45-minütige Klausur zum Hörverständnis und zur Sprechfertigkeit am Ende des Semesters;</p> <p>mündliche Gruppenprüfung mit insgesamt max. 3 Kandidaten bzw. Kandidatinnen (ca. 15. Min. je Kandidat bzw. Kandidatin)</p>
Modulnote:	Die Modulnote entspricht dem arithmetischen Mittel aus den Noten beider Prüfungsteile.
Medienformen:	Multimediaanwendungen, z.B. Internet, Video- und Audiomaterialien.
Literatur:	Keine

Modulbezeichnung:	Softskills im IT-Umfeld
ggf. Kürzel	
Empfohlenes Studiensemester:	1. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Semester
Modulverantwortliche(r):	Birke, Lehrbeauftragte des Zentrums für Schlüsselqualifikationen
Dozent(in):	Birke/Strunz
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Wahlpflicht Schlüsselqualifikation
Lehrform/SWS:	Blockveranstaltung (4 Tage)
Arbeitsaufwand:	30 Std. Präsenz + 60 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	3
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	keine
Angestrebte Lernergebnisse:	<p>Kenntnisse: die Studierenden kennen grundlegende Methoden und Fähigkeiten, um Fachwissen der Informatik in die Praxis zu übertragen und zu präsentieren. Sie gewinnen Einblicke in typische Herausforderungen im Berufsleben als IT-Dienstleister</p> <p>Kompetenzen: die Studierenden besitzen die Fähigkeit, komplexe Fachinhalte durch Anwendung von Soft Skills übersichtlich und strukturiert darzustellen und können strukturierte Problemlösungsstrategien in Praxissituationen anwenden.</p>
Inhalt:	Das Seminar ergänzt mit Focus auf Praxisrelevanz die universitäre Ausbildung. Zentrale Themen sind Methoden für strukturierte Problemlösung, Teamarbeit und –führung, Umgang mit Konfliktsituationen im Team, effektives Halten von Präsentationen und überzeugende Darstellung komplexer Inhalte. Dies wird erreicht über Einüben von Fallstudien, Leiten von und Teilhabe an Diskussionen, kurze Vorträge vor dem Plenum, Lösen von Übungsaufgaben sowie Gruppenarbeit.
Studien-/Prüfungsleistungen:	Präsentation (ca. 20 Minuten)
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Beamer, Tafel
Literatur:	Wird vom Dozenten oder der Dozentin bekannt gegeben

Modulbezeichnung:	Gewerblicher Rechtsschutz einschließlich Softwareschutz mit Fallstudien zu Patentrecht und Patentrecherche
ggf. Kürzel	5884
Empfohlenes Studiensemester:	1. Semester
Moduldauer	1 Semester
Häufigkeit des Modulangebots	Jedes Sommersemester
Modulverantwortliche(r):	Möhring, Röder
Dozent(in):	Möhring, Röder
Sprache:	Deutsch
Zuordnung zum Curriculum	MES, Wahlpflicht Schlüsselqualifikation
Lehrform/SWS:	2V
Arbeitsaufwand:	30 Std. Präsenz + 60 Std. Nachbearbeitung des Vorlesungsstoffs und Prüfungsvorbereitung
ECTS-Leistungspunkte:	3
Voraussetzungen nach Prüfungsordnung	keine
Empfohlene Voraussetzungen:	keine
Angestrebte Lernergebnisse:	<p>Kenntnisse: Die Studierenden erwerben Kenntnisse des gewerblichen Rechtsschutzes (Patent-, Gebrauchsmuster-, Geschmacksmuster-, Markenrecht) und des Urheberrechts unter besonderer Berücksichtigung des Softwareschutzes zu vermitteln.</p> <p>Fähigkeiten: Die Studierenden werden für den Schutz technischer Innovationen durch Patente, insbesondere im Bereich Software, sensibilisiert. Sie werden befähigt, eigenständige Recherchen in Patentdatenbanken durchzuführen, um den Stand der Technik zu ermitteln.</p> <p>Kompetenzen: Die Studierenden können den Stellenwert des Gewerblichen Rechtsschutzes und dessen Bedeutung für technische Entwicklungen einschätzen und würdigen. Sie sind erster Ansprechpartner für Privaterfinder und mittelständische Gewerbetreibende in Fragen des Patent- und Markenschutzes.</p>
Inhalt:	<ul style="list-style-type: none"> - Wie liest man eine Patentschrift? (Mit Beispielen verschiedener Schriftenarten) - Gemeinsamkeiten und Unterschiede zwischen Patenten und Gebrauchsmustern - Absicherung einer Idee durch gewerbliche Schutzrechte (Patente, Marken, Design), Gesetzesüberblick - Das Markengesetz DE/EU/IR

	<ul style="list-style-type: none"> - Der internationale Patentschutz - einschl. regionaler Patentsysteme - Erfindungen an Hochschulen und Forschungseinrichtungen, Gesetz über Arbeitnehmererfindungen - Softwareschutz, Urheberrechtsgesetz - Die internationale Patentklassifikation - Online-Patentrecherchen in frei verfügbaren Datenbanken (DEPATISnet, DPINFO, DPMApublikationen, ...) und kommerziellen Datenbanken (STN International)
Studien-/Prüfungsleistungen:	ca. 15-minütige mündliche Prüfung am Ende des Semesters
Modulnote:	Modulnote entspricht der Note der Prüfung
Medienformen:	Präsentation und Beamer
Literatur:	<p><i>Wettbewerbsrecht, Gewerblicher Rechtsschutz und Urheberrecht</i> Vorschriftensammlung C.F. Müller Verlag ISBN 9783811432130</p> <p>Horst-Peter Götting: <i>Gewerblicher Rechtsschutz</i> 8. Auflage Verlag C.H. Beck ISBN 9783406557149</p> <p>Joachim Gruber: <i>Gewerblicher Rechtsschutz und Urheberrecht</i> 2. Auflage Niederle Media ISBN 9783867241311</p>