



Otto-Friedrich Universität Bamberg

---

# **Modulhandbuch**

**B.Sc. Angewandte Informatik (ab 1.10.2016)**

**Wirtschaftsinformatik und Angewandte Informatik**

---



---

## Module

AI-EinfAI-B: Einführung in die Angewandte Informatik.....	9
AI-Seminar1-B: Bachelorseminar 1 der Fächergruppe Angewandte Informatik.....	11
AI-Seminar2-B: Bachelorseminar 2 der Fächergruppe Angewandte Informatik oder Informatik.....	13
AI-Thesis-B: Bachelorarbeit im Studiengang Angewandte Informatik.....	15
DSG-AJP-B: Fortgeschrittene Java-Programmierung.....	16
DSG-EiAPS-B: Einführung in Algorithmen, Programmierung und Software.....	18
DSG-EiRBS-B: Einführung in Rechner- und Betriebssysteme.....	21
DSG-PKS-B: Programmierung komplexer interagierender Systeme.....	24
DSG-Project-B: Bachelorprojekt zur Praktischen Informatik.....	26
EESYS-GEI-B: Grundlagen der Energieinformatik.....	28
EESYS-IITP-B: Internationales IT-Projektmanagement.....	30
Gdl-GTI-B: Grundlagen der Theoretischen Informatik.....	32
Gdl-MfI-1: Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik).....	34
Gdl-NPP-B: Nichtprozedurale Programmierung.....	36
Gdl-Proj-B: Bachelorprojekt Grundlagen der Informatik.....	38
Gdl-SaV-B: Logik (Specification and Verification).....	40
HCI-DISTP-B: Design Interaktiver Systeme: Theorie und Praxis.....	42
HCI-IS-B: Interaktive Systeme.....	44
HCI-KS-B: Kooperative Systeme.....	46
HCI-Proj-B: Projekt Mensch-Computer-Interaktion.....	48
HCI-US-B: Ubiquitäre Systeme.....	50
ISDL-WAWI-B: Wissenschaftliches Arbeiten in der Wirtschaftsinformatik.....	52
KInf-DigBib-B: Digitale Bibliotheken und Social Computing.....	56
KInf-GeoInf-B: Geoinformationssysteme.....	58
KInf-Projekt-B: Bachelorprojekt Kulturinformatik.....	60
KogSys-GAI-B: Genderaspekte in der Informatik.....	62
KogSys-IA-B: Intelligente Agenten.....	64
KogSys-Proj-B: Bachelor-Projekt Kognitive Systeme.....	66
KTR-Datkomm-B: Datenkommunikation.....	68

---

# Inhaltsverzeichnis

---

KTR-MfI-2: Mathematik für Informatik 2 (Lineare Algebra).....	72
KTR-Proj: Projekt Kommunikationsnetze und -dienste.....	74
Mathe-B-01 (BWL): Mathematik für Wirtschaftswissenschaftlerinnen und Wirtschaftswissenschaftler I (BWL).....	77
MI-AuD-B: Algorithmen und Datenstrukturen.....	79
MI-EMI-B: Einführung in die Medieninformatik.....	82
MI-Proj-B: Projekt zur Medieninformatik [Bachelor].....	85
MI-WebT-B: Web-Technologien.....	87
MOBI-IMP-B: Implementation of Data Management Systems.....	90
MOBI-MSS-B: Mobility in Software Systems.....	92
MOBI-PRAI-B: Bachelor Project Mobile Software Systems (AI).....	94
SEDA-DMS-B: Datenmanagementsysteme.....	96
SEDA-EuU-B: Entrepreneurship und Unternehmensgründung.....	98
SEDA-PT-B: Methoden der Präsentation, Gesprächsführung und Diskussion.....	100
SEDA-TA-B: Technikfolgeabschätzung / -bewertung.....	101
SME-Phy-B: Physical Computing.....	103
SME-Projekt-B: Bachelorprojekt zu Smart Environments.....	105
Stat-B-01: Methoden der Statistik I.....	107
Stat-B-02: Methoden der Statistik II.....	109
SWT-FSE-B: Foundations of Software Engineering.....	111
SWT-PR1-B: Bachelorprojekt Softwaretechnik und Programmiersprachen.....	113
SWT-RSD-B: Reactive Systems Design.....	115
SWT-SSP-B: Soft Skills in IT-Projekten.....	118
SWT-SWL-B: Software Engineering Lab.....	120

---

---

# Übersicht nach Modulgruppen

## 1) A1 Fachstudium Mathematische Grundlagen (Modulgruppe) ECTS: 27

Mathe-B-01 (BWL): Mathematik für Wirtschaftswissenschaftlerinnen und Wirtschaftswissenschaftler I (BWL) (3,00 ECTS, WS, SS).....	77
Gdl-Mfl-1: Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (6,00 ECTS, WS, jährlich).....	34
KTR-Mfl-2: Mathematik für Informatik 2 (Lineare Algebra) (6,00 ECTS, SS, jährlich).....	72
Stat-B-01: Methoden der Statistik I (6,00 ECTS, WS, SS).....	107
Stat-B-02: Methoden der Statistik II (6,00 ECTS, WS, SS).....	109

## 2) A2 Fachstudium Informatik (Modulgruppe) ECTS: 42 - 48

### a) BA AI Fachstudium Informatik P (Pflichtbereich) ECTS: 30

DSG-EiAPS-B: Einführung in Algorithmen, Programmierung und Software (6,00 ECTS, WS, jährlich).....	18
DSG-EiRBS-B: Einführung in Rechner- und Betriebssysteme (6,00 ECTS, SS, jährlich).....	21
MI-AuD-B: Algorithmen und Datenstrukturen (6,00 ECTS, SS, jährlich).....	79
Gdl-GTI-B: Grundlagen der Theoretischen Informatik (6,00 ECTS, SS, jährlich).....	32
SWT-FSE-B: Foundations of Software Engineering (6,00 ECTS, SS, jährlich).....	111

### b) BA AI Fachstudium Informatik WP (Wahlpflichtbereich) ECTS: 12 - 18

DSG-AJP-B: Fortgeschrittene Java-Programmierung (3,00 ECTS, SS, jährlich).....	16
DSG-PKS-B: Programmierung komplexer interagierender Systeme (3,00 ECTS, WS, jährlich).....	24
Gdl-NPP-B: Nichtprozedurale Programmierung (6,00 ECTS, WS, jährlich).....	36
Gdl-SaV-B: Logik (Specification and Verification) (6,00 ECTS, WS, jährlich).....	40
KTR-Datkomm-B: Datenkommunikation (6,00 ECTS, WS, jährlich).....	68
MOBI-IMP-B: Implementation of Data Management Systems (6,00 ECTS, WS, jährlich).....	90
MOBI-MSS-B: Mobility in Software Systems (6,00 ECTS, WS, jährlich).....	92
SEDA-DMS-B: Datenmanagementsysteme (6,00 ECTS, SS, jährlich).....	96
SWT-RSD-B: Reactive Systems Design (6,00 ECTS, SS, jährlich).....	115

## 3) A3 Fachstudium Angewandte Informatik (Modulgruppe) ECTS: 36 - 42

### a) BA AI Fachstudium Angewandte Informatik P (Pflichtbereich) ECTS: 6

---

AI-EinfAI-B: Einführung in die Angewandte Informatik (6,00 ECTS, WS, SS)..... 9

**b) BA AI Fachstudium Angewandte Informatik WP (Wahlpflichtbereich) ECTS: 30 - 36**

**aa) Kognitive Systeme (Fach) ECTS: 0 - 6**

KogSys-IA-B: Intelligente Agenten (6,00 ECTS, SS, jährlich)..... 64

**bb) Kulturinformatik (Fach) ECTS: 0 - 12**

KInf-GeoInf-B: Geoinformationssysteme (6,00 ECTS, SS, jährlich)..... 58

KInf-DigBib-B: Digitale Bibliotheken und Social Computing (6,00 ECTS, WS, jährlich)..... 56

**cc) Medieninformatik (Fach) ECTS: 0 - 12**

MI-EMI-B: Einführung in die Medieninformatik (6,00 ECTS, WS, jährlich)..... 82

MI-WebT-B: Web-Technologien (6,00 ECTS, SS, jährlich)..... 87

**dd) Mensch-Computer-Interaktion (Fach) ECTS: 0 - 18**

HCI-IS-B: Interaktive Systeme (6,00 ECTS, WS, jährlich)..... 44

HCI-KS-B: Kooperative Systeme (6,00 ECTS, SS, jährlich)..... 46

HCI-US-B: Ubiquitäre Systeme (6,00 ECTS, WS, jährlich)..... 50

**ee) Smart Environments (Fach) ECTS: 0 - 6**

SME-Phy-B: Physical Computing (6,00 ECTS, SS, jährlich)..... 103

**ff) Energieeffiziente Systeme (Fach) ECTS: 0 - 6**

EESYS-GEI-B: Grundlagen der Energieinformatik (6,00 ECTS, WS, jährlich)..... 28

**4) A4 Fachstudium Anwendungsfächer (Modulgruppe) ECTS: 27 - 33**

In dieser Modulgruppe sind Module aus Anwendungsfächern gemäß den Regelungen zu Modulgruppe A4 Fachstudium Anwendungsfächer aus Anhang 1 der StuFPO zu wählen.

WICHTIG: Die Modulprüfungen einzelner Module eines gewählten Anwendungsfachs müssen inklusive aller erforderlichen zugehörigen Modulteilprüfungen gemäß der gültigen Bestimmungen des gewählten Anwendungsfachs abgelegt werden. Ergibt sich aufgrund der Modulvorgaben des Anwendungsfachs ein Mindestvolumen, welches die genannte Minimalanforderung von 12 ECTS-Punkten überschreitet, z.B. 15 ECTS, so ist diese Vorgabe zu berücksichtigen. Die von der anbietenden Fakultät in der Studien- und Fachprüfungsordnung und im jeweiligen Modulhandbuch festgelegten Regelungen zur Anmeldung und Anrechnung von Modulteilprüfungen einzelner Module und ihrer zugeordneten Lehrveranstaltungen sind bindend. Im Hinblick auf einzelne Anwendungsfächer gelten Zugangsvoraussetzungen, so sind zum Teil Vorpraktika erforderlich. Zu den Details können die Informationen des jeweiligen Anwendungsfachs und

---

die Informationsangebote der dortigen Fachstudienberatung herangezogen werden, vgl: <http://www.uni-bamberg.de/?id=29722>

Mit der ersten Prüfungsanmeldung zu einem Modul verpflichtet sich der Studierende sowohl zur Beachtung der Zuordnungsregelungen von Lehrveranstaltungen und Modulteilprüfungen des gewählten Moduls als auch der Anerkennung der Prüfungsanmeldungs- und -anrechnungsregelungen des Moduls in dem gewählten Anwendungsfach.

## 5) A5 Kontextstudium (Modulgruppe) ECTS: 6 - 12

### a) Allgemeine Schlüsselqualifikationen (Teil-Modulgruppe) ECTS: 0 - 12

EESYS-IITP-B: Internationales IT-Projektmanagement (6,00 ECTS, SS, jährlich).....	30
HCI-DISTP-B: Design Interaktiver Systeme: Theorie und Praxis (3,00 ECTS, SS, jährlich).....	42
KogSys-GAI-B: Genderaspekte in der Informatik (3,00 ECTS, SS, jährlich).....	62
SEDA-EuU-B: Entrepreneurship und Unternehmensgründung (3,00 ECTS, WS, SS).....	98
SEDA-PT-B: Methoden der Präsentation, Gesprächsführung und Diskussion (3,00 ECTS, WS, SS).....	100
SWT-SSP-B: Soft Skills in IT-Projekten (3,00 ECTS, SS, jährlich).....	118

### b) Wissenschaftliches Arbeiten (Teil-Modulgruppe) ECTS: 0 - 12

Beachten Sie auch die Informationen des Prüfungsausschusses.

ISDL-WAWI-B: Wissenschaftliches Arbeiten in der Wirtschaftsinformatik (3,00 ECTS, WS, jährlich).....	52
--	----

### c) Fremdsprachen (Teil-Modulgruppe) ECTS: 0 - 12

Module gemäß des aktuellen Angebots des Sprachenzentrums, speziell IT English I und IT English II.

### d) Philosophie / Ethik (Teil-Modulgruppe) ECTS: 0 - 9

Aktuelle Angebote der Fakultät WIAI oder Angebote der Fakultät GuK im Umfang 0 - 6 ECTS-Punkten.

SEDA-TA-B: Technikfolgeabschätzung / -bewertung (3,00 ECTS, SS, jährlich).....	101
--	-----

## 6) A6 Seminare und Projekte (Modulgruppe) ECTS: 18

### a) Seminare (Wahlpflichtbereich) ECTS: 6

Es sind zwei Seminare zu absolvieren. Mindestens ein Seminar muss der Angewandten Informatik entstammen.

AI-Seminar1-B: Bachelorseminar 1 der Fächergruppe Angewandte Informatik (3,00 ECTS, WS, SS).....	11
AI-Seminar2-B: Bachelorseminar 2 der Fächergruppe Angewandte Informatik oder Informatik (3,00 ECTS, WS, SS).....	13

### b) Projekte (Wahlpflichtbereich) ECTS: 12

Es sind zwei Projekte zu wählen. Mindestens ein Projekt muss der Angewandten Informatik entstammen.

---

**aa) Projekte in Angewandter Informatik (Fach) ECTS: 6 - 12**

HCI-Proj-B: Projekt Mensch-Computer-Interaktion (6,00 ECTS, WS, jährlich)..... 48

KInf-Projekt-B: Bachelorprojekt Kulturinformatik (6,00 ECTS, WS, jährlich)..... 60

KogSys-Proj-B: Bachelor-Projekt Kognitive Systeme (6,00 ECTS, WS, SS)..... 66

MI-Proj-B: Projekt zur Medieninformatik [Bachelor] (6,00 ECTS, WS, jährlich)..... 85

SME-Projekt-B: Bachelorprojekt zu Smart Environments (6,00 ECTS, )..... 105

**bb) Projekte in Informatik (Fach) ECTS: 0 - 6**

DSG-Project-B: Bachelorprojekt zur Praktischen Informatik (6,00 ECTS, SS, jährlich)..... 26

Gdl-Proj-B: Bachelorprojekt Grundlagen der Informatik (6,00 ECTS, WS, SS)..... 38

KTR-Proj: Projekt Kommunikationsnetze und -dienste (6,00 ECTS, WS, jährlich)..... 74

MOBI-PRAI-B: Bachelor Project Mobile Software Systems (AI) (6,00 ECTS, SS, jährlich)..... 94

SWT-PR1-B: Bachelorprojekt Softwaretechnik und Programmiersprachen (6,00 ECTS, SS, jährlich).... 113

SWT-SWL-B: Software Engineering Lab (6,00 ECTS, WS, jährlich)..... 120

**7) A7 Bachelorarbeit (Modulgruppe) ECTS: 12**

Bachelorarbeit im Studiengang Angewandte Informatik gemäß § 35 der StufPO B. Sc. AI.

