

Dokumenthistorie

17.02.2017 (06) 07:07:21	JN: Korrekturen am Text
30.01.2017 (04) 08:49:41	RS: Workloads
27.01.2017 (03) 18:51:52	AR: Vereinheitlichung
27.01.2017 (03) 11:41:21	Korrekturen von RS und FM übernommen, Formatierung angepasst
24.01.2017 (03) 11:58:06	Korrekturen von JN übernommen
19.01.2017 (02) 17:27:33	ASE Update von Hr. Germann
18.01.2017 (02) 11:04:00	Neue Module per Macro verlinkt
17.01.2017 (02) 14:57:11	Literaturänderungen Mobil Communication, D. Schoop
17.01.2017 (02) 18:03:06	Security Engineering Änderungen übernommen

Fakultät Informationstechnik

Modulhandbuch

Master-Studiengang **Angewandte Informatik**
mit Vertiefung

Autonome Systeme,

Business Intelligence,

Mobile Computing

Inhaltsverzeichnis

Kern- und Pflichtmodule

Modulnummer	Modul	Seite
1. Semester		
AI 101	Sprachen und Automaten	5
AI 102	Security Engineering	7
AI 103, AI 104	Wahlpflichtmodul 1 und 2	9
AI 105	Forschungsprojekt 1	10
2. Semester		
AI 201	Advanced Software Engineering	12
AI 202	IT Management	14
AI 203, AI 204	Wahlpflichtmodul 3 und 4	16
AI 205	Forschungsprojekt 2	17
3. Semester		
AI 301	Masterarbeit	19
AI 302	Publikation	21

Wahlpflichtmodule

der Vertiefungsrichtung **Autonome Systeme**

Modulnummer	Modul	Seite
AI W03	Mobile Communication	27
AI W04	Datenfusion	29
AI W05	Automotive System Design	31
AI W06	Advanced Control	33
AI W07	Systemtheorie	35
AI W08	Programmiermethoden der KI	37
AI W09	Self-Adaptive Software	39
AI W10	Software Tooling Workshop	41

Wahlpflichtmodule der Vertiefungsrichtung **Business Intelligence**

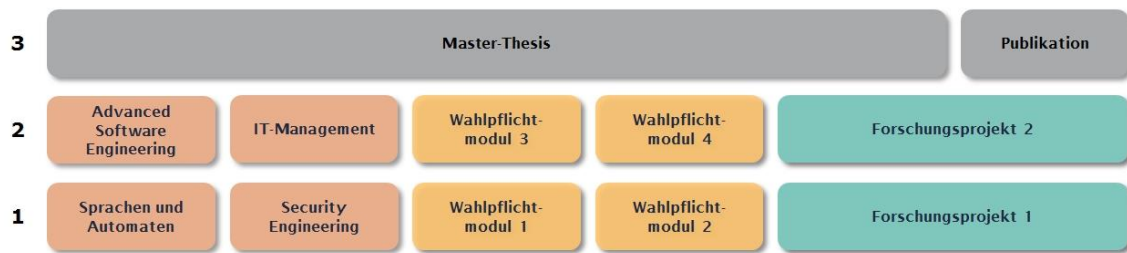
Modulnummer	Modul	Seite
AI W01	Intelligent Data Analytics	23
AI W02	Business Intelligence	25
AI W04	Datenfusion	29
AI W08	Programmiermethoden der KI	37
AI W11	Rechnerorganisation	43
AI W12	Web Services	45
AI W13	Mobile Applikationen	47
AI W14	Data Mining	49
AI W16	Web Technologien	53

Wahlpflichtmodule der Vertiefungsrichtung **Mobile Computing**

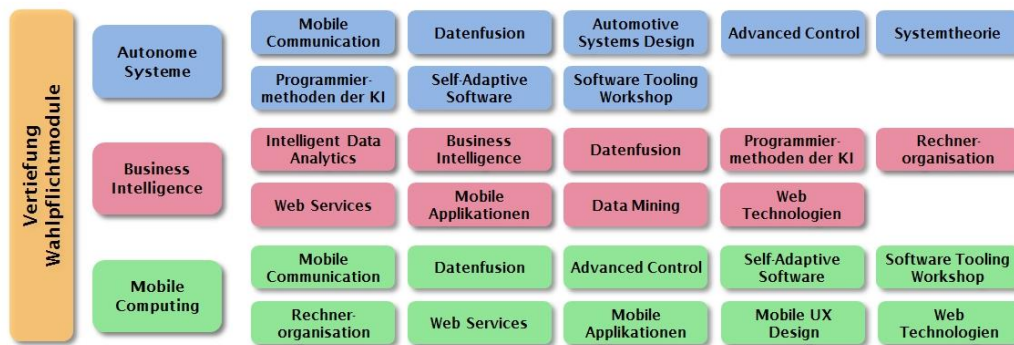
Modulnummer	Modul	Seite
AI W03	Mobile Communication	27
AI W04	Datenfusion	29
AI W06	Advanced Control	33
AI W09	Self-Adaptive Software	39
AI W10	Software Tooling Workshop	41
AI W11	Rechnerorganisation	43
AI W12	Web Services	45
AI W13	Mobile Applikationen	47
AI W15	Mobile UX Design	51
AI W16	Web Technologien	53

Übersicht Modulplan

Studienschwerpunkte **Autonome Systeme, Business Intelligence, Mobile Computing**



Legende: Kernmodule (orange), Vertiefung (gelb), Forschung (hellblau), Thesis (grau)



Modulbeschreibung Sprachen und Automaten

Schlüsselworte: Theoretische Informatik

Zielgruppe:	1. Semester AIM	Modulnummer:	AI 101
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr. Heinrich Weber		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen der Informatik

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- Grammatiken
- Chomsky-Hierarchie
- Syntaxbäume
- Reguläre Sprachen
- Endliche, nicht endliche Automaten
- Reguläre Ausdrücke
- Kontextfreie Sprachen
- Kellerautomaten
- Kontextsensitive und allgemeine Sprachen
- Attributierte Grammatiken
- die grundsätzlichen Möglichkeiten und Grenzen der Algorithmisierbarkeit

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- den Aufbau formaler Sprachen zu beschreiben
- Probleme mit formalen Sprachen und den ihnen zugeordneten Automaten zu lösen
- nichtdeterministischer und deterministischer Vorgänge anhand von Automatenmodellen zu unterscheiden
- Sprachen innerhalb der Chomsky-Hierarchie mit den zugeordneten Automatenmodellen zu klassifizieren

Übergreifende Kompetenzen

Die Studierenden können

- Probleme hinsichtlich ihrer Berechenbarkeit analysieren und
- die Komplexität von Problemen einschätzen

Literaturhinweise:

Hoffmann v., D.: Theoretische Informatik; Hanser Verlag; 2011.

Hedtstück, U.: Einführung in die Theoretische Informatik / Formale Sprachen und Automatentheorie, Oldenbourg Verlag; 2012.

Erk, K; Priese, L: Theoretische Informatik / Eine umfassende Einführung; Springer Verlag; Berlin, Heidelberg, New York,... ; 2000.

Wird angeboten:

in jedem Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung
Leistungskontrolle:	Klausur (90 Minuten)
Anteil Semesterwochenstunden:	3 SWS
Geschätzte studentische Arbeitszeit:	90 Stunden

Lernergebnisse:

Die Studierenden erwerben das notwendige theoretische Wissen, um Probleme mit Formalen Sprachen und Automaten zu beschreiben. Sie verfügen über die Fähigkeit, Methoden zur Beschreibung nichtdeterministischer und deterministischer Vorgänge anzuwenden. Sie können damit verschiedene Fragestellungen aus dem Bereich der Grammatiken und Automaten in Hinblick auf ihre Vollständigkeit, Abgeschlossenheit, Berechenbarkeit und Komplexität beurteilen. Sie erlernen die grundsätzlichen Möglichkeiten und Grenzen der Algorithmisierbarkeit kennen.

Lehr- und Lernform:	Übungen
Leistungskontrolle:	Testat
Anteil Semesterwochenstunden:	1 SWS
Geschätzte studentische Arbeitszeit:	60 Stunden

Lernergebnisse:

Die Studierenden erlernen an Beispielen die formale Spezifikation von Sprachen. Sie haben das notwendige Wissen zur Einordnung von Sprachen in die bekannten Sprachklassen und kennen die ihnen jeweils zugeordneten Automatenmodelle. Sie erlernen die Dualität zwischen Akzeption und Generation. Sie kennen Verfahren zur Zustandsreduktion von Automaten und wenden diese in Übungen an. Sie lernen den grundsätzlichen Aufbau mehrschichtiger Grammatiken an Beispielen. Sie kennen die Grundlagen des Übersetzerbaus für höhere Programmiersprachen.

Bildung der Modulnote:

Klausur

Modulbeschreibung Security Engineering

Schlüsselworte: Security, Engineering, Software, Design, Standards

Zielgruppe:	1. Semester AIM	Modulnummer:	AI 102
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Englisch		
Modulverantwortung:	Prof. Dr. Dominik Schoop		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen des Software Engineerings

Modulziel – angestrebte Lernergebnisse:

Die Studierenden werden in die Lage versetzt, nach der Einschätzung des Sicherheitsrisikos, Technologien und Vorgehensweisen anzuwenden, um ein sicheres, komplexes und heterogenes IT-System zu entwickeln.