AI-Thesis-B: Bachelorarbeit im Studiengang Angewandte Informatik (12,00 ECTS, WS, SS)..... 15

---



<b>Modul AI-EinfAI-B Einführung in die Angewandte Informatik</b> <i>Introduction to Applied Computer Science</i>		6 ECTS / 180 h
(seit WS11/12) Modulverantwortliche/r: Prof. Dr. Christoph Schlieder Weitere Verantwortliche: Schmid, Ute; Henrich, Andreas; Gross, Tom		
<b>Inhalte:</b> Im Modul werden exemplarisch Fragestellungen aus den Lehrstühlen und Professuren der Angewandten Informatik betrachtet, die die Breite und den Charakter der Themen verdeutlichen sollen. Durch diese Einführung sollen die anderen Module des Studiums in ihren Kontext gestellt werden. Ferner sollen Hilfestellungen und Orientierung zur Wahl von Modulen in den Wahlpflichtbereichen des Studiums und im Bereich der Anwendungsfächer gegeben werden.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden sollen in dieser einführenden Veranstaltung einen Überblick über die Angewandte Informatik erhalten, der ihnen das Verständnis und die Einordnung der verschiedenen Veranstaltungen aus den Anwendungsfächern und die Informatik erleichtert und die Ziele und das Konzept des Bachelor-Studiengangs Angewandte Informatik verdeutlicht.		
<b>Sonstige Informationen:</b> Das Modul wird gemeinschaftlich von den Professorinnen und Professoren der Angewandten Informatik angeboten ( <a href="http://www.uni-bamberg.de/ai/">http://www.uni-bamberg.de/ai/</a> ). Der Einstieg ist zu jedem Semester möglich. Die Prüfung wird nach jedem Semester angeboten.  Die beiden Veranstaltungen hängen nicht voneinander ab. Man kann in das Modul daher sowohl im Winter als auch im Sommer einsteigen.  Der <b>Arbeitsaufwand</b> für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung (inkl. Bearbeitung der Aufgaben sowie Recherche und Studium zusätzlicher Quellen): ca. 75 Stunden</li> <li>• Prüfungsvorbereitung und Prüfung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 2 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Einführung in die AI: HCI &amp; Kognitive Systeme</b> <b>Lehrformen:</b> Vorlesung und Übung		2,00 SWS

<p><b>Dozenten:</b> Prof. Dr. Tom Gross, Ute Schmid, Mitarbeiter Mensch-Computer-Interaktion, Mitarbeiter Angewandte Informatik, insb. Kognitive Systeme  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>          Grundlagen und Konzepte zu:</p> <ul style="list-style-type: none"> <li>• Mensch-Computer-Interaktion</li> <li>• Kognitive Systeme</li> </ul> <p>Dabei werden jeweils Einführungen in typische Problemstellungen und Methoden der Fächer gegeben.</p> <hr/> <p><b>Literatur:</b>          wird in der Veranstaltung bekannt gegeben</p>	
<p><b>2. Einführung in die AI: Kulturinformatik &amp; Medieninformatik</b>  <b>Lehrformen:</b> Vorlesung und Übung  <b>Dozenten:</b> Prof. Dr. Christoph Schlieder, Prof. Dr. Andreas Henrich, Mitarbeiter Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften, Mitarbeiter Medieninformatik  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>          Grundlagen und Konzepte zu:</p> <ul style="list-style-type: none"> <li>• Kulturinformatik und</li> <li>• Medieninformatik</li> </ul> <p>Dabei werden jeweils Einführungen in typische Problemstellungen und Methoden der Fächer gegeben.</p> <hr/> <p><b>Literatur:</b>          wird in der Veranstaltung bekannt gegeben</p>	<p>2,00 SWS</p>
<p><b>Prüfung</b>          schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

<b>Modul AI-Seminar1-B Bachelorseminar 1 der Fächergruppe Angewandte Informatik</b> <i>Bachelorseminar 1 der Fächergruppe Angewandte Informatik</i>		3 ECTS / 90 h
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Eigenständige Erarbeitung und Präsentation eines Themas aus dem gewählten Fachgebiet mit wissenschaftlichen Methoden.		
<b>Lernziele/Kompetenzen:</b> Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorseminar aus einem der Fachgebiete der Angewandten Informatik zu wählen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Bachelorseminar 1</b> <b>Lehrformen:</b> Seminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS		2,00 SWS
<b>Lernziele:</b> Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
<b>Inhalte:</b> Die Inhalte der Bachelorseminare werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		
<b>Literatur:</b> Die Literatur wird zu Beginn eines Seminars von jedem anbietenden Fachgebiet bekannt gegeben.		
<b>Prüfung</b> Hausarbeit mit Referat <b>Zulassungsvoraussetzung zur Modulprüfung:</b>		

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Referat mit schriftlicher Hausarbeit zu dem im Seminar bearbeiteten Thema.

Die Bearbeitungsfrist der Hausarbeit, die Prüfungsdauer des Referats sowie die Prüfungs- und Lehrsprache werden in der Vorbesprechung zur Lehrveranstaltung von der Seminarleiterin bzw. dem Seminarleiter bekannt gegeben.

<b>Modul AI-Seminar2-B Bachelorseminar 2 der Fächergruppe Angewandte Informatik oder Informatik</b> <i>Bachelorseminar 2 der Fächergruppe Angewandte Informatik oder Informatik</i>		3 ECTS / 90 h
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Eigenständige Erarbeitung und Präsentation eines Themas aus dem gewählten Fachgebiet mit wissenschaftlichen Methoden.		
<b>Lernziele/Kompetenzen:</b> Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
<b>Sonstige Informationen:</b> Es ist ein Bachelorseminar aus einem der Fachgebiete der Angewandten Informatik oder der Informatik zu wählen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Empfohlene Vorkenntnisse werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Bachelorseminar 1</b> <b>Lehrformen:</b> Seminar <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, SS		2,00 SWS
<b>Lernziele:</b> Kompetenzerwerb in den Bereichen kritische und systematische Literaturanalyse, Strukturierung komplexer Sachverhalte, bewertender Vergleich konkurrierender Ansätze. Professionelle Präsentation von Fachthemen. Erlernen des Verfassens wissenschaftlicher Arbeiten.		
<b>Inhalte:</b> Die Inhalte der Bachelorseminare werden von jedem anbietenden Fachgebiet festgelegt und bekannt gegeben.		
<b>Literatur:</b> Die Literatur wird zu Beginn eines Seminars von jedem anbietenden Fachgebiet bekannt gegeben.		
<b>Prüfung</b>		

Hausarbeit mit Referat

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Referat mit schriftlicher Hausarbeit zu dem im Seminar bearbeiteten Thema.

Die Bearbeitungsfrist der Hausarbeit, die Prüfungsdauer des Referats sowie die Prüfungs- und Lehrsprache werden in der Vorbesprechung zur Lehrveranstaltung von der Seminarleiterin bzw. dem Seminarleiter bekannt gegeben.

<b>Modul AI-Thesis-B Bachelorarbeit im Studiengang Angewandte Informatik</b> <i>Bachelor Thesis in Applied Computer Science</i>		12 ECTS / 360 h
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Andreas Henrich Weitere Verantwortliche: Professorinnen und Professoren der Angewandten Informatik		
<b>Inhalte:</b> siehe § 35 der Studien- und Fachprüfungsordnung für den Bachelor-Studiengang Angewandte Informatik an der Otto-Friedrich-Universität Bamberg		
<b>Lernziele/Kompetenzen:</b> siehe § 35 der Studien- und Fachprüfungsordnung für den Bachelor-Studiengang Angewandte Informatik an der Otto-Friedrich-Universität Bamberg		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> siehe § 33 und § 35 der Studien- und Fachprüfungsordnung für den Bachelor-Studiengang Angewandte Informatik an der Otto-Friedrich-Universität Bamberg		
<b>Empfohlene Vorkenntnisse:</b> siehe § 33 und § 35 der Studien- und Fachprüfungsordnung für den Bachelor-Studiengang Angewandte Informatik an der Otto-Friedrich-Universität Bamberg		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> 6.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Prüfung</b> schriftliche Hausarbeit / Bearbeitungsfrist: 4 Monate <b>Zulassungsvoraussetzung zur Modulprüfung:</b> siehe § 33 der Studien- und Fachprüfungsordnung für den Bachelor-Studiengang Angewandte Informatik an der Otto-Friedrich-Universität Bamberg <b>Beschreibung:</b> siehe Modulbeschreibung	
--	--

<b>Modul DSG-AJP-B Fortgeschrittene Java-Programmierung</b>		3 ECTS / 90 h 23 h Präsenzzeit 67 h Selbststudium
<i>Advanced Java Programming</i>		
(seit WS12/13)		
Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
<b>Inhalte:</b>		
<p>Aufbauend auf den Grundkenntnissen der objekt-orientierten Programmierung in Java aus DSG-EiAPS-B soll der Umgang mit modernen objekt-orientierten Programmiersprachen durch einen genaueren Blick auf die Möglichkeiten, die eine moderne Programmierumgebung heute liefert, vertieft und gefestigt. Dazu gehören als Themen - jeweils am Beispiel 'Java' praktisch erläutert und geübt - insbesondere:</p> <ul style="list-style-type: none"> <li>• Interfaces, abstrakte Klassen und komplexere Vererbungsstrukturen, Nutzung von Package-Strukturen,</li> <li>• Einsatz und Behandlung von Exceptions,</li> <li>• Nutzung komplexer Java-APIs, z.B. für Ein- und Ausgabe,</li> <li>• grundlegende XML Verarbeitung,</li> <li>• Debugging, Profiling und Testen,</li> <li>• Überblick über das Programmieren von (grafischen) Benutzerschnittstellen (G)UIs.</li> </ul> <p>Zusätzlich werden die ersten Schritte zur Nutzung komplexer Programmierumgebungen, die über den einfachen Editor-Compiler-Ausführungs-Zyklus hinausgehen, insbesondere der Umgang mit einfachen Testszenarien zur Entwicklung verlässlicher Systeme, eingeübt.</p>		
<b>Lernziele/Kompetenzen:</b>		
Die Studierenden beherrschen die grundlegenden Mechanismen der objekt-orientierten Programmierung vertieft und sind auch in der Lage, einfache Probleme mit Hilfe der über die Standardprogrammiersprachen-Konstrukte hinausgehenden Hilfsmittel einer modernen Programmierumgebung effizient und flexibel zu lösen.		
<b>Sonstige Informationen:</b>		
Der Arbeitsaufwand von 90 Std. gliedert sich in		
<ul style="list-style-type: none"> <li>• 22.5 Std. Teilnahme an der Praktischen Übung</li> <li>• 55 Std. Bearbeiten der Programmieraufgabe (Assignment)</li> <li>• 12 Std. Vorbereitung auf das Abschlusskolloquium</li> <li>• 0.5 Std. Abschlusskolloquium inklusive Warten auf Ergebnis usw.</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
<p>Programmierkenntnisse in Java sowie Grundkenntnisse aus dem Bereich der Algorithmik und Softwareentwicklung, wie sie z.B. im Modul DSG-EiAPS-B vermittelt werden.</p> <p>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen</p>		keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester



<b>Lehrveranstaltungen</b>	
<p><b>Praktische Übung Fortgeschrittene Java-Programmierung</b></p> <p><b>Lehrformen:</b> Vorlesung und Übung</p> <p><b>Dozenten:</b> Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b></p> <p>vgl. Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b></p> <p>vgl. Modulbeschreibung</p> <hr/> <p><b>Literatur:</b></p> <p>Jedes weiterführende Buch zu Java ist verwendbar.</p>	2,00 SWS
<p><b>Prüfung</b></p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten</p> <p>Bearbeitungsfrist: 3 Monate</p> <p><b>Beschreibung:</b></p> <p>Die zu Beginn des Semesters ausgegebene Programmieraufgabe (Assignment) wird als Vorbereitung auf das Abschlusskolloquium von den Studierenden gelöst; zu den Lösungen gibt es inhaltliches Feedback und Hilfestellungen von den betreuenden Mitarbeitern schon während des Semesters. Im Abschlusskolloquium stellen die Studierenden (jeweils einzeln) die von ihnen während des Semesters erarbeiteten Lösungen zu der Programmieraufgabe vor, erläutern diese und beantworten Fragen zu ihrer Lösung und den dabei verwendeten Techniken und Sprachkonstrukten.</p>	

<p><b>Modul DSG-EiAPS-B Einführung in Algorithmen, Programmierung und Software</b>  <i>Introduction to Algorithms, Programming and Software</i></p>	<p>6 ECTS / 180 h  45 h Präsenzzeit  135 h Selbststudium</p>
<p>(seit WS13/14)  Modulverantwortliche/r: Prof. Dr. Guido Wirtz</p>	
<p><b>Inhalte:</b>  Das Modul gibt einen ersten Einblick in die Informatik aus Sicht der Entwicklung von Algorithmen und deren Realisierung durch Programme in einer imperativen, objekt-orientierten Programmiersprache (am Beispiel von Java) sowie einen Ausblick auf die Problematik der Softwareentwicklung. Behandelt werden die Grundprinzipien der Informatik zu:</p> <ul style="list-style-type: none"> <li>• Präsentation, Interpretation und Manipulation von Information,</li> <li>• Syntax und Semantik von einfachen Sprachen,</li> <li>• Probleme, Problemklassen und -Instanzen,</li> <li>• Design, Entwicklung und Implementierung von Algorithmen für einfache Problemklassen,</li> <li>• einfache Datenstrukturen wie Keller, Warteschlangen, Listen und Bäume, sowie</li> <li>• Techniken zur Spezifikation, zur Datenabstraktion und funktionalen Abstraktion.</li> </ul> <p>All diese Begriffe werden am Beispiel der Programmiersprache 'Java' diskutiert, so dass auch die wesentlichen Konzepte imperativer und objekt-orientierter Programmiersprachen wie</p> <ul style="list-style-type: none"> <li>• Wertebereiche, Namensräume, Speichermodelle und Zuweisungen,</li> <li>• Kontroll- und Datenfluss in einem Programm,</li> <li>• Iteration und Rekursion, sowie</li> <li>• Klassen, Schnittstellen, Vererbung, Polymorphie und Fehlerbehandlung</li> </ul> <p>besprochen und auch praktisch eingeübt werden.</p>	
<p><b>Lernziele/Kompetenzen:</b>  Studierende haben einen ersten Überblick über das Fach 'Informatik' mit seinen verschiedenen Gebieten und kennen die grundlegenden Begriffe, Methoden und Techniken der Informatik aus Sicht von Algorithmen, Programmiersprachen und Softwareentwicklung. Studierende sind in der Lage, geeignete Abstraktions- und Repräsentationsmethoden zur maschinellen Bearbeitung auszuwählen und Methoden zur Beschreibung von Syntax und Semantik einfacher Sprachen anzuwenden. Sie verstehen die Zusammenhänge zwischen Spezifikation und Implementierung wie auch die Arbeitsweise einer Programmiersprache und können die wesentlichen Schritte der Softwareentwicklung nachzuvollziehen. Studierende können einfache Problemstellungen beschreiben, algorithmische Lösungen dazu entwickeln und diese auch in Java mittels einfacher Datenstrukturen umsetzen.</p>	
<p><b>Sonstige Informationen:</b>  Der Arbeitsaufwand von 180 Std. verteilt sich ausgehend von einem 15 Arbeitswochen dauernden Semester in etwa wie folgt:</p> <ul style="list-style-type: none"> <li>• 22.5 Std. Vorlesungsteilnahme</li> <li>• 22.5 Std. Übungsteilnahme</li> <li>• 60 Std. Bearbeiten von wöchentlichen Übungsaufgaben (d.h. ca. 4 Std./Woche)</li> <li>• 30 Std. Vor- und Nachbereitung (Literatur, Recherchen usw.) von Vorlesung und Übung (d.h. ca. 1.5 Std./Woche ohne Bearbeiten der Übungsaufgaben)</li> </ul>	

- 45 Std. Vorbereitung auf und Zeit für die Abschlussklausur (unter Annahme der o.g. Arbeitsaufwände während des Semesters)

Bei diesen Angaben handelt es sich um Empfehlungen; es besteht weder in Vorlesung noch Übung Anwesenheitspflicht noch wird die regelmäßige Bearbeitung von Aufgaben formal überprüft. Der Gesamtaufwand für das Modul ist aber nur einzuhalten, wenn die o.g. Empfehlung in etwa eingehalten wird.