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen:

- Moderne Verschlüsselungsverfahren
- Kryptographische Protokolle
- Security System Modelling
- Security Requirements Engineering
- Security Design Principles
- Verifikation von Sicherheitskomponenten

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Sicherheitssysteme zu analysieren,
- Sicherheitsrisiken zu analysieren,
- Systeme auf Sicherheit zu validieren und zu verifizieren
- Informationstechnisch sichere Systeme zu entwerfen

Übergreifende Kompetenzen

Die Studierenden können

- nach der Einschätzung des Sicherheitsrisikos, Vorgehensweisen und Technologien anwenden, um ein sicheres, komplexes und heterogenes IT-System zu entwickeln.

Inhalt:

- Diskrete Mathematik
- Moderne Verschlüsselungsmethoden
- Elliptische Kurven
- Faktorisierungsverfahren
- Kryptographische Protokolle
- Security System Modelling
- Security Requirements Engineering
- Security risk assessment
- Security design principles and patterns
- Verifikation von Sicherheitsprotokollen
- Verifikation von Sicherheitskomponenten
- Secure Coding

Zu:

Inhaltsverzeichnis - Autonome Systeme - Business Intelligence - Mobile Computing - Modulplan

Literaturhinweise:

Ross Anderson: Security Engineering: A Guide to Building Dependable Distributed Systems, Wiley, 2008
Jonathan Katz and Yehuda Lindell: Introduction to Modern Cryptography, Chapman and Hall/CRC, 2014
Bruce Schneier: Angewandte Kryptographie, Pearson Studium, 2006

Wird angeboten:

in jedem Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform: Vorlesung mit Übungen und Security Case Study
Leistungskontrolle: Klausur (90 Minuten), Bericht
Anteil Semesterwochenstunden: 4 SWS
Geschätzte studentische Arbeitszeit: 150 Stunden
Lernergebnisse:

Die Studierenden beherrschen die Methoden zur Einschätzung von Risiken und sind in der Lage, der Situation angemessene Methoden für den sicheren Entwurf, Implementierung und Betrieb eines IT-Systems auszuwählen und anzuwenden. Die Studierenden werden befähigt, die IT-Sicherheit von Anwendungen zu analysieren und zu bewerten. Sie sind in der Lage angemessene IT-Sicherheitslösungen systematisch zu entwickeln und darzustellen.

Bildung der Modulnote:

Klausur und Bericht bis zu 25%

Modulbeschreibung Wahlpflichtmodul 1 und 2

Schlüsselworte: Vertiefung im eigenen Studienprofil

Zielgruppe:	1. Semester AIM	Modulnummer:	AI 103 AI 104
Arbeitsaufwand:	5 ECTS		
Davon	Kontaktzeit		Abhängig vom gewählten Modul
	Selbststudium		Abhängig vom gewählten Modul
	Prüfungsvorbereitung		Abhängig vom gewählten Modul
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr. Andreas Rößler		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Abhängig vom gewählten Modul

Gesamtziel:

Die Studierenden erlangen eine wissenschaftliche und fachliche Vertiefung auf dem Gebiet der Vertiefung.

Inhalt:

Abhängig vom gewählten Modul

Literaturhinweise:

Abhängig vom gewählten Modul

Wird angeboten:

Abhängig vom gewählten Modul

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Abhängig vom gewählten Modul
Leistungskontrolle:	Abhängig vom gewählten Modul
Anteil Semesterwochenstunden:	Abhängig vom gewählten Modul
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden verfügen über eine wissenschaftliche und fachliche Vertiefung im eigenen Studienprofil.

Bildung der Modulnote:

Abhängig vom gewählten Modul

Modulbeschreibung Forschungsprojekt 1

Schlüsselworte: Wissenschaftliches Arbeiten im Team

Zielgruppe:	1. Semester AIM	Modulnummer:	AI 105
Arbeitsaufwand:	10 ECTS		300 h
Davon	Kontaktzeit		30 h
	Selbststudium		270 h
	Prüfungsvorbereitung		0 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr. Andreas Rößler		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Anwendung der Methoden der Softwareentwicklung, Kenntnisse in der gewählten Vertiefungsrichtung, Grundkenntnisse wissenschaftlichen Arbeitens

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Qualitätskriterien für wissenschaftliches Arbeiten
- die Methoden des Projektmanagement

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- wissenschaftliche Projekte im Team zu planen und durchzuführen,
- die in den Kern- und Vertiefungsfächern erworbenen Kenntnisse zur Lösung von Problemen aus dem Bereich der Forschung einzusetzen,
- Lösungsansätze (Stand der Technik) zu recherchieren und zu verstehen,
- gefundene Lösungsansätze bewerten,
- die Ergebnisse ihres wissenschaftlichen Arbeitens nachvollziehbar dokumentieren.

Übergreifende Kompetenzen

Die Studierenden können

- unter Anleitung komplexe Problemstellungen aus der Forschung oder aus der Industrie innerhalb einer vorgegebenen Frist zu lösen,
- neue Kenntnisse in der Informatik zu gewinnen und neue Verfahren zu entwickeln,
- Wissen aus verschiedenen Domänen integrieren,
- in einem Team gemeinsam eine Aufgabe erfolgreich umzusetzen.

Inhalt:

Im Forschungsprojekt bearbeiten Studierende in einem Team unter Anleitung eines Dozenten aktuelle Forschungsthemen aus wissenschaftlichen Einrichtungen oder forschungsnahe Themen aus der Industrie. Die Projekte sind auf ein Jahr angelegt, wobei alle Phasen eines Softwareprojekts durchlaufen werden sollen: Problem- und Anforderungsanalyse, Recherche des Standes der Technik, Projektplanung, Erarbeitung von Lösungsansätzen, Softwareentwurf, Implementierung, Testphase. Die Studierenden erarbeiten Arbeits- und Zeitpläne und berichten regelmäßig über ihren Fortschritt. Am Ende der Semester tragen die Studierenden jeweils Zwischen- bzw. Endergebnisse vor.

Literaturhinweise:

Abhängig von der gewählten Problemstellung

Wird angeboten:

Sommer- und Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Projektarbeit mit Bericht und Referat
Leistungskontrolle:	Schriftlicher Bericht und Referat (20 Minuten)
Anteil Semesterwochenstunden:	2 SWS
Geschätzte studentische Arbeitszeit:	300 Stunden

Bildung der Modulnote:

Projektarbeit: Referat und Dokumentation

Modulbeschreibung Advanced Software Engineering

Schlüsselworte: Software Engineering, Metriken, Empirie

Zielgruppe:	2. Semester AIM	Modulnummer:	AI 201
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Englisch oder Deutsch		
Modulverantwortung:	Dipl.-Inf. Holger Cermann		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen des Softwareengineering

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- Agile Vorgehensweisen in Großprojekten: Scrum/Agile@Scale, Agile Requirements Engineering
- Architekturmodelle für große Systeme: (a)synchrone Kommunikation, Batches, Transaktionen
- Aktuelle Methoden der Software-Qualität, Software Measurements und Metriken, Testing
- Verfahren der Software-Archäologie
- Securityaspekte bei der Systementwicklung

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Methoden des Projekt- & Risikomanagement in realen Softwareprojekten anzuwenden
- Unternehmensarchitekturen zu entwickeln: SOA, Microservices, Governance, Software in der Cloud, DevOps
- an der Ausgestaltung und Umsetzung von Architektur und SE-Prozessen von realen Projekten mitzuwirken

Übergreifende Kompetenzen

Die Studierenden können

- Probleme des Softwareengineerings in realen modernen Projekten lösen
- Arbeiten in global verteilten Teams durchführen

Inhalt:

Fokus auf wichtige aktuelle Aspekte mit Praxisbezug zu Architektur, Qualität, Vorgehen und Digitale Transformation im Software Engineering

Literaturhinweise:

Somerville: Software Engineering. Pearson 2012.
Ludewig, Lichter: Software Engineering. dpunkt.verlag 2013.

Wird angeboten:

in jedem Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung mit Übungen
Leistungskontrolle:	Mündliche Prüfung (20 Minuten)
Anteil Semesterwochenstunden:	4 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden
Lernergebnisse:	

Die Studierenden sind in der Lage das theoretisch erlangte Wissen aus der Vorlesung in Übungen inkl. Hackathon anzuwenden und auch vor der Gruppe vorzustellen.