#### Zulassungsvoraussetzung für die Belegung des Moduls:

keine

#### Empfohlene Vorkenntnisse:

Die Veranstaltung hat als grundlegende Einführungsveranstaltung in das Gebiet der Informatik weder Inhalte anderer Lehrveranstaltungen noch Informatikkenntnisse oder Programmierkenntnisse zur Voraussetzung. *Insbesondere ist das Modul DSG-EiRBS-B, das regelmäßig im Sommersemester angeboten wird, keine Voraussetzung für DSG-EiAPS-B.*

#### Besondere

#### Bestehensvoraussetzungen:

keine

**Angebotshäufigkeit:** WS, jährlich

**Empfohlenes Fachsemester:**

**Minimale Dauer des Moduls:**

1 Semester

#### Lehrveranstaltungen

##### 1. DSG-EiAPS-B: Vorlesung Einführung in Algorithmen, Programmierung und Software

2,00 SWS

**Lehrformen:** Vorlesung

**Dozenten:** Prof. Dr. Guido Wirtz

**Sprache:** Deutsch

**Angebotshäufigkeit:** WS, jährlich

#### Inhalte:

vgl. Modulbeschreibung

#### Literatur:

Jede Einführung in die Informatik oder in die Programmiersprache Java kann als Ergänzung zur Veranstaltung genutzt werden, allerdings orientiert sich die Vorlesung nicht an einem Buch; deshalb ist die Liste hier nur als Auswahl "nützlicher" Bücher zu verstehen:

- Heinz-Peter Gumm, Manfred Sommer: Einführung in die Informatik. Oldenbourg Verlag, 2011 (9th)
- Barbara Liskov with John Guttag: Program Development in Java. Addison-Wesley, 2001
- Timothy Budd: An Introduction to Object-Oriented Programming, Pearson/ Addison Wesley, 2002 (3rd)
- Christian Ullenboom: Java ist auch eine Insel. Galileo Computing, 2012 (10th)
- John Lewis, Joseph Chase: Java Software Structures. Pearson/ Addison-Wesley, 2010 (3rd)

##### 2. DSG-EiAPS-B Übung

2,00 SWS

<p><b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Praktische Informatik <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b> In der Übung werden die wichtigsten Konzepte der gleichnamigen Vorlesung an einfachen Beispielen praktisch umgesetzt und durch die Besprechung von typischen Aufgaben zum jeweiligen Thema, die den Studierenden regelmäßig zum freiwilligen Üben angeboten werden, vertieft. Dabei wird insbesondere Wert auf die Vorstellung von Lösungen durch die Studierenden und deren Diskussion in der Übungsgruppe gelegt. Im Rahmen der Übungen finden auch Rechnerübungen zum Thema 'Einführung in Java und die Java-Umgebung' in den Rechnerpools der Fakultät statt, die insbesondere Programmieranfängerinnen und -anfängern den Einstieg durch vor Ort Hilfe erleichtern sollen.</p> <hr/> <p><b>Literatur:</b> vgl. Vorlesung</p>	
---	--

<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> Klausur zum Stoff des gesamten Moduls, also der Vorlesung, Übung und Rechnerübung zur DSG-EiAPS-B.</p> <p><b>Hinweis:</b> Bestehen der Klausur durch Erreichen von 50% der maximal erreichbaren Punkte. Konzeptionelle und praktische Aufgaben zu Java machen 50% der Vorlesung und auch der Klausurpunkte aus.</p>	
--	--

<b>Modul DSG-EiRBS-B Einführung in Rechner- und Betriebssysteme</b> <i>Introduction to Computer Architecture and Operating Systems</i>	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit SS14) Modulverantwortliche/r: Prof. Dr. Guido Wirtz	
<p><b>Inhalte:</b></p> <p>Die Modul bietet einen ersten Einblick in die Informatik der Systeme. Neben einer an Systemen ausgerichteten Einführung in die Informatik behandelt die Veranstaltung die Aufgaben und Architekturmerkmale von Rechner- und Betriebssystemen. Sie bietet einen Einblick in Aufbau und Architektur monolithischer Rechnersysteme. Dazu gehört neben dem schrittweisen Aufbau eines minimalen Rechners, beginnend mit aussagenlogischen Ausdrücken über ihre Realisierung durch Gatter und Standardbausteine sowie zustandsbehaftete Schaltungen und Speicherbausteinen auch die Darstellung von Daten im Rechner und ihre detaillierte Speicherung und Verarbeitung. Zusätzlich wird ein Überblick über das Zusammenspiel von Konzepten der Rechnerarchitektur mit den wichtigsten Prinzipien und Komponenten von Systemsoftware (Prozess- und Ressource-Scheduling, Speicherverwaltung, Hintergrundspeicher, I/O-Handhabung) gegeben. Die Vorlesung gibt zusätzlich einen Ausblick auf moderne Techniken der Prozessorarchitektur und Multiprozessorarchitekturen, wie sie in aktuellen Serverkonstellationen zum Einsatz kommen. Die Themen werden anhand von Modellen sowie anhand von marktgängigen Rechner- und Betriebssystemen behandelt.</p> <p><b>Bemerkung:</b> In diesem Modul wird bewusst vollständig auf die Vermittlung von Programmierkenntnissen verzichtet.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Studierende haben einen ersten Überblick über die verschiedenen Gebiete der Informatik und kennen die grundlegenden Begriffe und Methoden der Informatik wie die wichtigsten in der Informatik verwendeten Techniken sowohl aus Sicht der 'Informatik der Systeme'. Die Studierenden haben ein grundlegendes Verständnis zustandsbasierter Systeme und der darin möglichen Abläufe (Prozesse). Zusätzlich kennen sie den Aufbau moderner Rechner- und Betriebssysteme und die dabei zur Anwendung kommenden Informatiktechniken.</p>	
<p><b>Sonstige Informationen:</b></p> <p>Der Arbeitsaufwand von 180 Std. verteilt sich ausgehend von einem 15 Arbeitswochen dauernden Semester in etwa wie folgt:</p> <ul style="list-style-type: none"> <li>• 22.5 Std. Vorlesungsteilnahme</li> <li>• 22.5 Std. Übungsteilnahme</li> <li>• 60 Std. Bearbeiten von wöchentlichen Übungsaufgaben (d.h. ca. 4 Std./Woche)</li> <li>• 30 Std. Vor- und Nachbereitung (Literatur, Recherchen usw.) von Vorlesung und Übung (d.h. ca. 1.5 Std./Woche ohne Bearbeiten der Übungsaufgaben)</li> <li>• 45 Std. Vorbereitung auf und Zeit für die Abschlussklausur (unter Annahme der o.g. Arbeitsaufwände während des Semesters)</li> </ul> <p>Bei diesen Angaben handelt es sich um Empfehlungen; es besteht weder in Vorlesung noch Übung Anwesenheitspflicht noch wird die regelmäßige Bearbeitung von Aufgaben formal überprüft. Der Gesamtaufwand für das Modul ist aber nur einzuhalten, wenn die o.g. Empfehlung in etwa eingehalten wird.</p>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p>	

<b>Empfohlene Vorkenntnisse:</b> Die Veranstaltung hat als grundlegende Einführungsveranstaltung in das Gebiet der Informatik der Systeme weder Inhalte anderer Lehrveranstaltungen noch Informatikkenntnisse oder Programmierkenntnisse zur Voraussetzung. <i>Insbesondere ist das Modul DSG-EiAPS-B, das regelmäßig im Wintersemester angeboten wird, keine Voraussetzung für DSG-EiRBS-B.</i>		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

Lehrveranstaltungen	
<p><b>1. DSG-EiRBS-B: Vorlesung Einführung in Rechner- und Betriebssysteme</b>  <b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Dr. Guido Wirtz  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> vgl. Modulbeschreibung</p> <hr/> <p><b>Literatur:</b> Zum Bereich Rechnerarchitektur und Betriebssysteme gibt es eine ganze Reihe guter einführender Bücher, die aber alle über den in der Vorlesung behandelten Stoff hinausgehen. Deshalb ist die folgende Liste nur als Hinweis auf ergänzende Literatur gedacht - die Veranstaltung kann auch ohne auch nur eins dieser Bücher erfolgreich absolviert werden. Zu Beginn des Semesters wird zudem ein vollständiges, ausführliches Skript elektronisch zur Verfügung gestellt.</p> <ul style="list-style-type: none"> <li>• Tanenbaum, A.S./Austin, T.: Structured Computer Organization. Addison-Wesley, 2012 (6th)</li> <li>• Murdocca, M./Heuring, V.P.: Computer Architecture and Organization. Prentice Hall 2007 (1th)</li> <li>• Tanenbaum, A.S.: Moderne Betriebssysteme. Pearson Studium 2009 (3rd)</li> <li>• Silberschatz, A./Gagne, G./Galvin, P B.: Operating Systems Concepts. John Wiley and Sons, 2012 (9th)</li> </ul>	2,00 SWS
<p><b>2. DSG-EiRBS-B Übung</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Mitarbeiter Praktische Informatik  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> In der Übung werden die wichtigsten Konzepte der gleichnamigen Vorlesung an einfachen Beispielen praktisch umgesetzt und durch die Besprechung von typischen Aufgaben zum jeweiligen Thema, die den Studierenden regelmäßig zum freiwilligen Üben angeboten werden, vertieft. Dabei wird insbesondere Wert</p>	2,00 SWS

auf die Vorstellung von Lösungen durch die Studierenden und deren Diskussion in der Übungsgruppe gelegt.	
--	--

**Literatur:**

vgl. Vorlesung

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Beschreibung:**

Klausur zum Stoff des gesamten Moduls, also der Vorlesung und Übung zur DSG-EiRBS-B.

<b>Modul DSG-PKS-B Programmierung komplexer interagierender Systeme</b> <i>Introduction to Parallel and Distributed Programming</i>		3 ECTS / 90 h 23 h Präsenzzeit 67 h Selbststudium
(seit WS12/13) Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
<b>Inhalte:</b> Die Veranstaltung erläutert und übt den Umgang mit (explizit) parallelen Programmen und schafft damit auch ein vertieftes Verständnis für die Arbeitsweise heutiger Mehrkernprozessoren und Multiprozessoren. Dabei wird sowohl auf die grundlegenden Probleme und Techniken eingegangen als auch das praktische Entwerfen und Programmieren solcher Systeme (derzeit auf der Grundlage von Java) eingeübt. Dabei geht es um <ul style="list-style-type: none"> <li>• Prozesse und Threads,</li> <li>• Prozesskommunikation,</li> <li>• Synchronisation bei Shared Memory,</li> <li>• einfache C/S-Systeme mit TCP sockets,</li> <li>• Message-Passing im Aktor-Modell.</li> </ul> Zusätzlich wird die Problematik robuster verteilter Systeme diskutiert und ein Ausblick auf alternative Interaktionsparadigmen gegeben.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden kennen die gebräuchlichen Prozessbegriffe, die grundsätzlichen Probleme der Programmierung echt- und pseudo-paralleler Prozesssysteme sowie die grundlegenden Mechanismen zur Inter-Prozess-Kommunikation. Die Studierenden sind in der Lage, einfache parallele Programme mittels Threads zu schreiben, diese über Synchronisationsverfahren zu koordinieren sowie durch Kommunikationsmechanismen kooperativ und verlässlich zusammen arbeiten zu lassen.		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand von 90 Std. gliedert sich in <ul style="list-style-type: none"> <li>• 22.5 Std. Teilnahme an der Praktischen Übung</li> <li>• 55 Std. Bearbeiten der Programmieraufgabe (Assignment)</li> <li>• 12 Std. Vorbereitung auf das Abschlusskolloquium</li> <li>• 0.5 Std. Abschlusskolloquium inklusive Warten auf Ergebnis usw.</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Programmierkenntnisse in Java sowie Grundkenntnisse aus dem Bereich der Betriebssysteme, wie sie z.B. im Modul DSG-EiRBS-B vermittelt werden.  Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Einführung in Rechner- und Betriebssysteme (DSG-EiRBS-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester



<b>Lehrveranstaltungen</b>	
<p><b>Praktische Übung Programmierung komplexer interagierender Systeme</b>  <b>Lehrformen:</b> Vorlesung und Übung  <b>Dozenten:</b> Mitarbeiter Praktische Informatik  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b>  vgl. Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b>  vgl. Modulbeschreibung</p> <hr/> <p><b>Literatur:</b>  - wird jeweils aktuell zur Veranstaltung angegeben -</p>	2,00 SWS
<p><b>Prüfung</b>  Hausarbeit mit Kolloquium / Prüfungsdauer: 10 Minuten  Bearbeitungsfrist: 3 Monate  <b>Beschreibung:</b>  Die zu Beginn des Semesters ausgegebene Programmieraufgabe (Assignment) wird als Vorbereitung auf das Abschlusskolloquium von den Studierenden gelöst; zu den Lösungen gibt es inhaltliches Feedback und Hilfestellungen von den betreuenden Mitarbeitern schon während des Semesters. Im Abschlusskolloquium stellen die Studierenden (jeweils einzeln) die von ihnen während des Semesters erarbeiteten Lösungen zu der Programmieraufgabe vor, erläutern diese und beantworten Fragen zu ihrer Lösung und den dabei verwendeten Techniken und Sprachkonstrukten.</p>	

<p><b>Modul DSG-Project-B Bachelorprojekt zur Praktischen Informatik</b>  <i>Bachelor Project in Distributed Systems</i></p>	<p>6 ECTS / 180 h          130 h Präsenzzeit          50 h Selbststudium</p>
<p>(seit WS12/13)          Modulverantwortliche/r: Prof. Dr. Guido Wirtz</p>	
<p><b>Inhalte:</b>          Überschaubare Themen aus der aktuellen Forschungsarbeit der Arbeitsgruppe Verteilte Systeme (DSG), die ohne umfangreiche Einarbeitung zu bearbeiten sind, werden in einer zum Teil gemeinsam, zum Teil arbeitsteilig, arbeitenden Gruppe von Studierenden von der Konzeption bis zur praktischen Umsetzung im Rahmen eines 6-wöchigen Projekts durchgeführt. Dabei geht es nicht nur um die programmiertechnische Umsetzung, sondern insbesondere auch um die Entwicklung tragfähiger und mit den vorgegebenen Rahmenbedingungen kompatibler Konzepte zur Lösung der gestellten Aufgabe. In der Regel wird dazu das Studium aktueller Literatur und die Auswahl, Umsetzung und/oder Adaption zum Thema vorgeschlagener Ansätze notwendig sein. Typische Themen - die sich jeweils den aktuellen Arbeiten der DSG anpassen - sind z.B. Transformationen zwischen verschiedenen Prozesssprachen oder XML-Darstellungen, die Erstellung einfacher, neuer Werkzeuge im Kontext der Beschreibung und Analyse verteilter Systeme oder aber die Erweiterung von Werkzeugen um neue Funktionalitäten inklusive Einarbeitung in schon vorhandene Programmsysteme usw. Dabei wird sowohl durch die konkrete Themenstellung als auch die enge Betreuung und Unterstützung des Projekts darauf geachtet, dass die gestellten Aufgaben auch im (fortgeschrittenen) Bachelorstudium sinnvoll zu bearbeiten sind.</p>	
<p><b>Lernziele/Kompetenzen:</b>          Studierende sollen ein vertieftes Verständnis der bei der Durchführung von praktischen, arbeitsteilig organisierten, Softwareprojekten auftretenden Probleme wie auch von erfolgversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich der praktischen Informatik geschieht, gewinnen die TeilnehmerInnen wichtige Erfahrungen mit der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in einem wissenschaftlich ausgerichteten Arbeitsbericht.</p>	
<p><b>Sonstige Informationen:</b>          Der Arbeitsaufwand von insgesamt 180 Std. gliedert sich in etwa in:</p> <ul style="list-style-type: none"> <li>• 50 Std. Einführung, Vorstellen von Werkzeugen, Vorträge zum Projektstand</li> <li>• 30 Std. Recherchen zu und Einarbeitung in Thematik des Praktikums inkl. Vorbereitung von Kurzvorträgen</li> <li>• 80 Std. praktische Projektarbeit (Softwareentwicklung)</li> <li>• 10 Std. Abfassen des Projektberichts</li> <li>• 10 Std. Vorbereitung auf und Zeit für das Kolloquium (unter o.g. schon erbrachten Aufwänden)</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>          keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b>          Programmierkenntnisse sowie Kenntnisse in den Grundlagen des im Projekt behandelten Themengebiets.           Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen</p>	<p><b>Besondere Bestehensvoraussetzungen:</b>          keine</p>

Modul Einführung in Rechner- und Betriebssysteme (DSG-EIRBS-B) - empfohlen		
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>Projektübung zur Praktischen Informatik</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> vgl. Modulbeschreibung</p> <hr/> <p><b>Literatur:</b> - je nach Projektthematik -</p>	4,00 SWS

<p><b>Prüfung</b>  Hausarbeit mit Kolloquium / Prüfungsdauer: 15 Minuten  Bearbeitungsfrist: 2 Monate  <b>Zulassungsvoraussetzung zur Modulprüfung:</b>  Regelmäßige Teilnahme an der Lehrveranstaltung  <b>Beschreibung:</b>  Anfertigen eines schriftlichen Berichts über das im Projekt durchgeführte Softwareprojekt. Diskussion des vorliegenden Projektberichts sowie der erstellten Artefakte vor dem Hintergrund des allgemeinen Themas der Projektarbeit.</p>	
--	--