Bildung der Modulnote:

Klausur

Modulbeschreibung IT Management

Schlüsselworte: IT Management

Zielgruppe:	2. Semester AIM	Modulnummer:	AI 202
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch		
Modulverantwortung:	Prof. Dr. Thomas Rodach		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundkenntnisse in Betriebswirtschaft sowie von Informationssystemen in Unternehmen

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Bedeutung der IT-Strategie
- Methoden, die IT-Strategie aus der Geschäftsstrategie zu entwickeln und deren Wertbeitrag einzuschätzen
- unterschiedliche Organisationsformen einer IT-Abteilung
- Aufgaben und Methoden des IT-Controlling
- die Bedeutung von IT-Governance und – Compliance

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- die Auswirkungen unterschiedlicher Organisationsformen einer IT-Abteilung einzuschätzen
- IT-Controlling Konzepte zu entwickeln
- Methoden und Vorgehensweisen zur Entwicklung und Management einer IT Unternehmensarchitektur einzusetzen

Übergreifende Kompetenzen

Die Studierenden können

- eine IT-Unternehmensarchitektur entwickeln und das sich daraus ergebende Life Cycle Management der Anwendungssysteme konzipieren
- IT-Governance und – Compliance Konzepte entwickeln
- Managementaufgaben in der IT, die im Rahmen von Führungsfunktionen wahrzunehmen sind, übernehmen

Inhalt:

- IT-Strategie und IT Business Alignment
- Aufbau und Organisation einer IT-Abteilung
- IT-Controlling
- IT-Unternehmensarchitektur und IT-Life Cycle Management
- IT-Governance und -Compliance

Literaturhinweise:

Gadatsch, Mayer: Masterkurs IT-Controlling, 5. Auflage, 2014
Hofmann, Schmidt: Masterkurs IT-Management, 2. Auflage, 2010
Krcmar: Informationsmanagement, 5. Auflage, 2010

Wird angeboten:

in jedem Sommersemester

Zu:

Inhaltsverzeichnis - Autonome Systeme - Business Intelligence - Mobile Computing - Modulplan

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung mit Übungen
Leistungskontrolle:	Mündliche Prüfung (20 Minuten)
Anteil Semesterwochenstunden:	4 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden

- verstehen die Bedeutung der IT-Strategie und kennen Methoden, die IT-Strategie aus der Geschäftsstrategie zu entwickeln und deren Wertbeitrag einzuschätzen
- sind mit unterschiedlichen Organisationsformen einer IT-Abteilung vertraut und können deren Auswirkungen einschätzen
- kennen Aufgaben und Methoden des IT-Controlling und können ein IT-Controlling Konzept entwickeln
- können eine IT-Unternehmensarchitektur entwickeln und das sich daraus ergebende Life Cycle Management der Anwendungssysteme konzipieren
- verstehen die Bedeutung von IT-Governance und – Compliance und können entsprechende Konzepte entwickeln
- können eine komplexe Problemstellung systematisch analysieren
- sind in der Lage, die Methoden entsprechend dem aktuellen Stand der Wissenschaft darauf anzuwenden
- verstehen es, Recherchemethoden nach neuen Forschungsergebnissen einzusetzen
- sind in der Lage, die Anwendbarkeit der Methoden in der Praxis kritisch einzuschätzen, Methoden weiterzuentwickeln oder eigene Methoden zu entwickeln
- können daraus eine eigenständige Problemlösung entwickeln

Bildung der Modulnote:

Mündliche Prüfung (20 Minuten)

Modulbeschreibung Wahlpflichtmodul 3 und 4

Schlüsselworte: Vertiefung im eigenen Studienprofil

Zielgruppe:	1. Semester AIM	Modulnummer:	AI 203 AI 204
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit	Abhängig vom gewählten Modul	
	Selbststudium	Abhängig vom gewählten Modul	
	Prüfungsvorbereitung	Abhängig vom gewählten Modul	
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr. Andreas Rößler		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Abhängig vom gewählten Modul

Gesamtziel:

Die Studierenden erlangen eine wissenschaftliche und fachliche Vertiefung auf dem Gebiet der Vertiefung.

Inhalt:

Abhängig vom gewählten Modul

Literaturhinweise:

Abhängig vom gewählten Modul

Wird angeboten:

Abhängig vom gewählten Modul

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform: Abhängig vom gewählten Modul

Leistungskontrolle: Abhängig vom gewählten Modul

Anteil Semesterwochenstunden: Abhängig vom gewählten Modul

Geschätzte studentische Arbeitszeit: 150 Stunden

Lernergebnisse:

Die Studierenden verfügen über eine wissenschaftliche und fachliche Vertiefung im eigenen Studienprofil.

Bildung der Modulnote:

Abhängig vom gewählten Modul

Modulbeschreibung Forschungsprojekt 2

Schlüsselworte: Wissenschaftliches Arbeiten im Team

Zielgruppe:	2. Semester AIM	Modulnummer:	AI 205
Arbeitsaufwand:	10 ECTS		300 h
Davon	Kontaktzeit		30 h
	Selbststudium		270 h
	Prüfungsvorbereitung		0 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr. Andreas Rößler		
Stand:	Jan. 2017		

Voraussetzungen nach Prüfungsordnung:

Keine

Empfohlene Voraussetzungen:

Anwendung der Methoden der Softwareentwicklung, Kenntnisse in der gewählten Vertiefungsrichtung, Grundkenntnisse wissenschaftlichen Arbeitens

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Qualitätskriterien für wissenschaftliches Arbeiten
- die Methoden des Projektmanagement

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- wissenschaftliche Projekte im Team zu planen und durchzuführen,
- die in den Kern- und Vertiefungsfächern erworbenen Kenntnisse zur Lösung von Problemen aus dem Bereich der Forschung einzusetzen,
- Lösungsansätze (Stand der Technik) zu recherchieren und zu verstehen,
- gefundene Lösungsansätze bewerten,
- die Ergebnisse ihres wissenschaftlichen Arbeitens nachvollziehbar dokumentieren.

Übergreifende Kompetenzen

Die Studierenden können

- unter Anleitung komplexe Problemstellungen aus der Forschung oder aus der Industrie innerhalb einer vorgegebenen Frist zu lösen,
- neue Kenntnisse in der Informatik zu gewinnen und neue Verfahren zu entwickeln,
- Wissen aus verschiedenen Domänen integrieren,
- in einem Team gemeinsam eine Aufgabe erfolgreich umzusetzen.

Inhalt:

Im Forschungsprojekt bearbeiten Studierende in einem Team unter Anleitung eines Dozenten aktuelle Forschungsthemen aus wissenschaftlichen Einrichtungen oder forschungsnahe Themen aus der Industrie. Die Projekte sind auf ein Jahr angelegt, wobei alle Phasen eines Softwareprojekts durchlaufen werden sollen: Problem- und Anforderungsanalyse, Recherche des Standes der Technik, Projektplanung, Erarbeitung von Lösungsansätzen, Softwareentwurf, Implementierung, Testphase. Die Studierenden erarbeiten Arbeits- und Zeitpläne und berichten regelmäßig über ihren Fortschritt. Am Ende der Semester tragen die Studierenden jeweils Zwischen- bzw. Endergebnisse vor.

Literaturhinweise:

Abhängig von der gewählten Problemstellung

Wird angeboten:

Sommer- und Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Projektarbeit mit Bericht und Referat
Leistungskontrolle:	Schriftlicher Bericht und Referat (20 Minuten)
Anteil Semesterwochenstunden:	2 SWS
Geschätzte studentische Arbeitszeit:	300 Stunden

Bildung der Modulnote:

Projekt: Referat und Dokumentation

Modulbeschreibung Masterarbeit

Schlüsselworte: Selbstständiges wissenschaftliches Arbeiten

Zielgruppe: 3. Semester AIM

Modulnummer: AI 301

Arbeitsaufwand:	25 ECTS	750 h
Davon	Kontaktzeit	30 h
	Selbststudium	660 h
	Prüfungsvorbereitung	60 h
Unterrichtssprache:	Deutsch oder Englisch	
Modulverantwortung:	Prof. Dr. Andreas Rößler	

Stand: Jan. 2017

Empfohlene Voraussetzungen:

Kenntnisse der Methoden wissenschaftlichen Arbeitens, sichere Anwendung der Methoden des Softwareengineering, umfassende Kenntnisse in der gewählten Vertiefungsrichtung

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Qualitätskriterien für wissenschaftliches Arbeiten
- die Methoden des Projektmanagement

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- wissenschaftliche Fragestellungen zu formulieren,
- wissenschaftliche Methoden anzuwenden,
- wissenschaftliche Projekte zu planen und durchzuführen,
- die in den Kern- und Vertiefungsfächern erworbenen Kenntnisse zur Lösung von Problemen einzusetzen,
- Lösungsansätze (Stand der Forschung) zu recherchieren und zu verstehen, und zu bewerten,
- eigene Lösungsansätze zu entwickeln und umzusetzen,
- die Ergebnisse ihres wissenschaftlichen Arbeitens nachvollziehbar dokumentieren.

Übergreifende Kompetenzen

Die Studierenden können

- eine komplexe Problemstellung der Informatik selbstständig, wissenschaftlich, innerhalb einer vorgegebenen Frist zu bearbeiten,
- den dazugehörigen Stand der Forschung zu recherchieren, zu strukturieren und zu verstehen,
- geeignete Methoden und Verfahren auszuwählen, diese korrekt einzusetzen und falls notwendig sie anzupassen oder weiter zu entwickeln,
- ihre Ergebnisse mit anderen Ergebnissen zu vergleichen und ihre Lösungsansätze kritische zu überprüfen,
- ihre Ergebnisse strukturiert zu dokumentieren und in wissenschaftlicher Form zu veröffentlichen.