<b>Modul EESYS-GEI-B Grundlagen der Energieinformatik</b>		6 ECTS / 180 h
<i>Fundamentals of Energy Informatics</i>		
(seit WS13/14) Modulverantwortliche/r: Prof. Dr. Thorsten Staake		
<b>Inhalte:</b> Die Veranstaltung vermittelt die technischen und wirtschaftlichen Grundlagen der Energieinformatik. Kursteilnehmende erhalten einen Überblick über wichtige Fakten zur Bereitstellung und Nutzung von Energie, die Chancen und Herausforderungen bei der Einbindung erneuerbarer Energieträger sowie die Einsatzmöglichkeiten der Informations- und Kommunikationstechnologie zur Erreichung von Effizienz- und Emissionszielen. Anwendungsgebiete umfassen die Bereiche Automatisierung, Verhaltensbeeinflussung und Markt-Design.		
<b>Lernziele/Kompetenzen:</b> Nach einer erfolgreichen Teilnahme kennen die Studierenden die wesentlichen Komponenten eines zukünftigen Energienetzes, können Kosten und Nutzen abschätzen, Risiken benennen und Anwendungsbereiche ihres Methodenwissens aus der (Wirtschafts-)Informatik identifizieren.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Grundlagen der Energieinformatik</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Thorsten Staake <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	2,00 SWS
<b>Inhalte:</b> Inhalte der Vorlesung umfassen insbesondere: Grundbegriffe der Energietechnik (Arbeit, Leistung, Wirkungsgrade etc.) sowie der Energiewandlung (technische und wirtschaftliche Aspekte der Bereitstellung von Energie); konventionelle und erneuerbare Energiequellen (Potenziale, Grenzen und Umweltimplikationen); konventionelle Übertragungs- und Energienetze; Grundzüge von Smart Grids; Informations- und Kommunikationstechnik (IKT) in modernen Heizungsanlagen; Einfluss des Konsumentenverhaltens auf den Energieverbrauch; verhaltensbeeinflussende Informationssysteme; Umweltinformationssysteme in Unternehmen; Energieverbrauch von IKT; Gegenüberstellung von Ressourcenverbrauch und Nutzenpotenziale von IKT.	
<b>Literatur:</b>	

Sustainable energy – without the hot air; David JC McKay (ausgewählte Kapitel), verfügbar online unter: <a href="http://www.withouthotair.com">www.withouthotair.com</a>	
<b>2. Übung Grundlagen der Energieinformatik</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich <hr/> <b>Inhalte:</b> Anwendungen und Vertiefungen zu den Inhalten der Vorlesung.	2,00 SWS

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b> In der Klausur werden die in Vorlesung und Übung behandelten Inhalte geprüft. Es können 90 Punkte erzielt werden. Die Prüfungsdauer beträgt 90 Minuten. Durch die freiwillige Bearbeitung von semesterbegleitenden Studienleistungen können Teilnehmende 12 Punkte sammeln, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus Studienleistungen bestanden ist. Mögliche Studienleistungen sind schriftliche Hausarbeiten, Referate oder kleinere Software-Projekte. Zu Beginn der Lehrveranstaltung werden die Art der Studienleistungen sowie die Bearbeitungsdauer und die Punkte pro optionaler Studienleistung angegeben. Eine Bewertung von 1.0 kann auch ohne Punkte aus den Übungen erreicht werden.	
---	--

<b>Modul EESYS-IITP-B Internationales IT-Projektmanagement</b> <i>International IT Project Management</i>		6 ECTS / 180 h
(seit SS13) Modulverantwortliche/r: Prof. Dr. Thorsten Staake		
<b>Inhalte:</b> Die Studierenden erhalten ein Verständnis über die grundlegenden Methoden des IT-Projektmanagements. Besonderheiten internationaler Projekte werden ebenfalls diskutiert.		
<b>Lernziele/Kompetenzen:</b> Studierende sollen nach erfolgreichem Abschluss der Veranstaltung in der Lage sein, IT-Projekte in kleinen und großen Organisationen zu initiieren, planen, leiten und zu überwachen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Internationales IT-Projektmanagement</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Thorsten Staake <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich <b>Inhalte:</b> Gegenstand der Lehrveranstaltung sind Grundlagen des Managements von IT-Projekten mit ausgewählten Vertiefungen zu den spezifischen Aspekten internationaler Projekte. Eingenommen wird sowohl die Perspektive kleiner Unternehmen/Startups und großer Organisationseinheiten mit etablierten Prozessen. Die Lehrveranstaltung geht auf die Initiierung, Planung, Durchführung und das Controlling von IT-Projekten ein. Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.	2,00 SWS
<b>2. Übung Internationales IT-Projektmanagement</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich <b>Inhalte:</b> Anwendungen und Vertiefungen zu den Inhalten der Vorlesung, teilweise in Kleingruppen; wenn es die Anzahl der Teilnehmerinnen und Teilnehmer gestattet, wird ein konkretes Projekt bearbeitet.	2,00 SWS

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Beschreibung:**

In der Klausur werden die in Vorlesung und Übung behandelten Inhalte geprüft. Es können 90 Punkte erzielt werden. Die Prüfungsdauer beträgt 90 Minuten. Durch die freiwillige Bearbeitung von semesterbegleitenden Studienleistungen können Teilnehmende 12 Punkte sammeln, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus Studienleistungen bestanden ist. Mögliche Studienleistungen sind schriftliche Hausarbeiten, Referate oder kleinere Software-Projekte. Zu Beginn der Lehrveranstaltung werden die Art der Studienleistungen sowie die Bearbeitungsdauer und die Punkte pro optionaler Studienleistung angegeben. Eine Bewertung von 1.0 kann auch ohne Punkte aus den Übungen erreicht werden.

<b>Modul Gdl-GTI-B Grundlagen der Theoretischen Informatik</b> <i>Machines and Languages</i>		6 ECTS / 180 h
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> In der Veranstaltung wird die Theorie der Automaten, Sprachen und Algorithmen in ihren Grundzügen entwickelt. Das intuitiv einfach zu erfassende Modell der Turingmaschine als das Standardmodell der Berechenbarkeit und historischer Ausgangspunkt für die Entwicklung von programmierbaren Rechenmaschinen sowie der Lambda-Kalkül als Basis zum Verständnis funktionaler und anderer höherer Programmiersprachen stehen dabei im Mittelpunkt. Mit Turingmaschinen und anderer damit äquivalenter Berechnungsmodelle stößt die Veranstaltung zur Grenze dessen vor, was nach heutigem Wissen als prinzipiell maschinell berechenbar angesehen wird. Hierbei werden die wichtigsten Begriffe der Berechenbarkeits- und Komplexitätstheorie, etwa die Chomsky Hierarchie und die P/NP Komplexitätsklassen, besprochen. Über die klassischen Modelle der Algorithmentheorie hinaus werden, je nach verfügbarer Zeit, auch neuere Semantiken für nebenläufige und verteilte sowie objektorientierte Programmierung eingeführt und an Beispielen diskutiert.		
<b>Lernziele/Kompetenzen:</b> Kenntnis der wichtigsten Ergebnisse der Berechenbarkeits- und Komplexitätstheorie und den damit zusammenhängenden grundlegenden Einsichten in die Struktur und die Grenzen der Berechenbarkeit; Fähigkeit, Berechnungsmodelle unterschiedlicher Ausdruckskraft systematisch aufeinander zu reduzieren und die Turing-Äquivalenz von Programmiersprachen nachzuweisen oder zu widerlegen; Kenntnis konkreter mathematischer Grundmodelle zur Beschreibung von Algorithmus und Prozess, welche die wissenschaftlich-methodische Basis der Informatik bilden; Fähigkeit, rekursive und iterative Problemlösungen einerseits, sowie funktionale und reaktive Vorgänge andererseits gegeneinander abzugrenzen und ihre jeweilige Angemessenheit für die Modellierung praktischer Steuerungs- und Datenverarbeitungsaufgaben zu erkennen.		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden</li> <li>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig): 15 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> <li>• schriftliche Prüfung: 90 Minuten</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> gute Englischkenntnisse  Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>



		1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Vorlesung Grundlagen der Theoretischen Informatik</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Ph.D. Michael Mendler <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich		2,00 SWS
<b>Inhalte:</b> In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben. Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.		
<b>Literatur:</b> <ul style="list-style-type: none"> <li>• Hopcroft, J. E., Motwani, R., Ullman, J. D.: Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, Pearson Studium, 2002.</li> <li>• Asteroth, A., Baier, Ch.: Theoretische Informatik, Pearson Studium, 2002.</li> <li>• Martin, J. C.: Introduction to Languages and the Theory of Computation, McGraw Hill, (2nd ed.), 1997.</li> </ul>		
<b>2. Übung Grundlagen der Theoretischen Informatik</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Ph.D. Michael Mendler, N.N. <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich		2,00 SWS
<b>Inhalte:</b> Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung. Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.		
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b> Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben.		

<b>Modul Gdl-Mfl-1 Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik)</b> <i>Propositional and Predicate Logic</i>		6 ECTS / 180 h
(seit WS15/16) Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> In dieser Basisvorlesung werden die für die Informatik wesentlichen Elemente der Aussagen- und Prädikatenlogik, sowie ihre Anwendung zur Spezifikation und Analyse diskreter Strukturen eingeführt. Am Beispiel der Prädikatenlogik wird der Prozess der Abstraktion im Aufbau und der Anwendung von formalen Systemen eingehend dargestellt. Der zentrale Unterschied zwischen Syntax und Semantik und das Prinzip rekursiver Konstruktionen und induktiven Schließens werden dabei ausführlich erläutert.		
<b>Lernziele/Kompetenzen:</b> Die Fähigkeit, informell gegebene Strukturen und Prozesse der natürlichen und technischen Umwelt, speziell solche mit nicht-numerischem Charakter mit symbolischen Formalismen zu erfassen und mit Hilfe kombinatorischer und logischer Lösungsansätze zu analysieren; Die Fähigkeit zur Abstraktion und die Einsicht in die methodische Bedeutung des hierarchischen Aufbaus informatischer Systeme, des systematischen Fortschreitens von einfachen zu komplexen Beschreibungen sowie umgekehrt des inkrementellen Abstützens komplexer Problemlösungen auf elementare Lösungsbausteine; Die Kenntnis elementarer Grundbegriffe der Beweis- und Modelltheorie der klassischen Aussagen- und Prädikatenlogik.		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 60 Stunden</li> <li>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig) und Teilnahme an Rechnerübungen: 45 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> <li>• schriftliche Prüfung: 90 Minuten</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> gute Englischkenntnisse		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> Semester
<b>Lehrveranstaltungen</b>		
<b>1. Vorlesung Mathematik für Informatik 1</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Ph.D. Michael Mendler <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich <b>Inhalte:</b>		2,00 SWS

<p>In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.</p>	
<p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Ehrig, H., Mahr, B., Cornelius, F., Große-Rhode, Zeitz, M. P.: Mathematisch strukturelle Grundlagen der Informatik. Springer Verlag, 2. Aufl., 2001.</li> <li>• Grassmann, W. K., Tremblay, J.-P.: Logic and Discrete Mathematics - A Computer Science Perspective. Prentice Hall, 1996.</li> <li>• Scheinerman, E. R.: Mathematics – A Discrete Introduction. Brooks/Cole, 2000.</li> <li>• Barwise, J., Etchemendy, J: Language, Proof, and Logic. Seven Bridges Press, 2000.</li> </ul>	
<p><b>2. Übung Mathematik für Informatik 1</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Ph.D. Michael Mendler, N.N.</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <p><b>Inhalte:</b> Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.</p>	2,00 SWS
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

<b>Modul Gdl-NPP-B Nichtprozedurale Programmierung</b> <i>Functional Programming</i>		6 ECTS / 180 h
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> Die Veranstaltung beschäftigt sich mit den Grundlagen der logischen und funktionalen Programmierung als die wichtigsten Alternativen zu herkömmlichen prozeduralen Sprachen. Diese nichtprozeduralen Sprachen, welche dem deklarativen und rekursiven Programmierprinzip folgen, werden besonders für ihre hohe Programmiereffizienz und -Sicherheit geschätzt. Der systematische Aufbau einer funktionalen Programmiersprache wird schrittweise erläutert und anhand konkreter Aufgabenstellungen nachvollzogen. Ausführliche praktische Übungen mit der Programmiersprache Haskell ergänzen die theoretischen Inhalte. Besonderes Augenmerk wird auf die Einführung in polymorphe Typsysteme gelegt und ihre Anwendung in der Typprüfung und Typsynthese als automatisches Softwarevalidierungsverfahren. An Beispielen wird die deklarative Programmierung interaktiver Anwendungen nach dem synchronen Programmierprinzip (synchrone Kahn-Netzwerke) aufgezeigt.		
<b>Lernziele/Kompetenzen:</b> Fähigkeit zur Entwicklung algorithmischer Problemlösungen in nichtprozeduralen Programmiersprachen; Einsicht in die Bedeutung formaler Semantiken für die Implementierung von Programmiersprachen und die Fähigkeit, die funktionale Korrektheit einfacher Programme über ihre formale Semantik zu verifizieren; Kenntnis verschiedener Techniken zur Semantikgebung, insbesondere die denotationelle, operationelle, und Termersetzungsemantik; die Fähigkeit neue Sprachkonstrukte mit diesen Techniken zu spezifizieren; Fähigkeit, sich neue Programmiersprachen systematisch zu erarbeiten und diese in ihren Anwendungsmöglichkeiten kompetent einzuordnen; Kenntnis deklarativer Modelle interaktiver Software und die Fähigkeit, diese in einer konkreten Programmiersprache zu implementieren.		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 60 Stunden</li> <li>• Bearbeiten der Übungsaufgaben (unbenotet und freiwillig) und Teilnahme an Rechnerübungen: 45 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> <li>• schriftliche Prüfung: 90 Minuten</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> gundlegende Programmierkenntnisse, gute Englischkenntnisse Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>1. Vorlesung Nichtprozedurale Programmierung</b></p> <p><b>Lehrformen:</b> Vorlesung</p> <p><b>Dozenten:</b> Prof. Ph.D. Michael Mendler</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <p>In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.</p> <p>Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Pierce, B. C.: Types and Programming Languages, MIT Press, 2002</li> <li>• Thompson, S.: Haskell – The Craft of Functional Programming, Addison-Wesley 1999.</li> <li>• O’Keefe, R. A.: The Craft of Prolog. MIT Press, 2nd printing, 1994.</li> </ul>	2,00 SWS
<p><b>2. Übung Nichtprozedurale Programmierung</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Ph.D. Michael Mendler</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <p>Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.</p> <p>Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p>	2,00 SWS
<p><b>Prüfung</b></p> <p>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b></p> <p>Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p>	

<b>Modul Gdl-Proj-B Bachelorprojekt Grundlagen der Informatik</b> <i>Foundations of Computing Project</i>		6 ECTS / 180 h
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> Im Projektmodul werden wechselnde Themen angeboten, etwa zum Einsatz automatischer Verifikationswerkzeuge (Theorembeweiser, Modellprüfer, Verzögerungsanalyse) oder zum Bau und der Anwendung von visuellen Entwurfswerkzeugen für eingebettete Systeme (UML, Statecharts). Ein weiterer Bereich ist die prototypische Implementierung neuer algorithmischer Verfahren aus aktuellen Forschungsgebieten der Arbeitsgruppe (Informationssicherheit, Theorie verteilter Systeme, Logik).		
<b>Lernziele/Kompetenzen:</b> Fähigkeit zur selbständigen Erarbeitung von Problemlösungen, sowohl auf der Basis des erlernten Wissens und der angeeigneten Fähigkeiten aus dem Studium als auch der aktuellen wissenschaftlichen Literatur; Fähigkeit, komplexe Problemlösungsansätze im Rahmen eines systematischen ingenieurtechnischen Entwicklungsprozesses in Software umzusetzen und professionell zu dokumentieren; Fähigkeit zur Teamarbeit; Wissenschaftliche Neugier und die Ausbildung einer selbstbewussten und forschenden Einstellung zur Technik.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> grundlegende Programmierkenntnisse, insbesondere in funktionaler Programmierung; gute Englischkenntnisse  Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Grundlagen der Theoretischen Informatik (Gdl-GTI-B) - empfohlen Modul Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen Modul Nichtprozedurale Programmierung (Gdl-NPP-B) - empfohlen Modul Logik (Specification and Verification) (Gdl-SaV-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Übung Gdl Projekt</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, SS	4,00 SWS
<b>Inhalte:</b> In der Projektübung werden wechselnde Themen angeboten, etwa zum Einsatz automatischer Verifikationswerkzeuge (Theorembeweiser, Modellprüfer, Verzögerungsanalyse) oder zum Bau und der Anwendung von visuellen Entwurfswerkzeugen für eingebettete Systeme (UML, Statecharts). Ein weiterer	

Bereich ist die prototypische Implementierung neuer algorithmischer Verfahren aus aktuellen Forschungsgebieten der Arbeitsgruppe (Informationssicherheit, Theorie verteilter Systeme, Logik). Die Aufgabenstellung wird bei Ankündigung bzw. zu Beginn des Projektes bekanntgegeben.

Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.

**Literatur:**

Literatur wird bei Ankündigung bzw. zu Beginn des Projektes bekanntgegeben.