Inhalt:

- Problemanalyse und Eingrenzung des Themas
- Literaturrecherche
- Planung der Vorgehensweise, Erarbeitung eines Lösungsansatzes
- Zeit- und Projektmanagement
- Herstellen eines Bezugs zwischen eigenen Ansätze und dem Stand der Forschung
- Wissenschaftliche Darstellung der Ergebnisse
- Verteidigung

Literaturhinweise:

Abhängig von der gewählten Problemstellung

Wird angeboten:

in jedem Semester

Teilgebiete und Leistungsnachweise:

Modulteil:	Masterarbeit
Lehr- und Lernform:	Wissenschaftliche Arbeit mit Bericht und Verteidigung
Leistungskontrolle:	Schriftlicher Bericht und mündliche Prüfung (30 Minuten)
Anteil Semesterwochenstunden:	2 SWS
Geschätzte studentische Arbeitszeit:	750 Stunden
Lernergebnisse:	

Die Studierenden verfügen über **Kenntnisse** der grundlegenden Techniken des wissenschaftlichen Arbeitens, insbesondere Gliederung, Zitieren und formale Aspekte. Sie besitzen die **Fertigkeiten** Probleme zu analysieren, zu strukturieren und zu lösen. Dazu sind sie in der Lage, den dazugehörigen Stand der Forschung zu recherchieren, zu strukturieren und zu verstehen sowie geeignete Methoden und Verfahren auszuwählen, diese korrekt einzusetzen und falls notwendig sie anzupassen oder weiter zu entwickeln. Sie können ihre Ergebnisse mit anderen Ergebnissen vergleichen und ihre Lösungsansätze kritisch überprüfen. Die Studierenden verfügen über die **Kompetenzen** zur selbstständigen Projekt- und Zeitplanung und haben Verantwortungsbewusstsein in Bezug auf wissenschaftliche Qualitätskriterien wie Überprüfbarkeit und Verlässlichkeit von wissenschaftlichen Ergebnissen erworben.

Bildung der Modulnote:

Gemittelte Note aus Bericht (Faktor 22) und mündlicher Prüfung (Faktor 3)

Modulbeschreibung Publikation

Schlüsselworte: Selbstständiges wissenschaftliches Schreiben

Zielgruppe:	3. Semester AIM	Modulnummer:	AI 302
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		15 h
	Selbststudium		135 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr. Andreas Rößler		
Stand:	3. März 2016		

Empfohlene Voraussetzungen:

Kenntnisse der Methoden wissenschaftlichen Arbeitens, erfolgreiche Teilnahme am Forschungsprojekt (AI 105 und AI 205)

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die formalen Aspekte einer wissenschaftlichen Veröffentlichung
- geeignete Journalen und Konferenzen

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- eine Thema für eine Veröffentlichung einzugrenzen
- den Stand der Forschung zu recherchieren, zu strukturieren, zu verstehen und wiederzugeben
- Bezüge zwischen eigenen Ansätzen und dem Stand der Forschung herzustellen

Übergreifende Kompetenzen

Die Studierenden können

- Forschungsergebnisse strukturiert dokumentieren und in eine publikationsreife Form bringen.

Literaturhinweise:

Balzert, Helmut; Schröder, Marion; Schaefer, Christian (2013): Wissenschaftliches Arbeiten. Ethik Inhalt & Form wiss. Arbeiten Handwerkszeug Quellen Projektmanagement Präsentation. 2. Aufl., 1. korr. Nachdr. Herdecke, Witten: W3L-Verl. (Soft skills).

Kornmeier, Martin (2013): Wissenschaftlich schreiben leicht gemacht. Für Bachelor Master und Dissertation. 6., aktualisierte Aufl. Bern, [Stuttgart]: Haupt (UTB, 3154 : Schlüsselkompetenzen).

Theisen, Manuel René (2013): Wissenschaftliches Arbeiten. Erfolgreich bei Bachelor- und Masterarbeit. 16., vollst. überarb. Aufl. München: Vahlen.

Wird angeboten:

in jedem Semester

Lehr- und Lernform:	Verfassen einer wissenschaftlichen Publikation
Leistungskontrolle:	Veröffentlichungsreifer Text
Anteil Semesterwochenstunden:	1 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden verfügen über Kenntnisse darüber, wie und wo wissenschaftliche Ergebnisse publiziert werden können. Sie sind in der Lage, die Ergebnisse ihrer wissenschaftlichen Tätigkeit nachvollziehbar nach wissenschaftlichen Kriterien zusammenzufassen. Sie verfügen über die Kompetenz, selbstständig für die Qualitätssicherung der Publikation zu sorgen und diese fristgerecht einzureichen.

Bildung der Modulnote:

Bewertung einer veröffentlichungsreifen Publikation

Modulbeschreibung Intelligent Data Analytics

Schlüsselworte: Big Data, Data Mining, Zeitreihen, Klassifikation, Vorhersage, Querying

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W01
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Englisch		
Modulverantwortung:	Prof. Jürgen Nonnast		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Gute mathematische Kenntnisse v.a. in Statistik und Optimierung
Kenntnisse in Informatik

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- Grundlagen der Zeitreihen
- Anwendungen, in denen Zeitreihen generiert und aufgezeichnet werden
- Verfahren der Klassifikation von Zeitreihendaten
- Verfahren zur Regressionsanalyse und zur Vorhersage
- Grundlagen der künstlichen Neuronalen Netze

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- geeignete Analyseverfahren auszuwählen und anzuwenden

Übergreifende Kompetenzen

Die Studierenden können

- Zeitreihen mit Hilfe von Algorithmen aus den Bereichen „Data Mining“ und „maschinelles Lernen“ intelligent analysieren.

Inhalt:

- Introduction to Data Mining with a focus on Time Series Data (Temporal Data Mining)
- Fundamentals of Time Series Data
- Classification, Time Series Querying, Regression/Forecasting
- Visualization of Time Series
- Artificial Neural Networks
- Applied Data Mining for Hybrid Vehicle Powertrain

Literaturhinweise:

T. Mitsa: Temporal Data Mining. Chapman & Hall/CRC Data Mining and Knowledge Discovery. 2010
J. Han, M. Kamber, J. Pei: Data Mining – Concepts and Techniques (3rd Edition). Morgan Kaufman, 2012
R. J. Hyndman, G. Athanasopoulos: Forecasting: principles and practice. Available online at <https://www.otexts.org/fpp>, 2014

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung mit praktischen Übungen
Leistungskontrolle:	Bericht und Referat (20 Minuten)
Anteil Semesterwochenstunden:	4 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden besitzen Grundkenntnisse in „Data Mining auf Zeitreihen“ und im Umgang mit der Software „R“. Sie sind in der Lage, ausgewählte Verfahren aus den Funktionalitäten „Querying“, „Klassifikation“ und „Vorhersage“ auf Zeitreihen anzuwenden. Diese Techniken werden in vielen industriellen Anwendungen, z.B. bei einer Modell-basierten Diagnose der Hochvoltbatterie eines Hybridfahrzeuges, eingesetzt. Die gelernten Methoden und Konzepte können zum Zwecke des „Data Mining“ auch auf andere Datentypen angewandt werden.

Bildung der Modulnote:

Bericht und Referat

Modulbeschreibung Business Intelligence

Schlüsselworte: Business Intelligence, Datawarehouse, OLAP, Business Performance

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W02
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		30 h
	Selbststudium		90 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch		
Modulverantwortung:	Prof. Dr. Dirk Hesse		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen Datenbanken

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundbegriffe der Business Intelligence
- die Bedeutung von BI für die unternehmerische Praxis
- verschiedene Konzepte und Methoden der BI

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- verschiedene Instrumente und Anwendungen der Business Intelligence zielgerichtet einzusetzen

Übergreifende Kompetenzen

Die Studierenden können

- für spezifische Anwendungsfelder geeignete BI-Verfahren auswählen und umsetzen.

Inhalt:

Vertiefung der Methodenkompetenz im Bereich Business Analytics:

- Verarbeitung (semi-) strukturierter Daten im ETL Prozess
- Logische Modellierung (Star Schema, Snowflake Schema etc.)
- Einrichtung von multidimensionalen Modellen (OLAP Cubes)
- Reporting und Analyse mittels verschiedener Tools, Queries und Webreports
- Performanceverbesserungen und Berechtigungskonzepte

Literaturhinweise:

Business Intelligence and Analytics: Systems for Decision Support, Sharda, Turba, Delen, Pearson Education Limited, 10. Auflage 2014.

Kemper, Hans-Georg, et al.: Business Intelligence - Grundlagen und praktische Anwendungen. Eine Einführung in die IT-basierte Managementunterstützung, Vieweg und Teubner, 3. Auflage 2010.

Wird angeboten:

Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung mit Seminar
Leistungskontrolle:	Klausur (90 Minuten)
Anteil Semesterwochenstunden:	1,5 SWS
Geschätzte studentische Arbeitszeit:	90 Stunden

Lernergebnisse:

Die Studierenden verfügen über Kenntnisse der grundlegenden Konzepte des Business Intelligence. Sie haben die Fertigkeit verschiedene Ansätze, Methoden und Werkzeuge des Business Intelligence zu unterscheiden und können Unternehmens- /Wettbewerbs- und Kundendaten analysieren. Sie verfügen über die Kompetenz, die vorgestellten Konzepte in das unternehmensweite Informations- und Wissensmanagement zu integrieren.

Lehr- und Lernform:	Projektarbeit (Labor)
Leistungskontrolle:	Referat, Testat
Anteil Semesterwochenstunden:	0,5 SWS
Geschätzte studentische Arbeitszeit:	60 Stunden

Lernergebnisse:

Die Studierenden haben die Fertigkeit BI-Systeme zu modellieren und zu implementieren. Insbesondere beherrschen sie verschiedene BI-Konzeptionen und können diese im Rahmen betriebswirtschaftlicher Fragestellungen anwenden und in BI-Planungssysteme eigenständig umsetzen.

Bildung der Modulnote:

Klausur

Modulbeschreibung Mobile Communication

Schlüsselworte: Automotive, Communication, Safety, Security

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W03
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Englisch		
Modulverantwortung:	Prof. Dr.-Ing. Harald Melcher, Prof. Dr. Dominik Schoop		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen Kommunikationstechnik
Module IT-Security Engineering and Advanced Software Engineering

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen der Intelligent Transportation Systems (ITS) und der Vehiculäre Ad-Hoc-Netzwerke (VANETs)
- C2I/V2I-Anwendungen
- Automotive wireless Netztechnologie (WLAN (IEEE 802.11p), CAM, DENM)
- relevante Standards (IEEE, ISO)
- Systeme zur Positionsbestimmung (GPS, ...)
- den Zusammenhang Safety und Security

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- die Ziele von ITS und VANETs zu erklären
- die Sicherheit von VANETs einzuschätzen und Sicherheitsmaßnahmen vorzuschlagen
- die Architektur und Technologie von VANETs zu erklären

Übergreifende Kompetenzen

Die Studierenden können

- einfache Car2Infrastructure- und Car2-Car-Applikationen zu implementieren.

Inhalt:

Vertiefung der Methodenkompetenz im Bereich Business Analytics:

- Verarbeitung (semi-) strukturierter Daten im ETL Prozess
- Logische Modellierung (Star Schema, Snowflake Schema etc.)
- Einrichtung von multidimensionalen Modellen (OLAP Cubes)
- Reporting und Analyse mittels verschiedener Tools, Queries und Webreports
- Performanceverbesserungen und Berechtigungskonzepte

Literaturhinweise:

Christoph Sommer, Falko Dressler: Vehicular Networking. Cambridge University Press, 2014
Erdal Cayirci, Chunming Rong: Security in Wireless Ad Hoc and Sensor Networks, John Wiley & Sons, 2009
Srikanta Patnaik, Xiaolong Li, Yeon-Mo Yang: Recent Development in Wireless Sensor and Ad-hoc Networks, Springer, 2014

Wird angeboten:

Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung mit Seminar und Projektarbeit
Leistungskontrolle:	Benotetes Referat
Anteil Semesterwochenstunden:	3 SWS
Geschätzte studentische Arbeitszeit:	60 Stunden

Lehr- und Lernform:	Projektarbeit (Labor)
Leistungskontrolle:	Unbenotetes Referat, Testat
Anteil Semesterwochenstunden:	1 SWS
Geschätzte studentische Arbeitszeit:	90 Stunden

Bildung der Modulnote:

Projektarbeit mit Referat (20 Minuten)

Modulbeschreibung Datenfusion

Schlüsselworte: Datafusion, Kalman-Filter, Bayes Netze, Markov Modelle

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W04
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		45 h
	Selbststudium		75 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch		
Modulverantwortung:	Prof. Dr.-Ing. Reiner Marchthaler		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen Statistik und Automatentheorie (im Speziellen in Zustandsautomaten)

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen der Kalman-Filter-Gleichungen
- die hier zugrunde liegenden Annahmen
- die Grundlagen von Markov-Modellen, Partikelfilter und Bayes-Netzen

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage Zustandsmodelle zu modellieren

Übergreifende Kompetenzen

Die Studierenden können in praktischen Übungen Kalman-Filter in einem Mikrocontroller implementieren

Inhalt:

- Kalman-Filter und Partikelfilter
- Markov Decision Processes und Hidden Markov Modelle
- Bayes Netzwerke

Literaturhinweise:

Richard E. Neapolitan: "Learning Bayesian Networks". Prentice Hall, 2003

Olle Häggström: "Finite Markov Chains and Algorithmic Applications". Cambridge University Press, 2002

Brown, R. G.; Hwang, P. Y. C.: "Introduction to Random Signals and Applied Kalman Filtering". Third Edition. John Wiley & Sons, Inc., 2012

Simon, Dan "Optimal State Estimation", John Wiley & Sons, Inc., 2006

Thrun, S.; u.a. "Probabilistic Robotics", MIT Press, 2005

Marchthaler, R.; Dingler S. "Kalman-Filter - Einführung in die Zustandsschätzung und ihre Anwendung", Springer-Vieweg Verlag, 2016

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung mit Übungen
Leistungskontrolle:	Klausur (90 Minuten)
Anteil Semesterwochenstunden:	3 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden beherrschen die mathematischen Konzepte zum Entwurf eines Kalman-Filters, einer Markov-Kette und eines Bayes-Netzwerkes. Darüber hinaus sind Sie in der Lage selbstständig ein Kalman-Filter für zur Sensordatenfusion zu entwerfen.

Bildung der Modulnote:

Klausur

Modulbeschreibung Automotive System Design

Schlüsselworte: System- und Softwarearchitekturen sowie Netzwerke und Kommunikationsprotokolle in der automobilen Anwendung, Funktionale Sicherheit

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W05
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Englisch oder Deutsch		
Modulverantwortung:	Prof. Dr.-Ing. Werner Zimmermann		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Kenntnisse der Informatik und Netzwerke auf Bachelorniveau, Kompetenzen zur Softwareentwicklung in einer gängigen Programmiersprache bevorzugt, C/C++

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen der Anwendungsdomänen (Powertrain, Chassis, Body, Infotainment, Driver Assistance)
- Grundlagen verteilter Systeme, Kommunikationsanforderungen, Analyse und Gestaltung unter Echtzeit- und Sicherheitsanforderungen
- die Architektur von Systemen im Automobil und deren Software

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Analysen der Zuverlässigkeit und der Sicherheit durchzuführen
- funktionale Sicherheitsanforderungen aufzustellen zu implementieren
- die Netzwerkkommunikation von Systemen im Automobil zu gestalten und quantitativ abzuschätzen

Übergreifende Kompetenzen

Die Studierenden können

- Systeme im Automobil hinsichtlich Sicherheit und Zuverlässigkeit abschätzen
- sichere und zuverlässige Systeme im Automobil konzipieren und implementieren

Literaturhinweise:

J. Schäufele, T. Zurawka: Automotive Software Engineering, Springer-Vieweg
C. Marscholik, P. Supke: Road Vehicles – Diagnostic Communication. VDE-Verlag
W. Zimmermann, R. Schmidgall: Bussysteme in der Fahrzeugtechnik – Protokolle, Standards und Softwarearchitektur. Springer-Vieweg

Wird angeboten:

Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung
Leistungskontrolle:	Klausur (60 Minuten)
Anteil Semesterwochenstunden:	3 SWS
Geschätzte studentische Arbeitszeit:	120 Stunden
Lernergebnisse:	

Die Studierenden kennen

- die Grundlagen der Anwendungsdomänen (Powertrain, Chassis, Body, Infotainment, Driver Assistance)
- Grundlagen verteilter Systeme, Kommunikationsanforderungen, Analyse und Gestaltung unter Echtzeit- und Sicherheitsanforderungen
- die Architektur von Systemen im Automobil und deren Software

Lehr- und Lernform:	Gruppenprojekt und Referat
Leistungskontrolle:	Referat (20 Minuten)
Anteil Semesterwochenstunden:	1 SWS
Geschätzte studentische Arbeitszeit:	30 Stunden
Lernergebnisse:	

Die Studierenden sind in der Lage

- Analysen der Zuverlässigkeit und der Sicherheit durchzuführen
- funktionale Sicherheitsanforderungen aufzustellen zu implementieren
- die Netzwerkkommunikation von Systemen im Automobil zu gestalten und quantitativ abzuschätzen

Die Studierenden können

- Systeme im Automobil hinsichtlich Sicherheit und Zuverlässigkeit abschätzen
- sichere und zuverlässige Systeme im Automobil konzipieren und implementieren

Bildung der Modulnote:

Klausur

Modulbeschreibung Advanced Control

Schlüsselworte: Fuzzy-Regelung; moderne Regelungstechnik (PI-Zustandsregler, Zustands- und Störgrößenbeobachter), Optimale Regler und Zustandsschätzer

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W06
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch		
Modulverantwortung:	Prof. Dr. Walter Lindermeir		
Stand:	20.12.2016		

Empfohlene Voraussetzungen:

Kenntnisse der Booleschen Schaltalgebra, der Wahrscheinlichkeitsrechnung sowie der Matrix-Vektor-Rechnung, Integral- und Differentialrechnung.
Kenntnisse der Regelungstechnik (PID-Regler, stationäres Verhalten von Regelkreisen, Blockschaltbildarithmetik, Zustandsregler und Luenberger Beobachter)
Beschreibung dynamischer Systeme im Frequenz (Laplace-Übertragungsfunktion) und Zeitbereich (Zustandsbeschreibung)
Kenntnisse einer Simulationssprache (z.B.. MATLAB/Simulink)

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen der Fuzzy-Logik (Einsatzgebiete, Fuzzifizierung, Regelbasis, Defuzzifizierung)
- Einsatz und Funktionsweise von Fuzzy-Reglern
- Entwurf und Einsatz von PI-Zustandsreglern und Störgrößenbeobachtern (Polvorgabe)
- Optimale Regler (LQR-Regelung) und optimale Zustandsschätzung dynamischer Systeme (Kalman-Filter)
- Regler in der Zwei-Freiheitsgrad-Struktur (2-DOF)
- Steuerungen bzw. Vorsteuerungen basierend auf der Flachheitsmethode
- IMC-Regler