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten

Bearbeitungsfrist: 4 Monate

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Darstellung der Projektergebnisse in einer Hausarbeit und deren Verteidigung in einem Kolloquium.

Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben.

<b>Modul Gdl-SaV-B Logik (Specification and Verification)</b>		6 ECTS / 180 h
<i>Specification and Verification</i>		
(seit WS15/16)		
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b>		
<p>Nicht nur die Verifikation der funktionalen Korrektheit von Algorithmen und die funktionale Analyse verteilter und verlässlicher Systeme erfordert logisch-symbolische Verfahren. Auch viele Steuerungsprobleme in Anwendungsfeldern wie der Automatisierung von Wirtschaftsprozessen, intelligenten autonomen Agenten oder in Sicherheitsprotokollen lassen sich nur schwer mit herkömmlichen analytisch-numerischen Methoden behandeln. Dank der sich kontinuierlich verbessernden Leistungsfähigkeit moderner Rechner und der Erfolge im Gebiet der <i>Computational Logic</i> kommt der formalen Logik in der Informationstechnik wachsende Bedeutung zu. Die Vorlesung gibt einen Einblick in die Familie der Modallogiken als die wichtigsten informatikrelevanten Logiken, stellt zugehörige Implementierungstechniken und Entscheidungsverfahren vor und zeigt typische Anwendungen auf.</p>		
<b>Lernziele/Kompetenzen:</b>		
<p>Einsicht in die besondere Stellung der Modallogik zwischen Aussagenlogik und Prädikatenlogik und die Kenntnis ihrer ingenieurtechnischen Einsatzmöglichkeiten in Anwendungen, etwa der semantischen Informationsverarbeitung oder der Verifikation robuster und funktionssicherer reaktiver Systeme; Kenntnis der wichtigsten Modallogiken, ihrer Ausdruckskraft und Automatisierbarkeit, sowie die Fähigkeit für vorgegebene Anwendungen maßgeschneiderte Modallogiken selbst zu entwickeln; Fähigkeit, dynamische und reaktive Abläufe sowie komplexe verteilte Kommunikationsvorgänge in modaler und temporaler Logik zu spezifizieren und diese mit Hilfe geeigneter formaler Kalküle zu analysieren.</p>		
<b>Sonstige Informationen:</b>		
<p>Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:</p> <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung und Übungen (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden</li> <li>• Bearbeiten der Übungsaufgaben (unbenotet): 15 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
<p>grundlegende Programmierkenntnisse, gute Englischkenntnisse</p> <p>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen</p> <p>Modul Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen</p>		keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester
<b>Lehrveranstaltungen</b>		
<b>Vorlesung Logik (Specification and Verification)</b>		2,00 SWS



<p><b>Lehrformen:</b> Vorlesung  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>  In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.  Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Fagin, R., Halpern, J. Y., Moses, Y., Vardi, M. Y.: Reasoning about Knowledge. MIT Press, (2nd printing) 1996.</li> <li>• Hughes, G. E., Cresswell, M. J.: A New Introduction to Modal Logic. Routledge, (3rd reprint) 2003.</li> <li>• Popkorn, S.: First Steps in Modal Logic. Cambridge University Press, 1994.</li> <li>• Berard, B., Bidoit, M., Finkel, A., Laroussinie, F., Petit, A., Petrucci, L., Schnoebelen, Ph., McKenzie, P.: Systems and Software Verification. Springer 1999.</li> </ul>	
<b>Lehrveranstaltungen</b>	
<p><b>Übung Logik (Specification and Verification)</b></p> <p><b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Ph.D. Michael Mendler  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>  Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.  Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Fagin, R., Halpern, J. Y., Moses, Y., Vardi, M. Y.: Reasoning about Knowledge. MIT Press, (2nd printing) 1996.</li> <li>• Hughes, G. E., Cresswell, M. J.: A New Introduction to Modal Logic. Routledge, (3rd reprint) 2003.</li> <li>• Popkorn, S.: First Steps in Modal Logic. Cambridge University Press, 1994.</li> <li>• Van Benthem, J.: Modal Logic for Open Minds. CSLI Publications, Stanford, 2010.</li> <li>• Berard, B., Bidoit, M., Finkel, A., Laroussinie, F., Petit, A., Petrucci, L., Schnoebelen, Ph., McKenzie, P.: Systems and Software Verification. Springer 1999.</li> </ul>	2,00 SWS
<p><b>Prüfung</b>  schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b>  Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p>	

<b>Modul HCI-DISTP-B Design Interaktiver Systeme: Theorie und Praxis</b> <i>Design of Interactive Systems: Theory and Practice</i>		3 ECTS / 90 h
(seit SS13) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Theoretische, methodische und praktische Grundlagen des Entwurfs sowie praktisches Entwerfen einer forschungsrelevanten Aufgabenstellung.		
<b>Lernziele/Kompetenzen:</b> Ziel ist die allgemeine Einführung und Vermittlung grundlegender Praktiken, Prozesse und Methoden des Designs mit besonderem, anwendungsbezogenem Fokus auf die nutzerzentrierte Gestaltung komplexer, interaktiver Systeme		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesungseinheiten</li> <li>• Teilnahme an Gruppenbesprechungen</li> <li>• Bearbeitung der Aufgabenstellung allein und im Team</li> <li>• Vorbereitung von Besprechungen und Präsentationen</li> <li>• Prüfungsvorbereitung</li> </ul> Die Aufwände können dabei in Abhängigkeit von der Aufgabenstellung und der in der Gruppe abgestimmten Aufgabenverteilung unter den Gruppenmitgliedern unterschiedlich auf die Bereiche verteilt sein. Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Vorlesung Design Interaktiver Systeme: Theorie und Praxis</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Jochen Denzinger <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	1,00 SWS
<b>Inhalte:</b> Im Rahmen der Vorlesung werden folgende Themen behandelt: <ul style="list-style-type: none"> <li>• Designtheorie und -geschichte</li> </ul>	

<ul style="list-style-type: none"> <li>• Gestaltung multimodaler Benutzungsoberflächen</li> <li>• User-Centered Design, User Experience Design</li> <li>• Entwurfspraxis inkl. praktischer Einsatz von Methoden für den iterativen Entwurf</li> </ul> <p>In der Übung werden wechselnde Aufgaben zu den Inhalten der Lehrveranstaltung bearbeitet. Im Rahmen der Veranstaltung ist ein iterativer Entwurf als praktische Übung von den Studierenden zu erstellen. Die bearbeitete Aufgabenstellung geht deutlich über den Umfang einer normalen Übungsaufgabe hinaus und wird in kleinen Gruppen bearbeitet. Das erarbeitete Ergebnis wird dokumentiert und in einer Abschlusspräsentation vorgestellt.</p>	
<p><b>Literatur:</b></p> <p>Die Veranstaltung ist eine Zusammensetzung verschiedener Quellen; als ergänzende Quellen und zum Nachschlagen wir u.a. empfohlen:</p> <ul style="list-style-type: none"> <li>• Krippendorff, K. The Semantic Turn. A New Foundation for Design. Taylor &amp; Francis Group, Boca Raton, FL, 2006.</li> <li>• Moggridge, B. Designing Interactions. MIT Press, Cambridge, MA, 2007.</li> </ul>	
<p><b>Prüfung</b></p> <p>Kolloquium, Design interaktiver Systeme: Theorie und Praxis / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b></p> <p>Kolloquium zum Übungsverlauf und Übungsergebnissen</p>	

<b>Modul HCI-IS-B Interaktive Systeme</b> <i>Interactive Systems</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS11/12) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Theoretische, methodische und praktische Grundlagen der Mensch-Computer-Interaktion.		
<b>Lernziele/Kompetenzen:</b> Ziel ist die allgemeine Einführung und Vermittlung von grundlegenden Paradigmen, Konzepten und Prinzipien der Gestaltung von Benutzungsoberflächen. Der primäre Fokus liegt dabei auf dem Entwurf, der Implementation und der Evaluierung von interaktiven Systemen.		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen, aber ohne Bearbeitung der optionalen Studienleistungen): ca. 30 Stunden</li> <li>• Bearbeiten der optionalen Studienleistungen: insgesamt ca. 45 Stunden</li> <li>• Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul> Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse in Informatik im Umfang einer Einführung in die Informatik		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Vorlesung Interaktive Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich		2,00 SWS
<b>Inhalte:</b> Im Rahmen der Vorlesung werden die folgenden Themen behandelt: <ul style="list-style-type: none"> <li>• Einführung in die Gestaltung von Benutzungsoberflächen</li> <li>• Benutzer und Humanfaktoren</li> </ul>		

<ul style="list-style-type: none"> <li>• Maschinen und technische Faktoren</li> <li>• Interaktion, Entwurf, Prototyping und Entwicklung</li> <li>• Evaluierung von interaktiven Systemen</li> <li>• Entwicklungsprozess interaktiver Systeme</li> <li>• Interaktive Systeme im größeren Kontext und verwandte Themen</li> </ul>	
<p><b>Literatur:</b> Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> <li>• Preece, J., Rogers, Y. und Sharp, H. Interaction Design: Beyond Human-Computer Interaction. Wiley, New York, NY, 3. Auflage, 2011</li> <li>• Dix, A., Finlay, J., Abowd, G.D. und Beale, R. Human-Computer Interaction. Pearson, Englewood Cliffs, NJ, 3. Auflage, 2004.</li> </ul>	
<p><b>2. Übung Interaktive Systeme</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Mensch-Computer-Interaktion <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b> praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen</p> <hr/> <p><b>Literatur:</b> siehe Vorlesung</p>	2,00 SWS
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> In der <b>Klausur</b> können 90 Punkte erzielt werden.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art, der Umfang und die Punktezahl der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die durch optionale Studienleistungen erreichten Punkte als Bonuspunkte angerechnet. Eine 1,0 ist in der Klausur auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	

<b>Modul HCI-KS-B Kooperative Systeme</b> <i>Cooperative Systems</i>		6 ECTS / 180 h
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Theoretische, methodische und praktische Grundlagen der rechnergestützten Gruppenarbeit.		
<b>Lernziele/Kompetenzen:</b> Ziel ist die Vermittlung von grundlegenden Paradigmen und Konzepten von Rechnergestützter Gruppenarbeit (Computer-Supported Cooperative Work; CSCW) sowie die daraus resultierenden Designprinzipien und Prototypen. Dabei wird der Begriff breit gefasst; das zentrale Anliegen ist entsprechend die generelle technische Unterstützung von sozialer Interaktion, welche vom gemeinsamen Arbeiten und Lernen bis zum privaten Austausch reichen kann.		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen, aber ohne Bearbeitung der optionalen Studienleistungen): ca. 30 Stunden</li> <li>• Bearbeiten der optionalen Studienleistung: insgesamt ca. 45 Stunden</li> <li>• Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul> Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse in Informatik im Umfang einer Einführung in Algorithmen, Programmierung und Software, sowie Programmierkenntnisse in Java.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Kooperative Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich <b>Inhalte:</b>	2,00 SWS

<p>Im Rahmen der Vorlesung werden nach einer Einführung in das Thema die folgenden Themen behandelt:</p> <ul style="list-style-type: none"> <li>• Grundlegende Konzepte</li> <li>• Technologische Unterstützung für wechselseitige Information, Kommunikation, Koordination, Gruppenarbeit und Online-Gemeinschaften</li> <li>• Analyse kooperativer Umgebungen</li> <li>• Entwurf von CSCW und Groupware</li> <li>• Implementation von CSCW und Groupware</li> <li>• CSCW im größeren Kontext und verwandte Themen</li> </ul>	
<p><b>Literatur:</b> Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> <li>• Gross, T. und Koch, M. Computer-Supported Cooperative Work. Oldenbourg, München, 2007.</li> <li>• Borghoff, U.M. und Schlichter, J.H. Computer-Supported Cooperative Work: Introduction to Distributed Applications. Springer-Verlag, Heidelberg, 2000.</li> </ul>	
<p><b>2. Übung Kooperative Systeme</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Mensch-Computer-Interaktion <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b> praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen</p> <hr/> <p><b>Literatur:</b> siehe Vorlesung</p>	2,00 SWS
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> In der <b>Klausur</b> können 90 Punkte erzielt werden.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die durch optionale Studienleistungen erreichten Punkte als Bonuspunkte angerechnet. Eine 1,0 ist in der Klausur auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	

<b>Modul HCI-Proj-B Projekt Mensch-Computer-Interaktion</b>		6 ECTS / 180 h
<i>Project Human-Computer Interaction</i>		
(seit WS11/12) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Praktische Bearbeitung einer forschungsrelevanten Aufgabenstellung der Mensch-Computer-Interaktion.		
<b>Lernziele/Kompetenzen:</b> Aufbauend auf den in den Vorlesungen und Übungen des Faches Mensch-Computer-Interaktion erworbenen Kenntnissen und Fertigkeiten wird in diesem Projekt ein kleiner Prototyp mit wissenschaftlichem Bezug in einer Gruppe umgesetzt. Dabei werden die Fähigkeiten im Bereich der Systementwicklung ebenso weiterentwickelt wie die Kompetenzen in der Projektdurchführung und in der Gruppenarbeit.		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a>  Der Arbeitsaufwand für dieses Modul gliedert sich in folgende Bereiche: <ul style="list-style-type: none"> <li>• Teilnahme an einführenden Präsenzveranstaltungen</li> <li>• Teilnahme an Gruppenbesprechungen</li> <li>• Bearbeitung der Projektaufgabenstellung allein und im Team</li> <li>• Vorbereitung von Projektbesprechungen und -präsentationen</li> <li>• Prüfungsvorbereitung</li> </ul> Die Aufwände können dabei in Abhängigkeit von der Aufgabenstellung und der in der Gruppe abgestimmten Aufgabenverteilung unter den Gruppenmitgliedern unterschiedlich auf die Bereiche verteilt sein.  Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Modul Algorithmen und Datenstrukturen (MI-AuD-B) Modul Interaktive Systeme (HCI-IS-B)		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Übung Projekt Mensch-Computer-Interaktion</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Tom Gross, Mitarbeiter Mensch-Computer-Interaktion <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich		4,00 SWS
<b>Inhalte:</b>		



<p>Im Praktikum werden wechselnde Projektthemen zu den Inhalten der Lehrveranstaltungen bearbeitet. Dabei sind im Regelfall Aspekte mehrerer Lehrveranstaltungen relevant, so dass sich Teams mit Studierenden, die unterschiedliche Lehrveranstaltungen besucht haben, gut ergänzen. Die in einem Projekt bearbeitete Aufgabenstellung geht deutlich über den Umfang einer normalen Übungsaufgabe hinaus und wird in kleinen Gruppen bearbeitet. Das erarbeitete Ergebnis wird dokumentiert und in einer Abschlusspräsentation vorgestellt.</p>	
<p><b>Literatur:</b> wird in der Veranstaltung bekannt gegeben</p>	
<p><b>Prüfung</b> Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 4 Monate <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung <b>Beschreibung:</b> Dokumentation des Systems und des Entwicklungsprozesses sowie Kolloquium zum System und zum Entwicklungsprozess.</p>	

<b>Modul HCI-US-B Ubiquitäre Systeme</b> <i>Ubiquitous Systems</i>		6 ECTS / 180 h
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Theoretische, methodische und praktische Grundlagen des Ubiquitous Computing.		
<b>Lernziele/Kompetenzen:</b> Ziel ist die Vermittlung fundierter Kenntnisse und Fähigkeiten im Bereich der ubiquitären Systeme sowie eines breiten theoretischen und praktischen Methodenwissens zum Entwurf, zur Konzeption und zur Evaluierung ubiquitärer Systeme. Nach dem Besuch dieser Lehrveranstaltung sollen Studierende die einschlägige Literatur und Systeme in Breite und Tiefe kennen und neue Literatur und Systeme kritisch bewerten können.		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/hci/leistungen/studium">http://www.uni-bamberg.de/hci/leistungen/studium</a> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung (inkl. Recherche und Studium zusätzlicher Quellen sowie Bearbeitung der optionalen Studienleistungen): ca. 75 Stunden</li> <li>• Prüfungsvorbereitung: ca. 30 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul> Der Unterricht erfolgt grundsätzlich in deutscher und bei Bedarf der Studierenden in englischer Sprache. Sämtliche Unterlagen (inkl. Prüfung) sind in englischer Sprache verfügbar.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Modul Algorithmen und Datenstrukturen (MI-AuD-B) Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B)		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Ubiquitäre Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	2,00 SWS
<b>Inhalte:</b> Im Rahmen der Vorlesung werden nach einer Einführung in das Thema Ubiquitous Computing - also der allgegenwärtigen Rechner, die verschwindend	