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Fuzzy-Inferenz-Systeme zu entwerfen und einzusetzen, z.B. unter Einsatz der Matlab Fuzzy-Logic-Toolbox
- PI-Zustandsregler zu entwickeln (einfache Systeme von Hand, bzw. unter Einsatz von Matlab/Simulink) und in Betrieb zu nehmen
- Störgrößenmodelle zu definieren und Störgrößenbeobachter zu entwickeln
- Kalman-Filter zu entwerfen und sinnvoll zu parametrieren
- 2-DOF-Reglerstrukturen zu verstehen und z.B. mit Hilfe der Flachheitsmethodik auslegen
- IMC-Regler zu entwerfen

Übergreifende Kompetenzen

Die Studierenden können

- die in der Systemtheorie gebräuchlichen Systembeschreibungen (Laplace-Übertragungsfunktionen und Zustandsbeschreibungen) vorteilhaft einsetzen
- Regelungen (PID-/Zustands-/IMC-/Fuzzy-Regler) eigenständig auslegen
- Zustandsschätzer (Luenberger, Kalman, Störgrößen) eigenständig auslegen

Zu:

- Erweiterte Regler-Strukturen (2-DOF-Struktur, IMC) sinnvoll einsetzen
- Sich im Umfeld von eingebetteten Systemen z.B. in den Bereichen der Automobiltechnik oder der Industrieautomatisierung fachlich einbringen

Literaturhinweise:

Föllinger, O.: Regelungstechnik. Hüthig Verlag
Lunze, J.: Regelungstechnik. 2 Bände, Springer Verlag
Unbehauen, H.: Regelungstechnik. 3 Bände, Vieweg Verlag
Simon, D.: Optimal State Estimation
Mann, Schiffelgen, Forriep: Einführung in die Regelungstechnik

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform: Vorlesung
Leistungskontrolle: Klausur (90 Minuten)
Anteil Semesterwochenstunden: 4 SWS
Geschätzte studentische Arbeitszeit: 120 Stunden

Lernergebnisse:

Die Studierenden beherrschen die Modellierung von dynamischen Systemen in der Zustandsdarstellung und können die in der Systemtheorie gebräuchlichen Systembeschreibungen (Laplace-Übertragungsfunktionen und Zustandsbeschreibungen) vorteilhaft einsetzen. Die Studierenden können PID, P- und PI-Zustandsregler sowie Fuzzy-Regler und erweiterte Regler-Strukturen (2-DOF, IMC) auslegen und deren jeweilige Vor- und Nachteile für eine Anwendung einschätzen. Die Studierenden sind in der Lage, Zustandsschätzer (Luenberger, Kalman, Störgrößen) zu entwerfen und in Projekten einzusetzen. Dies gilt jeweils sowohl für die analytische als auch für die numerische (Matlab/Simulink) Auslegung der jeweiligen Systeme.

Bildung der Modulnote:

Klausur

Modulbeschreibung Systemtheorie

Schlüsselworte: Mathematische Beschreibung technischer Systeme

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W07
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch		
Modulverantwortung:	Prof. Dr.-Ing. Reinhard Schmidt		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Mathematik aus Bachelor-Studium

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen mathematischer Beschreibungen technischer Systeme

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- mathematische Problemlösungsmöglichkeiten für technische System im Bereich der Forschung anzuwenden
- analoge, zeitdiskrete, nichtlineare und chaotische Systeme zu beschreiben und zu analysieren

Übergreifende Kompetenzen

Die Studierenden können

- technische Systeme auf mathematische Weise zu beschreiben, zu analysieren und zu verstehen,
- die mathematischen Methoden deterministische und stochastische Signale zu beschreiben. Sie beherrschen Methoden zur Signaldetektion und Signalschätzung.

Inhalt:

Beschreibung technischer Systeme

- Zeitkontinuierliche Systeme
- Ein- und mehrdimensionale zeitdiskrete Systeme
- Zeitdiskrete Filterbänke
- Adaptive Systeme
- Chaotische Systeme
- Nichtlineare Systeme

Beschreibung von Signalen

- Deterministische Signale
- Stochastische Signale

Transformationen der Systemtheorie

- Fourier-Transformation
- Laplace-Transformation
- z-Transformation
- Wavelet-Transformation
- Leistungsdichtespektrum, Wiener-Chintschin-Theorem,
- Karhunen-Loeve-Transformation

Zahlentheoretische Methoden in der Signalverarbeitung

- Faltung und schnelle Fouriertransformation

Methoden der Signaldetektion

Zu:

Inhaltsverzeichnis - Autonome Systeme - Business Intelligence - Mobile Computing - Modulplan

Methoden der Signalschätzung

- Filterung
- Prädiktion
- Interpolation

Literaturhinweise:

Rolf Unbehauen: Systemtheorie Band 1+2, Oldenbourg Verlag.
Kristian Kroschel: Statische Nachrichtentheorie Band 1+2, Springer Verlag.
Bernd Girod, Rudolf Rabenstein, Alexander Stenger: Einführung in die Systemtheorie,
Teubner Verlag

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform: Vorlesung mit Übungen, Nachbereitung und Prüfungsvorbereitung
Leistungskontrolle: Klausur (90 Minuten)
Anteil Semesterwochenstunden: 4 SWS
Geschätzte studentische Arbeitszeit: 150 Stunden

Lernergebnisse:

Die Studierenden beherrschen die mathematischen Konzepte der Systemtheorie.
Sie können technische Systeme mathematisch beschreiben und selbstständig Problemlösungen ableiten und weiterentwickeln.

Bildung der Modulnote:

Klausur

Modulbeschreibung Programmiermethoden der KI

Schlüsselworte: Theoretische Informatik

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W08
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch		
Modulverantwortung:	Prof. Dr. Heinrich Weber		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Eine höhere imperative Programmiersprache, Suchalgorithmen sowie elementare Logik

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die KI-Sprache PROLOG sowie die der Sprache zu Grunde liegenden Konzepte und Logikkalküle

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Probleme und ihre Lösungen als ausgezeichnete Zustände in einem Zustandsraum zu betrachten
- Probleme als gerichtete Transformation von Zuständen vom Ausgangszustand zu einer oder den möglichen Lösungen zu lösen
- die KI-Sprache PROLOG einzusetzen

Übergreifende Kompetenzen

Die Studierenden können

- Methoden der KI zielgerichtet zur Lösung von Problemen einsetzen

Inhalt:

- Ableitungssysteme
- nicht rücksetzende und rücksetzende Suchstrategien
- Navigation in Zustandsräumen, Breitensuche und Tiefensuche
- Algorithmenkonstruktion mit Hilfe von Ableitungssystemen
- Prädikatenlogik
- Normalformen
- Klauselmengen , Horn-Klauseln
- Termreduktionssysteme
- Unifikations-Algorithmus
- Resolutionskalkül
- Einführung in PROLOG
- klassische Beispiele und Übungsaufgaben zur logischen Programmierung

Literaturhinweise:

Clocksinn, W. F.: Programming in Prolog, Springer, 2013.

Schöning, U.: Logik für Informatiker, Spektrum Akademischer Verlag, 2000.

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung
Leistungskontrolle:	Klausur (90 Minuten)
Anteil Semesterwochenstunden:	3 SWS
Geschätzte studentische Arbeitszeit:	90 Stunden

Lernergebnisse:

Die Studierenden sind in der Lage, Probleme und ihre Lösungen als ausgezeichnete Zustände in einem Zustandsraum zu betrachten. Die Lösungsfindung gestaltet sich hier als gerichtete Transformation von Zuständen vom Ausgangszustand zu einer oder den möglichen Lösungen. Sie beherrschen den Umgang mit der KI-Sprache PROLOG sowie die der Sprache zu Grunde liegenden Konzepte und Logikkalküle.

Lehr- und Lernform:	Übungen (Labor)
Leistungskontrolle:	Testat
Anteil Semesterwochenstunden:	1 SWS
Geschätzte studentische Arbeitszeit:	60 Stunden

Lernergebnisse:

Die Studierenden beherrschen den praktischen Umgang mit einer KI Sprache.

Bildung der Modulnote:

Klausur

Modulbeschreibung Self-Adaptive Software

Schlüsselworte: Software Self-* Techniques

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W09
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		90 h
	Selbststudium		60 h
	Prüfungsvorbereitung		0 h
Unterrichtssprache:	Englisch oder Deutsch		
Modulverantwortung:	Prof. Hans-Gerhard Groß		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen des Softwareengineering, Software Architektur, Distributed Systems

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen von adaptiven Software-Systemen

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Geeignete Techniken zur Realisierung von adaptiven Systemen zu evaluieren, auszuwählen und integrieren.
- System Monitoring zu entwickeln und effizient einzusetzen.
- Diagnoseverfahren zu entwickeln und effizient einzusetzen.

Übergreifende Kompetenzen

Die Studierenden können

- neue Methoden und Techniken entwickeln, die eine automatische Software Adaption ermöglichen.

Inhalt:

- Individuelle Projektarbeit zum Thema Software Self-* Techniques.
- Adaptive Software Platforms, Software Monitoring and Diagnosis, Software Healing
- 24-7 Uptime Requirements of Software Systems.
- Entwicklung eines Prototypen, der die gewählte Self-* Technique realisiert.

Literaturhinweise:

Cheng, B.H.C., de Lemos, R., Inverardi, P., Magee, J. (Eds.) Software Engineering for Self-Adaptive Systems, Springer LNCS 5525, 2009.

Individuelle Literatursuche entspr. des gewählten Themas (Literature Search).

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Projekt und Seminar
Leistungskontrolle:	Projektarbeit
Anteil Semesterwochenstunden:	3 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden sind in der Lage aktuelle Themen im Bereich Software Self-* Techniques selbstständig zu recherchieren, anzuwenden und zu präsentieren.

Bildung der Modulnote:

Projektarbeit

Modulbeschreibung Software Tooling Workshop

Schlüsselworte: Software Tools, Qualität, DSL

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W10
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		90 h
	Prüfungsvorbereitung		0 h
Unterrichtssprache:	Englisch oder Deutsch		
Modulverantwortung:	Prof. Hans-Gerhard Groß		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlagen des Softwareengineering, Software Architektur

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen des Software Quality Managements

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Existierende Theorien des Software Quality Management in entsprechende Tools zur Messung von Qualität umzusetzen und prototypisch zu realisieren.

Übergreifende Kompetenzen

Die Studierenden können

- Tools für ausgewählte Phasen des Software-Lebenszyklus entwickeln und evaluieren.
- moderne Sprachen der 4. Generation effizient für die schnelle Entwicklung von Software-Tools anwenden

Inhalt:

- Individuelle Projektarbeit zum Thema Software Tools.
- Entwicklung eines Tool-Prototyps.
- Evaluierung der Technologien plus Evaluierung des Tools mittels Fallstudie.
- Dokumentation der zugrunde liegenden Technologien, des Tools, und der Evaluierung

Literaturhinweise:

Individuelle Literatursuche entspr. des gewählten Themas (Literature Search).

Wird angeboten:

Einmal pro Jahr (Sommer- oder Wintersemester)

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Seminar
Leistungskontrolle:	Seminararbeit
Anteil Semesterwochenstunden:	1 SWS
Geschätzte studentische Arbeitszeit:	30 Stunden

Lernergebnisse:

Die Studierenden sind in der Lage aktuelle Themen im Bereich Software Tooling selbstständig zu recherchieren und zu präsentieren.

Lehr- und Lernform: Projekt

Zu:

Inhaltsverzeichnis - Autonome Systeme - Business Intelligence - Mobile Computing - Modulplan

Leistungskontrolle: Projektarbeit
Anteil Semesterwochenstunden: 1 SWS
Geschätzte studentische Arbeitszeit: 120 Stunden

Bildung der Modulnote:

Bericht und Referat benotet, unbenotetes Testat

Modulbeschreibung Rechnerorganisation

Schlüsselworte: Praktische Informatik

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W11
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch		
Modulverantwortung:	Prof. Dr. Heinrich Weber		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Vorlesung Betriebssysteme, eine Programmiersprache

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen von verteilten, heterogenen Rechnernetzen, die in Rechenzentren oder von Dienstleistungs Providern vorgehalten werden

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- die erweiterte Systemprogrammierung und Systemkonfiguration zur Erstellung und zum Betrieb eines verteilten, heterogenen Rechnernetzes durchzuführen,
- Hardwarekomponenten und Treibersoftware zu integrieren

Übergreifende Kompetenzen

Die Studierenden können

- Rechnerdienste (z.B. DHCP/BOOTP, BIND, NFS, NIS, LDAP, HTTPD, SAMBA, etc.) für den Betrieb einrichten,
- Verfahren zur Abwehr von Rechnerangriffen und Methoden der Computerforensik einsetzen.

Inhalt:

- Rechner- und Betriebssystem-Bootvorgang
- Einbinden spezieller Rechner-Hardware, Treibermodule
- System-V und andere Upstart-Services
- Netzwerkimtegration und einfache Netzwerkservices
- Erweiterte Netzwerkdienste
- Userverwaltung und Authentifizierung in großen Netzen
- Serverdienste für größere Netzwerke
- Windows, Unix und Mac in einem Netzwerk
- Sicherheitsaspekte und Abwehr von Angriffen

Literaturhinweise:

Scott, M. Mitchell, E.L.: Linux System Security, Prentice Hall, 2-nd Ed., 2002
Kettner, M.: Fehlerdiagnose und Problembehebung unter Linux, Suse Press, 2004

Wird angeboten:

Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung
Leistungskontrolle:	Klausur (90 Minuten)
Anteil Semesterwochenstunden:	3 SWS
Geschätzte studentische Arbeitszeit:	90 Stunden

Lehr- und Lernform:	Übungen (Labor)
Leistungskontrolle:	Testat
Anteil Semesterwochenstunden:	1 SWS
Geschätzte studentische Arbeitszeit:	60 Stunden

Lernergebnisse:

Die Studierenden sind in der Lage ein verteiltes, heterogenes Rechnernetz zu konfigurieren und in Betrieb zu nehmen.

Bildung der Modulnote:

Klausur

Modulbeschreibung Web Services

Schlüsselworte: Serviceorientierte Architekturen, Web-Anwendungen

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W12
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		90 h
	Prüfungsvorbereitung		0 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr.-Ing. Andreas Rößler		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Grundlegende Web-Technologien, Rechnernetze, Softwareentwicklung

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- Grundlagen von Web Services
- Methoden zur Identifikation und Spezifikation von Web Services
- Konzepte von HTTP, SOAP und WSDL
- Konzepte von REST
- Best Practices für den Entwurf von REST Web Services
- Methoden zur Absicherung von Web Services
- Varianten für den Betrieb von Web Services (bspw. Docker Images)
- Methoden zur Entwicklung von REST Web Service Clients
- Grundlagen serviceorientierter Architekturen
- Unterschiede zwischen serviceorientierten Architekturen als Integrations- und strategische IT-Architekturen
- Grundlagen des Business Process Management

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Web Services auf Basis von SOAP zu entwickeln
- REST Web Services unter Berücksichtigung aktueller Best Practices zu entwickeln
- REST Web Services als Microservices in Docker Images zu deployen
- Web Service Clients zu entwickeln

Übergreifende Kompetenzen

Die Studierenden können

- verteilte Web-Architekturen mit Hilfe von Web Services konzipieren

Inhalt:

- Grundlagen der Web Services
- Spezifikation von Web Service Requirements
- Web Services mit SOAP und WSDL
- Web Services mit REST
- Best Practices für REST Web Services
- Web Service Security
- Web Service Deployment
- Entwicklung von Web Service Clients
- Serviceorientierte Architekturen und Business Process Management

Literaturhinweise:

Erl, Th. et.al.: SOA with REST. Prentice Hall 2012.
Erl, Th.: Service-Oriented Architecture: Concepts, Technology, and Design. Prentice Hall 2015
Papazoglou, M.P.: Web Services: Principles and Technology, Pearson Education, 2008.

Wird angeboten:

Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform: Vorlesung / Seminar
Leistungskontrolle: Referat (20 Minuten)
Anteil Semesterwochenstunden: 2 SWS
Geschätzte studentische Arbeitszeit: 90 Stunden

Lernergebnisse:

Die Studierenden kennen die wichtigsten Technologien und Standards zur Entwicklung von Web Services. Sie sind in der Lage die Architektur serviceorientierter Web-Anwendungen zu verstehen, zu beurteilen und anzuwenden.

Lehr- und Lernform: Projektarbeit
Leistungskontrolle: Referat (20 Minuten)
Anteil Semesterwochenstunden: 2 SWS
Geschätzte studentische Arbeitszeit: 60 Stunden

Lernergebnisse:

Die Studierenden können Anforderungen an verteilte Web-Architekturen und Web Services modellieren und die Technologien und Werkzeuge für Web Services in Projekten anwenden.

Bildung der Modulnote:

Referat (3) und Projektarbeit (2)

Modulbeschreibung Mobile Applikationen

Schlüsselworte: Apps, Client, Server, Soziale Netze

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W13
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Dr.-Ing. Harald Melcher		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Kenntnisse in der objektorientierten Programmierung

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- Plattformen und Programmiersprachen für mobile Applikationen
- Techniken zur Anpassung von Oberflächen an unterschiedlichste Anforderungen mobiler Endgeräte („responsive design“)
- Cross-Plattform-Aspekte von mobilen Anwendungen

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Oberflächen für mobile Anwendung zu entwerfen und umzusetzen
- die lokale Datenhaltung sowie die Client-Server-Kommunikation zu entwickeln und zu implementieren
- Wirtschaftlichkeitsanalysen durchzuführen

Übergreifende Kompetenzen

Die Studierenden können

- komplette Anwendungen für gängige Betriebssysteme für mobile Endgeräte erstellen

Literaturhinweise:

Apple Developer Network: developer.apple.com
Android Developer Network: developer.android.com

Wird angeboten:

Wintersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung/Projekt
Leistungskontrolle:	Klausur (90 Minuten)
Anteil Semesterwochenstunden:	4 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden sind in der Lage Applikationen für ein gängiges Betriebssystem für mobile Endgeräte erstellen zu können, wie z.B. ein Online-Spiel oder eine Marketing-Plattform. Dies schließt Icon- und Oberflächen-Entwurf, Bedien-Elemente auf verschiedenen Hardware-Plattformen, Client-Server-Kommunikation, lokale und entfernte Datenhaltung sowie Einstellen in einen entsprechenden App-Shop ein.

Die Studierenden verstehen Programmiersprachen für mobile Applikationen, sowie Plattformen und ihrer Bibliotheken. Sie erlernen die Möglichkeiten der Realisierung von Oberflächen und kennen deren Layout-Alternativen. Cross-Plattform-Aspekte, Barrierefreiheit und Location Based Services.