<p>klein, teilweise in Alltagsgegenständen eingebaut, als Client und Server fungieren und miteinander kommunizieren können - die folgenden Themen konzeptionell, technisch und methodisch behandelt:</p> <ul style="list-style-type: none"> <li>• Grundlegende Konzepte</li> <li>• Basistechnologie und Infrastrukturen</li> <li>• Ubiquitäre Systeme und Prototypen</li> <li>• Kontextadaptivität</li> <li>• Benutzerinteraktion</li> <li>• Ubiquitäre Systeme im größeren Kontext und verwandte Themen</li> </ul>	
<p><b>Literatur:</b> Die Veranstaltung ist eine Zusammenstellung verschiedener Quellen; als ergänzende Quelle und zum Nachschlagen wird empfohlen:</p> <ul style="list-style-type: none"> <li>• Krumm, J., (Hrsg.). Ubiquitous Computing Fundamentals. Taylor &amp; Francis Group, Boca Raton, FL, 2010.</li> </ul>	
<p><b>2. Übung Ubiquitäre Systeme</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Mensch-Computer-Interaktion <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b> praktische Aufgaben zum Vorlesungsstoff einschließlich der Programmierung kleiner Prototypen.</p> <hr/> <p><b>Literatur:</b> siehe Vorlesung</p>	2,00 SWS
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> In der <b>Klausur</b> können 90 Punkte erzielt werden.</p> <p>Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die durch optionale Studienleistungen erreichten Punkte als Bonuspunkte angerechnet. Eine 1,0 ist in der Klausur auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	

<b>Modul ISDL-WAWI-B Wissenschaftliches Arbeiten in der Wirtschaftsinformatik</b>		3 ECTS / 90 h
<i>Wissenschaftliches Arbeiten in der Wirtschaftsinformatik</i>		
(seit WS13/14) Modulverantwortliche/r: Prof. Dr. Tim Weitzel		
<b>Inhalte:</b> Der Kurs beginnt mit einer allgemeinen Einführung in das wissenschaftliche Arbeiten sowie mit einem Überblick über die verschiedenen Forschungsmethoden, die im Rahmen der Wirtschaftsinformatik häufig verwendet werden (vgl. Wilde/Hess 2007). Nach einer Einführung in die Literaturanalyse und Literaturverwaltung, wird vor allem näher auf den Begriff der Theorie, qualitative und quantitative Forschung, Experimente sowie Design Science Research eingegangen.		
<b>Lernziele/Kompetenzen:</b> Ziel dieser Veranstaltung ist es, Studenten einen Einblick in das wissenschaftliche Arbeiten zu geben. Die Veranstaltung richtet sich dabei speziell an Studierende der Wirtschaftsinformatik und interessierte Studenten anderer Studiengänge, die im Bereich Wirtschaftsinformatik eine Projekt-, Seminar-, Bachelor-, oder Masterarbeit schreiben möchten.  Die Studenten sollen dabei im einzelnen folgende Lernziele erreichen:  1) Selbstständiges Suchen von Literatur und erste Möglichkeiten zur Auswertung der Literatur kennenlernen und anwenden können  2) Grundbegriffe des wissenschaftlichen Arbeitens kennenlernen und verstehen  3) Grundzüge der qualitativen und quantitativen Forschung, von Experimenten und des Design Science Research verstehen und anwenden können  4) Verstehen, wann welche Methodik bei der Bearbeitung von wissenschaftlichen Arbeiten angewandt werden kann und soll.		
<b>Sonstige Informationen:</b> Die Veranstaltung richtet sich gezielt an Studenten, die noch keine oder wenige Kenntnisse haben. Deshalb wird während der Veranstaltung jedes Thema anhand von Übungsaufgaben (sofern möglich auch am PC) praktisch vertieft. Hierzu werden unter anderem Citavi, MAXQDA, Excel, SPSS und SmartPLS verwendet.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Übung: Wissenschaftliches Arbeiten in der Wirtschaftsinformatik</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Wirtschaftsinformatik, insb. Informationssysteme in Dienstleistungsbereichen		2,00 SWS 3 ECTS

**Sprache:** Deutsch

**Angebotshäufigkeit:** WS, jährlich

**Inhalte:**

Die Veranstaltung gliedert sich in sieben Themenschwerpunkte.

1. Grundlagen: Aufbau eines Verständnisses was wissenschaftliches Arbeiten bedeutet, welche Anforderungen an das Schreiben einer wissenschaftlichen Arbeit gestellt werden und wie eine Forschungsarbeit strukturiert werden sollte. Diskussion des Zusammenspiels von Methode, Hypothese und Theorie sowie Kennenlernen der Bedeutung und Formulierung von Forschungsfragen.
2. Literaturanalyse: Aufbau eines Verständnisses, wie eine Literaturanalyse durchgeführt wird, welche Bedeutung Literatur im Rahmen von wissenschaftlichen Arbeiten besitzt, wie Literatur strukturiert gesucht und ausgewertet werden kann, sowie wie die Qualität von wissenschaftlichen Quellen beurteilt werden kann. Zudem Einführung in die Verwendung von Citavi zur Literaturverwaltung.
3. Theorien: Überblick über verschiedene Arten von Theorien. Aufbau eines Verständnisses für die Unterscheidung zwischen explorativer und konfirmatorischer bzw. induktiver und deduktiver Forschung. Diskussion der Bausteine und Inhalte von ausgewählten Theorien der Wirtschaftsinformatik.
4. Fallstudien/Qualitative Forschung: Aufbau eines Verständnisses für welche Arten von Forschungsfragen die Verwendung von Fallstudien eine geeignete Methodik ist. Diskussion der Unterschiede zwischen quantitativer und qualitativer Forschung sowie der Schritte, die im Rahmen von Fallstudien durchgeführt werden müssen und was es dabei zu beachten gilt. Einführung in das Erstellen eines Interviewleitfadens, sowie Einführung in die Software MAXQDA, zur Auswertung von qualitativen Daten.
5. Quantitative Forschung: Aufbau eines Verständnisses für welche Forschungsfragen sich quantitative Methoden eignen, in welche Phasen sich ein empirisches Forschungsprojekt gliedert, sowie wie ein Forschungsmodell im Rahmen von quantitativer Forschung aufgebaut ist. Einführung in den Zusammenhang zwischen latenten Variablen, Indikatoren, Skalen und Hypothesen. Einführung in Datenauswertungssoftware und Verfahren wie SPSS und PLS.
6. Experimente: Aufbau eines grundlegenden Verständnisses von Experimenten und Diskussion der Gebiete, in welchen Experimente in der Forschung eingesetzt werden können. Diskussion der Unterschiede zwischen Experimenten in den Wirtschaftswissenschaften und der Psychologie.
7. Design Science Research: Aufbau eines Verständnisses der grundlegenden Anforderungen an eine gestaltungsorientierte Forschung und wie gestaltungs- und verhaltensorientierte Forschung in der Wirtschaftsinformatik zusammenspielen. Einführung in Methoden mit welchen Design Science Research evaluiert werden kann.

**Literatur:**

- Backhaus, Klaus (2008): Multivariate Analysemethoden: Eine anwendungsorientierte Einführung. 12. Aufl. Berlin: Springer.

- Bühl, Achim (2008): SPSS 16. Einführung in die moderne Datenanalyse. 11. Aufl. München: Pearson Studium.
- Chin, W.W. "The Partial Least Squares Approach to Structural Equation Modeling," in: Modern Methods for Business Research, G.A. Marcoulides (ed.), Lawrence Erlbaum Associates, 1998b, pp. 295-336.
- Dubé, L.; Paré, G.: Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations. MIS Quarterly, Vol. 27, No. 4, 2003, pp. 597-635.
- Eisenhardt, K. M.; Graebner, M. E.: Theory Building from Cases: Opportunities and Challenges. Academy of Management Journal, Vol. 50, No. 1, 2007, pp. 25-32.
- Eisenhardt, K.M. "Building Theories from Case Study Research," Academy of Management Review (14:4) 1989, pp 532-550.
- Fettke, Peter (2006): Eine Untersuchung der Forschungsmethode „Review“ innerhalb der Wirtschaftsinformatik. In: Wirtschaftsinformatik, Jg. 48, H. 4, S. 257–266.
- Götz, O., and Liehr-Gobbers, K. "Analyse von Strukturgleichungsmodellen mit Hilfe der Partial-Least-Squares(PLS)-Methode," Die Betriebswirtschaft (64:6) 2004, pp 714-738.
- Hevner, A. R., March, S. T., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," MIS Quarterly (28:1), pp. 75-105.
- Lee, A.S. "Methodology for MIS Case Scientific Studies," MIS Quarterly (13:1) 1989, pp 33-50.
- Peffers, K., Tuunanen, T., Gengler, C., Rossi, M., Hui, W., Virtanen, V. and Bragge, J. (2006): The design science research process: a model for producing and presenting information systems research. In Proceedings of the First International Conference DESRIST, pp. 83–106.
- Sutton, Robert I.; Staw, Barry M. (1995): What Theory is Not. In: Administrative Science Quarterly, Jg. 40, S. 371-384.
- Webster, Jane; Watson, Richard T. (2002): Analyzing the Past to Prepare for the Future: Writing a Literature Review. In: MIS Quarterly, Jg. 26, H. 2, S. xiii–xxiii.
- Weitzel, T., Beimborn, D., König, W. (2006): A Unified Economic Model of Standard Diffusion: The Impact of Standardization Cost, Network Effects, and Network Topology. MIS Quarterly (30, Special Issue), pp. 489-514.
- Whetten, David A. (1989): What Constitutes a Theoretical Contribution? In: Academy of Management Review, Jg. 14, H. 4, S. 490-495.
- Wilde, T./ Hess, T. (2006): Methodenspektrum der Wirtschaftsinformatik: Überblick und Portfoliobildung, Arbeitsbericht Nr. 2/2006
- Winter, R. (2008): Design science research in Europe, European Journal of Information Systems 17, pp. 470-475.
- Yin, Robert K. (1996): Case Study Research: Design and Methods. 2. Aufl. Thousand Oaks: Sage Publications.

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten

**Beschreibung:**

In der Klausur werden die in der Übung behandelten Inhalte geprüft. Es können 60 Punkte erzielt werden. Während des Semesters erfolgt die Ausgabe von Übungsaufgaben zur (freiwilligen) Bearbeitung. Die Lösungen werden bewertet und bei bestandener Klausur (in der Regel sind hierzu 30 Punkte erforderlich) für die Berechnung der Note berücksichtigt. Eine 1,0 ist dabei auch ohne Punkte aus den Übungsaufgaben erreichbar.

<b>Modul KInf-DigBib-B Digitale Bibliotheken und Social Computing</b>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
<i>Digitale Bibliotheken und Social Computing</i>		
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Christoph Schlieder		
<b>Inhalte:</b> Das Modul führt ein in die Grundlagen Digitaler Bibliotheken und in die Verwaltung von Wissensbeständen mit Verfahren des Social Computing. Es besteht aus zwei Lehrveranstaltungen: einer Vorlesung, die Konzepte und Methoden vermittelt sowie einer Übung, in der die Anwendung der Methoden auf konkrete Problemstellungen eingeübt wird.  Eine weitergehende Inhaltsbeschreibung findet sich bei den Lehrveranstaltungen.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden lernen Grundbegriffe sowie wichtige Methoden aus dem Bereich der Digitalen Bibliotheken und Social Computing kennen. Sie erwerben folgende Kompetenzen: <ul style="list-style-type: none"> <li>• grundlegende Datenmodelle und Funktionen von digitalen Bibliotheken und Archiven zu vergleichen und in Bezug auf eine fachliche Problemstellung zu bewerten</li> <li>• grundlegende Methoden des Social Computing auf die Verwaltung von textuellen und nicht-textuellen Wissensbeständen anzuwenden</li> </ul>		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung: 15 Stunden</li> <li>• Vor- und Nachbereitung der Projektübung inkl. Recherche und Studium zusätzlicher Quellen aber ohne Bearbeitung der Projektübungsaufgaben: 30 Stunden</li> <li>• Bearbeiten der Projektübungsaufgaben: 60 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse in Algorithmen und Datenstrukturen, wie sie in dem empfohlenen Modul vermittelt werden  Modul Algorithmen und Datenstrukturen (MI-AuD-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Vorlesung Digitale Bibliotheken und Social Computing</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Christoph Schlieder <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	2,00 SWS



<p><b>Inhalte:</b> Digitale Bibliotheken im engeren Sinne organisieren Bestände digitaler Dokumente wie Texte, Bilder, Filme oder Tonaufzeichnungen und bieten diese über verschiedene Bibliotheksdienste den Nutzern an. Im Vordergrund steht dabei das Problem, die Inhalte der Bibliothek auf einheitliche und intuitive Weise zugänglich zu machen, d.h. das Problem der Informationssuche. Jenseits dieser klassischen Funktionen befassen sich digitale Bibliotheken im weiteren Sinn auch mit Fragen der Analyse von Inhalten und der Organisation von Wissensbeständen (Content Management, Knowledge Management). So helfen beispielsweise Technologien der Informationsvisualisierung beim Navigieren im Inhaltsangebot. Mit Methoden des Social Computing lässt sich einerseits die Vernetzung der Inhalte (Links, Zitationen, ...) andererseits die Vernetzung der Inhalte mit Akteuren (Autoren, Lesern) erfassen. Behandelt werden in diesem Zusammenhang Verfahren der Zitationsanalyse und Ansätze für Recommender Systems.</p>	
<p><b>Literatur:</b> Arms, William (2001): Digital libraries. Cambridge, MA: MIT Press. Langville, A. &amp; Meyer, C. (2006): Google's PageRank and beyond. The Science of Search Engine Rankings. Princeton, N.J: Princeton University Press. Breslin, J., Passant, A. &amp; Decker, S. (2009): The Social Semantic Web. Berlin: Springer.</p>	
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Beschreibung:</b> In der schriftlichen Prüfung werden die in der Vorlesung behandelten Themengebiete geprüft. Die Note der Klausur geht zu 50% in die Modulnote ein.</p>	
<p><b>Lehrveranstaltungen</b></p>	
<p><b>Projektübung Digitale Bibliotheken und Social Computing</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Mitarbeiter Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich</p>	2,00 SWS
<p><b>Inhalte:</b> Die Projektübung bietet eine praktische Vertiefung zu Themen der Digitalen Bibliotheken. Anhand wechselnder Themenstellungen wird das konzeptuelle Herangehen an Problemstellungen im Bereich Digitaler Bibliotheken sowie das Entwickeln passender Softwarelösungen eingeübt.</p>	
<p><b>Prüfung</b> schriftliche Hausarbeit, Hausarbeit / Bearbeitungsfrist: 4 Monate <b>Beschreibung:</b> Die Hausarbeit besteht aus der schriftlichen Bearbeitung von 3-6 im Laufe des Semesters gestellten Übungsaufgaben. Die Note der Hausarbeit geht zu 50% in die Modulnote ein.</p>	

<b>Modul KInf-GeoInf-B Geoinformationssysteme</b> <i>Geoinformationssysteme</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS12/13) Modulverantwortliche/r: Prof. Dr. Christoph Schlieder		
<b>Inhalte:</b> Das Modul führt ein in die Grundlagen der Geoinformationsverarbeitung. Es besteht aus zwei Lehrveranstaltungen: einer Vorlesung, die Konzepte und Methoden vermittelt sowie einer Übung, in der die Anwendung der Methoden auf konkrete Problemstellungen eingeübt wird. Eine weitergehende Inhaltsbeschreibung findet sich bei den Lehrveranstaltungen.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden lernen Grundbegriffe sowie wichtige Methoden aus dem Bereich der Geoinformationssysteme kennen. Sie erwerben folgende Kompetenzen: <ul style="list-style-type: none"> <li>• fachliche Anforderungen im Hinblick auf die Geodatenmodellierung zu analysieren und passende Geodatenmodelle zu erstellen</li> <li>• geoinformatische Analyseverfahren vergleichend zu bewerten und die für ein Anwendungsproblem geeigneten Verfahren zu identifizieren.</li> </ul>		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Vorlesung und Übung: 45 Stunden</li> <li>• Vor- und Nachbereitung der Vorlesung: 30 Stunden</li> <li>• Vor- und Nachbereitung der Übung inkl. Recherche und Studium zusätzlicher Quellen aber ohne Bearbeitung der Übungsaufgaben: 30 Stunden</li> <li>• Bearbeiten der Übungsaufgaben: 45 Stunden</li> <li>• Prüfungsvorbereitung: 30 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse der Informatik, wie sie in den empfohlenen Modulen vermittelt werden Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Informatik und Programmierkurs für die Kulturwissenschaften (KInf-IPKult-E) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Vorlesung Geoinformationssysteme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Christoph Schlieder		2,00 SWS