Bildung der Modulnote:

Klausur

Modulbeschreibung Data Mining

Schlüsselworte: Data Mining, Machine Learning, Big Data, Visual Analytics, Cluster Analysis, Classification, R, Hadoop

Zielgruppe:	1. oder 2.Semester AIM	Modulnummer:	AI W14
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Englisch		
Modulverantwortung:	Prof. Jürgen Nonnast		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

grundlegende Programmierkenntnisse
mathematische Grundlagen

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen des Data Mining: Data Mining-Prozess, Distanzmaße, Merkmalsraum
- Verfahren des maschinellen Lernens: unsupervised learning (Cluster-Analyse): partitionierendes Clustering, hierarchisches Clustering; supervised learning (Klassifikation): Klassifikatoren, z.B. k-nearest neighbours, neural networks, support vector machines
- Methoden des Visual Analytics: Scatter plots, Histogramme, Boxplots, Scatter plot matrix, Parallel Coordinates, interaktive Visualisierungen
- Grundlagen der Programmiersprache R
- Big Data-Technologien: high-performance distributed computing, Hadoop (HDFS, MapReduce Programmiermodell), In-Memory Processing mit Spark

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- vorhandene Datensätze und Strukturen zu analysieren,
- geeignete Verfahren für bestimmte Probleme auszuwählen
- die erlernten Verfahren mit Hilfe der Programmiersprache R einzusetzen

Übergreifende Kompetenzen

Die Studierenden können

- konkrete Probleme mit Hilfe von Data Mining Verfahren selbstständig lösen
- selbstständig Wissen aus Daten extrahieren

Inhalt:

Grundlagen des Data Mining:

- Data Mining-Prozess
- Distanzmaße
- Merkmalsraum

Visual Analytics:

- Scatter plots, Histogramme, Boxplots
- Scatter plot matrix, Parallel Coordinates
- interaktive Visualisierungen

Unsupervised learning (Cluster-Analyse):

- partitionierendes Clustering
- hierarchisches Clustering

Supervised learning (Klassifikation):

- Machine Learning
- Klassifikatoren, z.B. k-nearest neighbours, neural networks, support vector machines, ...

Big Data Technologien:

- Big Data Grundlagen
- Hadoop
- Hadoop Ecosystem

Praxis:

- Data Mining-Verfahren mit R
- Big Data Technologien mit dem Hadoop Ecosystem
- Projekt aus dem Umfeld Data Mining, Machine Learning, Big Data

Literaturhinweise:

Han, Jiawei: Data mining : concepts and techniques. - Amsterdam: Elsevier, 2012. - ISBN: 9780123814791

Hastie, Trevor J: The elements of statistical learning : data mining, inference, and prediction. - New York, NY: Springer, 2013. - ISBN: 9780387848570

Mazza, Riccardo: Introduction to information visualization. - London: Springer, 2009. - ISBN: 9781848002180

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung mit Übungen, Nachbereitung und Prüfungsvorbereitung
Leistungskontrolle:	Klausur (90 Minuten)
Anteil Semesterwochenstunden:	4 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Das Modul besteht aus theoretischer Vorlesung und einem praktischen Anteilen.

Nach erfolgreicher Absolvierung des Moduls sollen die Studierenden in der Lage sein, ausgewählte Data Mining-Verfahren selbstständig anzuwenden.

Die Studierenden haben Wissen erworben zu häufig verwendeten Data Mining-Verfahren und Big Data-Technologien und sollen in der Lage sein die geeignete Technologie für eine Aufgabenstellung zu wählen. Die Studierenden sollen Wissen erworben haben, das es ihnen erlaubt, selbstständig Wissen aus Daten zu extrahieren.

Bildung der Modulnote:

Klausur

Modulbeschreibung Mobile UX Design

Schlüsselworte: UX, UCD, Apps, Soziale Plattformen

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W15
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Astrid Beck		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Kenntnisse in der Programmierung

Modulziel – angestrebte Lernergebnisse:

Applikation für mobile Geräte zu entwickeln bedeutet neue Herausforderungen: Displaygröße, Kontextabhängigkeiten, neue Bedienkonzepte, unterschiedliche Außenbedingungen und die Forderung nach überzeugender User Experience (UX) müssen Entwickler meistern.

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- das User Centered Design und insbesondere die Methoden User Research, Anforderungsermittlung, Prototyping sowie Usability Test für mobile Applikationen
- Tools und Templates

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- Anforderungen, Konzeption und Design für mobile Endgeräte zu erstellen, wie z.B. ein Online-Spiel, eine Marketing-Plattform oder eine App im Automotive-Umfeld
- die unterschiedlichen Realisierungsmöglichkeiten zu unterscheiden und zu bewerten: Web, mobile Webseiten, Responsive Design, Native und Hybride Apps
- Tools und Templates in den Phasen des User Centered Design einzusetzen

Übergreifende Kompetenzen

Die Studierenden können

- Anforderungen des User Experience (UX) in Designs umsetzen
- soziale Plattformen in Apps einbinden

Inhalt:

- Statistiken: Systeme, Mobile Nutzer, Anwendungen
- Mobile Usability, Mobile UX
- User Centered design
- Projektplanung
- Marktanalyse Mitbewerber
- User Research, Kontextanalyse
- Anforderungsermittlung
- Prototyping
- Usability Test
- Realisierungsmöglichkeiten: Responsive Design, Native und Hybride Apps
- Style Guides und Normen
- Designprinzipien
- Graphische Gestaltungskonzepte, z.B. Material Design
- Zugänglichkeit/Barrierefreiheit

Zu:

Inhaltsverzeichnis - Autonome Systeme - Business Intelligence - Mobile Computing - Modulplan

- Flexibles Design: HTML5, CSS3, u.a. Media Queries, flexible Font- und Bildgrößen
- Innovative Ansätze und Nutzung

Literaturhinweise:

ISO 9241-210

ISO 9241-11

Style Guides: Android, Apple, Windows

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:

Vorlesung und Projekt

Leistungskontrolle:

Projektarbeit: Bericht und Referat (20 Minuten)

Anteil Semesterwochenstunden:

4 SWS

Geschätzte studentische Arbeitszeit:

150 Stunden

Bildung der Modulnote:

Bericht und Referat (20 Minuten)

Modulbeschreibung Web Technologien

Schlüsselworte: Web-Anwendungen, Java Server Faces

Zielgruppe:	1. oder 2. Semester AIM	Modulnummer:	AI W16
Arbeitsaufwand:	5 ECTS		150 h
Davon	Kontaktzeit		60 h
	Selbststudium		60 h
	Prüfungsvorbereitung		30 h
Unterrichtssprache:	Deutsch oder Englisch		
Modulverantwortung:	Prof. Astrid Beck		
Stand:	Jan. 2017		

Empfohlene Voraussetzungen:

Programmierkenntnisse in Java, Mobile UX Design

Modulziel – angestrebte Lernergebnisse:

Kenntnisse – fachliche Kompetenzen

Die Studierenden kennen

- die Grundlagen der Webentwicklung
- aktuelle Web-UI-Frameworks
- die Webtechnologien der Java Enterprise Edition (Java EE)

Fertigkeiten – methodische Kompetenzen

Die Studierenden sind in der Lage

- die Stärken und Schwächen von Web-UI-Frameworks beurteilen
- Web 2.0 Techniken (Ajax) einzusetzen
- Konzepte der Navigation, Datenvalidierung oder Eventing umzusetzen

Übergreifende Kompetenzen

Die Studierenden können

- eine eigene Webanwendung konzipieren und realisieren.

Inhalt:

- Grundlagen Webentwicklung
- Entwicklung eines Webshops
- Phasen des UCD
- Konzeption, Planung, Steuerung und Einbindung von Social Media
- Vorstellung aktueller Web-UI-Frameworks
- Installation von JSF
- Ein erstes kleines Beispiel
- Navigationskonzept
- Datenkonvertierung und -validierung
- Eventing
- Model-View-Controller Design Pattern
- Ajax-Grundlagen
- Verwendung verschiedener leistungsfähiger UI-Komponenten
- Bootstrap

Literaturhinweise:

Andy Bosch: Portlets und JavaServer Faces Broschiert., entwickler.Press 2009
Kai Laborenz und Andrea Ertel: Responsive Webdesign: Anpassungsfähige Websites programmieren und gestalten (Galileo Computing), 2014
ISO 9241-11

Wird angeboten:

Sommersemester

Teilgebiete und Leistungsnachweise:

Lehr- und Lernform:	Vorlesung und Projekt
Leistungskontrolle:	Bericht und Referat (20 Minuten)
Anteil Semesterwochenstunden:	4 SWS
Geschätzte studentische Arbeitszeit:	150 Stunden

Lernergebnisse:

Die Studierenden sind in der Lage Anforderungen, Konzeption und Design für einen Webshop erstellen zu können.

Die Studierenden können nutzerzentriertes Design (UCD) anwenden und praktisch umsetzen

Die Studierenden können auf Basis der Ergebnisse des User Research, der Anforderungsermittlung, des Prototyping in eine lauffähige Webanwendung unter Zuhilfenahme von JavaServer Faces umsetzen.

Die Studierenden können Anforderungen des User Experience (UX) in Designs umsetzen.
Die Studierenden können soziale Plattformen in Apps einbinden.

Bildung der Modulnote:

Bericht und Referat (20 Minuten)