<p><b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>          Geoinformationssysteme (GIS) dienen der effizienten Erfassung, Analyse und Bereitstellung georeferenzierter Daten. Die Lehrveranstaltung stellt die grundlegenden Konzepte vor, die der Modellierung von Geodaten zugrunde liegen. Hierzu gehört z.B. die unterschiedliche Repräsentation räumlicher Objekte in Vektor- und Raster-GIS. Weitere Themen sind die Geodaten-Erfassung sowie Ansätze zur Geodatenvisualisierung. Anwendungen der Geoinformationsverarbeitung werden an klassischen Einsatzfeldern (Umweltinformationssysteme) und aktuellen technologischen Entwicklungen (mobile Computing) illustriert. Querverbindungen zum Bereich der Semantischen Informationsverarbeitung ergeben sich vor allem im Zusammenhang mit der Interoperabilität von GIS.</p> <hr/> <p><b>Literatur:</b>          Longley, P., Goodchild, M., Maguire, D., Rhind, D. (2001): Geographic Information: Systems and Science, Wiley: Chichester, UK.          Shekhar, S., Chawla, S. (2003): Spatial Databases: A Tour, Prentice Hall: Upper Saddle River, NJ.          Smith, M., Goodchild, M., and Longley, P. (2007): Geospatial Analysis, 2nd edition, Troubador Publishing Ltd.</p>	
<p><b>2. Übung Geoinformationssysteme</b></p> <p><b>Lehrformen:</b> Übung  <b>Dozenten:</b> Mitarbeiter Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>          siehe Vorlesung</p> <hr/> <p><b>Literatur:</b>          siehe Vorlesung</p>	2,00 SWS
<p><b>Prüfung</b>          schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b>          In der schriftlichen Prüfung werden die in Vorlesung und Übung behandelten Themengebiete geprüft.</p>	

<b>Modul KInf-Projekt-B Bachelorprojekt Kulturinformatik</b> <i>Bachelorprojekt Kulturinformatik</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Christoph Schlieder		
<b>Inhalte:</b> Das Modul behandelt die praktische Anwendung grundlegender Methoden aus dem Bereich der Kulturinformatik im Rahmen eines Softwareentwicklungsprojekts. Die behandelten Problemstellungen stammen aus den Anwendungsfeldern der Angewandten Informatik der Kultur-, Geschichts- und Geowissenschaften.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden lernen im Projekt wie man mit Methoden der Kulturinformatik eine Softwarelösung für eine Problemstellung entwickelt. Sie erwerben folgende Kompetenzen: <ul style="list-style-type: none"> <li>• grundlegende Methoden aus dem Bereich der Kulturinformatik auf eine fachliche Problemstellung anzuwenden</li> <li>• ein Softwareentwicklungsprojekt unter Anleitung zu planen und selbständig durchzuführen</li> <li>• eine Softwarelösung zu konzipieren und zu implementieren</li> <li>• einen Lösungsansatz sowohl aus der Fachsicht wie in seinen informatischen Details darzustellen</li> </ul>		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Gruppen- und Einzelbesprechungen: 45 Stunden</li> <li>• Vor- und Nachbereitung inkl. Recherche und Studium zusätzlicher Quellen aber ohne Bearbeitung der Projektaufgaben: 30 Stunden</li> <li>• Bearbeiten der Projektaufgaben: 90 Stunden</li> <li>• Kolloquiumsvorbereitung: 15 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Die Inhalte der Veranstaltungen "Algorithmen und Datenstrukturen" sowie "Softwaretechnik" (oder entsprechende Vorkenntnisse) werden vorausgesetzt. Modul Algorithmen und Datenstrukturen (MI-AuD-B) -		<b>Besondere Bestehensvoraussetzungen:</b> Bestehen der Softwareentwicklungsaufgabe, Ausarbeitung und Kolloquium.
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Übung Bachelorprojekt Kulturinformatik</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Christoph Schlieder, Mitarbeiter Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	4,00 SWS
<b>Inhalte:</b>	

Wechselnde Themen aus dem Bereich der Kulturinformatik	
<b>Literatur:</b> Aktuelle Literatur wird in der Veranstaltung vorgestellt.	

<b>Prüfung</b> Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten Bearbeitungsfrist: 4 Monate <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung <b>Beschreibung:</b> Im Laufe des Semesters wird eine größere Softwareentwicklungsaufgabe bearbeitet und in Form einer Hausarbeit dokumentiert. Im Kolloquium stellen die Teilnehmer ihren Arbeitsprozess und ihr Arbeitsergebnis vor. In die Leistungsbewertung geht die Hausarbeit zu 67% und das Kolloquium zu 33% ein.	
---	--

<b>Modul KogSys-GAI-B Genderaspekte in der Informatik</b> <i>Gender Aspects of Computer Science</i>		3 ECTS / 90 h 15 h Präsenzzeit 75 h Selbststudium
(seit WS15/16) Modulverantwortliche/r: Ute Schmid		
<b>Inhalte:</b> In der Veranstaltung werden theoretische Ansätze und empirische Befunde zu geschlechtsspezifischen Aspekten in der Informatik behandelt, beispielsweise: Geschlechtsstereotype und Studienfachwahl, Informatik in der Schule, Image der Informatik, Einfluss von Rollenmodellen, Barrieren für berufliche Weiterentwicklung. Ein ausgewählter Aspekt wird praktisch bearbeitet, beispielsweise: Mentoring für Schülerinnen, Entwicklung eines Unterrichtsmoduls zur Informatik, Entwickeln einer Image-Kampagne.		
<b>Lernziele/Kompetenzen:</b> Einblick in Forschungsfragestellungen im Bereich Genderstudies, Verständnis sozialwissenschaftlicher Theorien und empirischer Forschungsmethoden, Kenntnis von Maßnahmen zur Förderung von Mädchen und Frauen in der Informatik, Einblick in Methoden der Evaluationsforschung, Praktische Erfahrung mit der Konzeption, Umsetzung und Evaluation von Maßnahmen.		
<b>Sonstige Informationen:</b> Das Seminar findet teilweise gemeinsam mit dem Seminar Genderaspekte in der Wirtschaftsinformatik statt, dass im Bachelor Wirtschaftsinformatik im Fachgebiet SNA angeboten wird.  Der Arbeitsaufwand von 90 Stunden gliedert sich in etwa wie folgt:  21 Std. Teilnahme an den Präsenzveranstaltungen 24 Std. Literaturarbeit, inklusive Vorbereitung von Kurzpräsentationen 30 Std. Konzeption und Umsetzung des Praxisteils 15 Std. Erstellung der schriftlichen Ausarbeitung		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine Einschränkung		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Genderaspekte in der Informatik</b> <b>Lehrformen:</b> Seminar <b>Dozenten:</b> Ute Schmid <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	2,00 SWS
<b>Inhalte:</b> siehe Modulbeschreibung	
<b>Literatur:</b>	

wird in der Veranstaltung bekannt gegeben	
---	--

**Prüfung**

Hausarbeit mit Referat / Prüfungsdauer: 30 Minuten

Bearbeitungsfrist: 4 Wochen

**Beschreibung:**

Referat mit schriftlicher Hausarbeit zu dem im Seminar bearbeiteten Thema.

<b>Modul KogSys-IA-B Intelligente Agenten</b> <i>Intelligent Agents</i>		6 ECTS / 180 h
(seit SS13) Modulverantwortliche/r: Ute Schmid		
<b>Inhalte:</b> Die Veranstaltung vermittelt grundlegendes Wissen und Kompetenzen im Bereich Kognitiv orientierte Künstliche Intelligenz mit Fokus auf Problemlösen und Planung.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden können: <ul style="list-style-type: none"> <li>• grundlegende Forschungsziele der Künstlichen Intelligenz nennen</li> <li>• grundlegende Forschungsfragen der Künstlichen Intelligenz erläutern</li> <li>• gegebene Planungsprobleme in formalen Sprachen modellieren</li> <li>• zentrale formale Methoden des Problemlösens, des Planens und des deduktiven Schließens nennen und auf gegebene Problemstellungen anwenden</li> <li>• Planungssysteme hinsichtlich ihrer Einsatzmöglichkeiten vergleichen</li> <li>• aktuelle Forschungsergebnisse im Bereich Action Planning analysieren und bewerten</li> </ul>		
<b>Sonstige Informationen:</b> Veranstaltung Deutsch (im Bedarfsfall English). Die Folien sowie weitere Materialien sind überwiegend in englischer Sprache.  Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: 22.5 h Vorlesung + 30 h Nachbereitung über 15 Wochen 22.5 h Übung + 75 h Bearbeitung von Übungsaufgaben über 15 Wochen 30 h Klausurvorbereitung		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Kenntnisse entsprechend den Modulen Gdl-Mfl-1 (Mathematik für Informatiker) und MI-AuD-B (Algorithmen und Datenstrukturen) oder des Moduls KogSys-KogInf-Psy.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Vorlesung Intelligente Agenten</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Ute Schmid <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich		2,00 SWS



<p><b>Inhalte:</b> In der Vorlesung werden wesentliche Konzepte und Methoden der kognitiv orientierten Künstlichen Intelligenz mit dem Fokus auf Problemlösen und Planen eingeführt. Wesentliche Themengebiete sind: STRIPS-Planung, Logik und Deduktives Planen, heuristische Suche und heuristisches Planen, Planning Graph Techniken, SAT-Planning, Multiagenten-Planung, Bezüge zum menschlichen Problemlösen und Planen. Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p>	
<p><b>Literatur:</b> Russell &amp; Norvig: Artificial Intelligence -- A Modern Approach; Ghallab, Nau, Traverso: Automated Planning; Wooldridge: An Introduction to Multiagent Systems; Schöning: Logik für Informatiker; Sterling, Shapiro: Prolog</p>	
<p><b>2. Übung Intelligente Agenten</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Michael Siebers <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich</p> <p><b>Inhalte:</b> Vertiefung von in der Vorlesung eingeführten Methoden und Techniken, zum Teil mit Programmieraufgaben in PROLOG. Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p> <p><b>Literatur:</b> siehe Vorlesung</p>	2,00 SWS
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> In der Klausur können 90 Punkte erzielt werden. Die Klausur ist bestanden, wenn mindestens 40 Prozent erreicht werden. Im Semester werden Übungsblätter ausgegeben für deren freiwillige Bearbeitung eine bzw. zwei Wochen zur Verfügung stehen. Die Lösung der Übungsblätter wird bewertet. Bei bestandener Klausur wird die Bewertung der Übungsblätter für die Berechnung der Note mit berücksichtigt. Eine 1.0 ist dabei auch ohne Punkte aus den Übungsblättern erreichbar. Erlaubte Hilfsmittel: Handschriftliche und gedruckte Materialien, Taschenrechner. Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p>	

<b>Modul KogSys-Proj-B Bachelor-Projekt Kognitive Systeme</b>		6 ECTS / 180 h
<i>Bachelor Project Kognitive Systeme</i>		
(seit WS09/10) Modulverantwortliche/r: Ute Schmid		
<b>Inhalte:</b> Aufbauend auf den in den Vorlesungen und Übungen des Faches Kognitive Systeme erworbenen Kenntnissen und Fertigkeiten wird eine wissenschaftliche Fragestellung in Kleingruppen bearbeitet. Dabei werden Kompetenzen des wissenschaftlichen Arbeitens im Forschungsgebiet Kognitive Systeme sowie Kompetenzen in der Teamarbeit erworben.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden können: <ul style="list-style-type: none"> <li>• konkrete Forschungsfragen in den Stand der Forschung einordnen</li> <li>• Forschungsfragen und Forschungsziele entwerfen und formulieren</li> <li>• Forschungsmethoden im Bereich Kognitive Systeme nennen und erläutern</li> <li>• Grundlegende Prinzipien der Bewertung und Evaluation von Forschungsergebnissen nennen und erläutern</li> <li>• eine Problemlösung bzw. Konzeption implementieren</li> <li>• eine empirische Studie nach Anleitung durchführen und auswerten</li> <li>• Algorithmen und Verfahren formal darstellen</li> <li>• eine wissenschaftliche Fragestellung im Team bearbeiten</li> <li>• Forschungsergebnisse mündlich wie schriftlich präsentieren</li> </ul>		
<b>Sonstige Informationen:</b> Veranstaltung Deutsch (im Bedarfsfall Englisch).  Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: 20 h persönliche Besprechungstermine mit dem Dozenten 30 h Erarbeitung der Literatur (inkl. Algorithmen, Systeme) 80 h Konkretisierung und Umsetzung der Projektaufgabe 10 h Vorbereitung der Abschluss-Präsentation 40 h Abfassen des Berichts  Hausarbeit und Kolloquium können in Deutsch oder Englisch abgefasst werden.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Kenntnisse entsprechend einer der folgenden Module: Modul Intelligente Agenten (KogSys-IA-B) - empfohlen Modul Grundlagen der Kognitiven Informatik (KogSys-KogInf-Psy) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>Projekt Kognitive Systeme</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Ute Schmid, Mitarbeiter Angewandte Informatik, insb. Kognitive Systeme  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> WS, SS</p>	4,00 SWS
<p><b>Inhalte:</b>  Im Projekt werden wechselnde Themen aus dem Bereich Kognitive Systeme, die in Zusammenhang mit aktuellen Forschungsarbeiten der Gruppe stehen, in Kleingruppen (2-3 Studierende) bearbeitet. Wissenschaftliches Arbeiten im Bereich Kognitive Systeme wird dabei exemplarisch eingeübt: Aufarbeitung der relevanten Literatur zur Verankerung des Themas gemäß des Standes der Forschung, Umsetzung in Form der Implementation eines Algorithmus, der Evaluation von Algorithmen oder Systemen anhand ausgewählter Probleme oder der empirischen Untersuchung einer kognitiven Fragestellung. Darstellung der Ergebnisse in Form einer wissenschaftlichen Publikation, Präsentation und Verteidigung der Arbeit in einem Kolloquium.</p>	
<p><b>Literatur:</b>  wird in der Veranstaltung bekanntgegeben</p>	
<p><b>Prüfung</b>  Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten  Bearbeitungsfrist: 6 Monate  <b>Zulassungsvoraussetzung zur Modulprüfung:</b>  Regelmäßige Teilnahme an der Lehrveranstaltung  <b>Beschreibung:</b>  Umsetzung der Projektaufgabe, Dokumentation in Form einer wissenschaftlichen Publikation als Hausarbeit.   Die Gewichtung von Hausarbeit und Kolloquium wird zu Beginn der Lehrveranstaltung von der Dozentin bzw. dem Dozenten bekannt gegeben.</p>	

<p><b>Modul KTR-Datkomm-B Datenkommunikation</b>  <i>Data communication</i></p>	<p>6 ECTS / 180 h          45 h Präsenzzeit          135 h Selbststudium</p>
<p>(seit WS10/11)          Modulverantwortliche/r: Prof. Dr. Udo Krieger</p>	
<p><b>Inhalte:</b>          Diese Lehrveranstaltung behandelt die technischen Grundlagen der öffentlichen, betrieblichen und privaten Rechnerkommunikation in lokalen Netzen und Weitverkehrsnetzen sowie grundlegende Aspekte ihres Dienstangebots. Es werden die geläufigsten Dienste-, Netz- und Protokollarchitekturen öffentlicher und privater Datenkommunikationsnetze wie das OSI-Referenzmodell bzw. die TCP/IP-Protokollfamilie mit aufgesetzten Dateitransfer, World Wide Web und Multimedia-Diensten vorgestellt.          Ferner werden die Grundprinzipien der eingesetzten Übertragungsverfahren, der Übertragungssicherungs- und Steuerungsalgorithmen und der wichtigsten Medienzugriffsverfahren diskutiert, z.B. geläufige Übertragungs- und Multiplextechniken wie FDMA, TDMA, CDMA, Medienzugriffstechniken der CSMA-Protokollfamilie inklusive ihrer Umsetzung in LANs nach IEEE802.x Standards, Sicherungsprotokolle der ARQ-Familie sowie Flusskontrollstrategien mit variablen Fenstertechniken und ihre Realisierung im HDLC-Protokoll.          Außerdem werden grundlegende Adressierungs- und Vermittlungsfunktionen in Rechnernetzen wie Paketvermittlung in Routern und Paketverkehrlenkung nach Kürzeste-Wege-Prinzipien bzw. Verkehrlenkung nach dem Prinzip virtueller Wege dargestellt. Darüber hinaus werden die Grundfunktionen der Transportschicht und ihre exemplarische Umsetzung in TCP erläutert.</p>	
<p><b>Lernziele/Kompetenzen:</b>          Die Studierenden sollen zu eigenständigem Arbeiten im Bereich moderner Kommunikationsnetze befähigt werden. Es werden Grundkenntnisse der Datenkommunikation und die systematische Analyse der verwendeten Algorithmen mit Hilfe eines interaktiven Konzeptes theoretischer und praktischer Übungsaufgaben vermittelt. Die Studierenden lernen, gegebene Implementierungen der vorgestellten Datenkommunikationsverfahren zu analysieren und durch Messungen im Kommunikationslabor ihr Leistungsverhalten zu überprüfen.</p>	
<p><b>Sonstige Informationen:</b>          Der Arbeitsaufwand gliedert sich grob wie folgt:</p> <ul style="list-style-type: none"> <li>• Teilnahme an Präsenzveranstaltungen (Vorlesung, Übung, Laborbesprechungen): 45 Stunden</li> <li>• Vor- und Nachbereitung von Vorlesungen und Übungen: 100 Stunden</li> <li>• Prüfungsvorbereitung: 35 Stunden</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>          keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b></p> <ul style="list-style-type: none"> <li>• erfolgreich abgeschlossene Prüfungen der Grundlagenfächer des Bachelorstudiums, insbesondere Einführung in Algorithmen, Programmierung und Software sowie grundlegende Kenntnisse effizienter Algorithmen</li> <li>• gute Programmierkenntnisse in JAVA (oder C++)</li> </ul> <p>Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen</p>	<p><b>Besondere Bestehensvoraussetzungen:</b>          keine</p>

Modul Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) - empfohlen		
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>1. Vorlesung Datenkommunikation</b></p> <p><b>Lehrformen:</b> Vorlesung</p> <p><b>Dozenten:</b> Prof. Dr. Udo Krieger</p> <p><b>Sprache:</b> Deutsch/Englisch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <p><b>Inhalte:</b> Diese Lehrveranstaltung behandelt die technischen Grundlagen der öffentlichen, betrieblichen und privaten Rechnerkommunikation in lokalen Netzen und Weitverkehrsnetzen sowie grundlegende Aspekte ihres Dienstangebots. Es werden die geläufigsten Dienste-, Netz- und Protokollarchitekturen öffentlicher und privater Datenkommunikationsnetze wie das OSI-Referenzmodell bzw. die TCP/IP-Protokollfamilie mit aufgesetzten Dateitransfer, World Wide Web und Multimedia-Diensten vorgestellt. Ferner werden die Grundprinzipien der eingesetzten Übertragungs-, Übertragungssicherungs- und Steuerungsalgorithmen und des Medienzugriffs diskutiert, z.B. geläufige Übertragungs- und Multiplextechniken wie FDMA, TDMA und CDMA Medienzugriffstechniken der CSMA-Protokollfamilie inklusive ihrer Umsetzung in LANs nach IEEE802.x Standards, Sicherungsprotokolle der ARQ-Familie sowie Flusskontrollstrategien mit variablen Fenstertechniken und ihre Realisierung. Außerdem werden grundlegende Adressierungs- und Vermittlungsfunktionen in Rechnernetzen wie Paketvermittlung in Routern und Paketverkehrslenkung dargestellt. Darüber hinaus werden die Grundfunktionen der Transportschicht und ihre exemplarische Umsetzung in TCP erläutert. Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.</p> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Lean-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2004</li> <li>• Tanenbaum, A. S.: Computernetzwerke, Pearson Studium, München, 4. Aufl., 2003</li> <li>• Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2014</li> <li>• Comer, D.: Computernetzwerke und Internets, Pearson Studium, München, 2004</li> </ul> <p>Weitere Angaben und Erläuterungen erfolgen in der 1. Vorlesung.</p>	2,00 SWS
<p><b>2. Übung Datenkommunikation</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Dr. Udo Krieger, Mitarbeiter Informatik, insbesondere Kommunikationsdienste, Telekommunikationssysteme und Rechnernetze</p>	2,00 SWS

<p><b>Sprache:</b> Deutsch/Englisch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>          Es werden Grundkenntnisse der Datenkommunikation und die systematische Analyse der dabei verwendeten Algorithmen mit Hilfe eines interaktiven Übungskonzeptes aus Haus- und Laboraufgaben vermittelt. Vorlesungsbegleitend werden diese Übungsaufgaben zu folgenden Themen bearbeitet:</p> <ul style="list-style-type: none"> <li>• Netzentwurfsprinzipien</li> <li>• OSI-Protokolle</li> <li>• TCP/IP-Protokollstapel</li> <li>• Netzelemente</li> <li>• Datenübertragungssicherungsschicht</li> <li>• Medienzugriffsschicht</li> </ul> <p>Die Studierenden lernen, gegebene Implementierungen der vorgestellten Datenkommunikationsverfahren mathematisch und kommunikationstechnisch zu analysieren, durch Messungen ihr Leistungsverhalten zu überprüfen und Vor- bzw. Nachteile der Lösungen zu bewerten.          Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.</p>	
<p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Lean-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2004</li> <li>• Tanenbaum, A. S.: Computernetzwerke, Pearson Studium, München, 4. Aufl., 2003</li> <li>• Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2014</li> <li>• Comer, D.: Computernetzwerke und Internets, Pearson Studium, München, 2004</li> </ul> <p>Weitere Literatur wird in der Übung benannt.</p>	
<p><b>Prüfung</b>          schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b>          Die Inhalte der Vorlesung sowie die Aufgabenstellungen, Lösungen und Erkenntnisse der Übung, die Haus- und Laboraufgaben beinhaltet, werden in Form einer Klausur geprüft.</p> <p>Im Verlauf des Semesters können durch die Bearbeitung der Laboraufgaben der Übung und die erfolgreiche Bewertung der entsprechenden Teilleistungen eine maximale Anzahl von Bonuspunkten erworben werden. Diese Bonuspunkte werden bei der Notenvergabe des Moduls berücksichtigt, wobei das Bestehen der Modulprüfung die Voraussetzung für die Berücksichtigung dieser individuell erbrachten Bonuspunkte ist. Die Berechnungs-, Vergabe- und Anrechnungsmodalitäten der Bonuspunktregelung werden zu Beginn der Lehrveranstaltung festgelegt und den Studierenden zur Kenntnis gebracht. Diese Bonuspunkte stellen eine freiwillige Zusatzleistung dar. Das Bestehen der</p>	

Modulprüfung ist grundsätzlich ohne diese Zusatzleistung möglich. Das Erreichen der Note 1.0 ist ebenfalls ohne die Erbringung dieser Zusatzleistung möglich.

Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstaltung.

Zulässige Hilfsmittel der Prüfung:

- Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay

<b>Modul KTR-Mfi-2 Mathematik für Informatik 2 (Lineare Algebra)</b> <i>Mathematics for Computer Science 2 (Linear Algebra)</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Udo Krieger		
<b>Inhalte:</b> Die Lehrveranstaltung stellt mathematische Grundlagen der Informatik bereit und ist dem Pflichtbereich der Modulgruppe A1 "Mathematische Grundlagen" für Angewandte Informatik/Software Systems Science zugeordnet. Der besondere Bezug zur Angewandten Informatik bzw. Software Systems Science wird in den Vorlesungsbeispielen und Übungen herausgearbeitet.  Es werden grundlegende Methoden und Algorithmen der Gruppen- und Ringtheorie, der linearen Algebra, der Matrizenalgebra, der Lösungstheorie linearer Gleichungssysteme, der Eigenwerttheorie sowie spezifische Anwendungen der Informatik vorgestellt.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden lernen, die grundlegende Methoden und Algorithmen der Lineare Algebra anzuwenden und spezifische Anwendungen der Angewandten Informatik als Probleme der linearen Algebra zu erkennen, zu formulieren und mit Hilfe geeigneter Verfahren zu lösen.		
<b>Sonstige Informationen:</b> Das Modul stellt die Grundlagen für Studierende der Angewandten Informatik und Software Systems Science sowie Studierende im Nebenfach verwandter Bachelorstudiengänge der Fakultät WIAI bereit.  Der Arbeitsaufwand gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Präsenzveranstaltungen (Vorlesung, Übung, Laborbesprechungen): 45 Stunden</li> <li>• Vor- und Nachbereitung von Vorlesungen und Übungen: 100 Stunden</li> <li>• Prüfungsvorbereitung: 35 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Schulkenntnisse auf dem Niveau eines Mathematik-Vorkurses Modul Mathematik-Vorkurs-Bachelorstudium (KTR-MVK-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> 1. - 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Mathematik für Informatik 2 (Lineare Algebra)</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Udo Krieger <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	4,00 SWS
<b>Inhalte:</b> Die Lehrveranstaltung stellt mathematische Grundlagen der Informatik bereit und ist dem Pflichtbereich der Modulgruppe A1 "Mathematische Grundlage"	



<p>für Angewandte Informatik/Software Systems Science zugeordnet. Es werden grundlegende Methoden und Algorithmen der Gruppen- und Ringtheorie, der linearen Algebra, der Matrizenalgebra, der Lösungstheorie linearer Gleichungssysteme, der Eigenwerttheorie sowie spezifische Anwendungen der Informatik vorgestellt.</p>	
<p><b>Literatur:</b></p> <ul style="list-style-type: none"><li>• A. Steger: Diskrete Strukturen 1, Springer, Heidelberg, 2002.</li><li>• G. Golub, C.F. van Loan: Matrix Computations, 3ed., Johns Hopkins, 1996.</li><li>• D. Hachenberger: Mathematik für Informatiker, Pearson, 2008.</li><li>• J. Liesen, V. Mehrmann: Lineare Algebra, Springer, Bachelorkurs Mathematik, 2. Auflage, 2015.</li><li>• B. Pareigis: Linear Algebra für Informatiker, Springer, 2000.</li><li>• M.P.H. Wolff u.a.: Mathematik für Informatik und Bioinformatik, Springer, 2004.</li><li>• Weitere Literatur wird in der Vorlesung benannt.</li></ul>	
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> Schriftliche Prüfung zu Inhalten der Vorlesung und Übungen im Umfang von 90 Minuten. Zugelassene Hilfsmittel werden in der Lehrveranstaltung bekannt gegeben.</p>	

<p><b>Modul KTR-Proj Projekt Kommunikationsnetze und -dienste</b>  <i>Project Communication Networks and Services</i></p>	<p>6 ECTS / 180 h          40 h Präsenzzeit          140 h Selbststudium</p>	
<p>(seit WS10/11)          Modulverantwortliche/r: Prof. Dr. Udo Krieger</p>		
<p><b>Inhalte:</b>          Die Lehrveranstaltung vermittelt Einblicke in die Entwicklung neuer Dienstarchitekturen und Netztechnologien aus dem Bereich des Internets der nächsten Generation. Im Mittelpunkt steht die eigenständige, teamorientierte praktische Umsetzung eines Entwicklungsauftrages unter Verwendung des erworbenen Wissens einzelner Lehrveranstaltungen des Fachgebietes der Professur für Informatik.          Die Betriebssystem-Grundausrüstung und erforderliche Software-Werkzeuge wie Vyatta-Router, Wireshark, Atheros, RapidStream und andere werden bereitgestellt. Grundlagen der Handhabung werden von den Studierenden im Projekt selbst erarbeitet. Die Organisation der Arbeiten erfolgt in einem industrienahen Projektrahmen aus Definitions-, Vorbereitungs-, Implementierungs- und Präsentationsphasen. Details zu den einzelnen Entwicklungsaufgaben, ihren Zielen und Methoden werden in der Vorbesprechung genannt.</p>		
<p><b>Lernziele/Kompetenzen:</b>          Wichtige Fertigkeiten bei der Anwendung neuer Kommunikationstechnologien und zur Entwicklung neuer Kommunikationsdienste sind nur durch die Vermittlung praktischer Fähigkeiten und Erfahrungen in teamorientierten Prozessen unter Zeit- und Zielvorgaben industrienah erlernbar. Die Studierenden werden in der Lehrveranstaltung in einem angeleiteten, aber ansonsten eigenverantwortlich durchgeführten, teamorientierten Arbeitsprozess aktuelle Entwicklungsaufgaben aus dem Forschungsbereich der Professur für Informatik bearbeiten.          Ziel ist der Erwerb praktischer Fertigkeiten auf dem Gebiet der IP-gestützten, qualitätsgesicherten Multimediakommunikation und die Fähigkeit, Lösungsvorschläge moderner Dienstarchitekturen im Internet der Zukunft konzipieren, implementieren und sicher beurteilen zu können.</p>		
<p><b>Sonstige Informationen:</b>          Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt:</p> <ul style="list-style-type: none"> <li>• Teilnahme an Gruppen- und Einzelbesprechungen: 40 Stunden</li> <li>• Bearbeiten der Projektaufgabe: 120 Stunden</li> <li>• Kolloquiumsvorbereitung: 20 Stunden</li> </ul> <p>The module can be selected by Erasmus or exchange students and master students speaking only English.</p>		
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>          keine</p>		
<p><b>Empfohlene Vorkenntnisse:</b></p> <ul style="list-style-type: none"> <li>• Kenntnisse der Datenkommunikation im Umfang von KTR-Datkomm-B</li> <li>• solide Kenntnisse in JAVA (oder C++)</li> </ul> <p>Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen</p>	<p><b>Besondere Bestehensvoraussetzungen:</b>          keine</p>	
<p><b>Angebotshäufigkeit:</b> WS, jährlich</p>	<p><b>Empfohlenes Fachsemester:</b></p>	<p><b>Minimale Dauer des Moduls:</b></p>

		1 Semester
<b>Lehrveranstaltungen</b>		
<b>Projekt Kommunikationsnetze und-dienste</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Udo Krieger <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich		4,00 SWS
<b>Inhalte:</b> Die Lehrveranstaltung vermittelt Einblicke in die Entwicklung neuer Dienstarchitekturen und Netztechnologien aus dem Bereich des Internets der nächsten Generation. Im Mittelpunkt steht die eigenständige, team-orientierte praktische Umsetzung eines Entwicklungsauftrages unter Verwendung des erworbenen Wissens einzelner Lehrveranstaltungen des Fachgebietes der Professur für Informatik. Die Betriebssystem-Grundausstattung und erforderliche Software-Werkzeuge wie Vyatta-Router, Wireshark, Atheris und RapidStream werden bereitgestellt. Grundlagen der Handhabung werden von den Studierenden im Projekt selbst erarbeitet. Die Organisation der Arbeiten erfolgt in einem industrienahen Projektrahmen aus Definitions-, Vorbereitungs-, Implementierungs- und Präsentationsphasen. Dabei soll wie in realen Projekten üblich eine inkrementelle Vorgehensweise durchgeführt werden, d.h: <ul style="list-style-type: none"> <li>• Unterteilung der Arbeiten in Arbeitspakete (laboratories/work packages),</li> <li>• ihre Untergliederung in Aufgaben (tasks) und Teilaufgaben (subtasks) mit Meilensteinen</li> <li>• und der Darlegung von Zwischenergebnissen bzw.</li> <li>• einem Abschlussbericht mit Abschlusspräsentation</li> </ul> Es werden Entwicklungsaufgaben zu aktuellen Forschungsfragen im "Future Generation Internet" bearbeitet. Details werden auf der Webseite der Lehrveranstaltung angekündigt. Eine aktuelle Liste der bearbeiteten Themen der Lehrveranstaltung wird in der Vorlesung bereitgestellt. Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.		
<b>Literatur:</b> Die aktuelle Literatur wird auf der Webseite der Lehrveranstaltung bekanntgegeben.		
<b>Prüfung</b> Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 4 Monate <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung <b>Beschreibung:</b>		

Die Bewertung der Prüfungsleistung erfolgt nach Abschluss der Lehrveranstaltung auf folgender Grundlage:

- Auswertung des von einem Studierenden individuell erstellten schriftlichen Berichts der bearbeiteten Aufgaben im Rahmen einer Einzelarbeit oder der von einem Studierenden im Rahmen einer Gruppenarbeit individuell bearbeiteten Aufgaben, die im schriftlichen Bericht der bearbeiteten Aufgaben unter Verwendung einer eindeutigen Kennzeichnung der Urheberschaft dokumentiert werden (mit Bearbeitungsdauer von 4 Monaten)
- Vorführung und Erläuterungen der Zusammenhänge einzelner Aufgaben und Ergebnisse im Rahmen einer individuellen Kolloquiumsprüfung im Umfang von 30 Minuten

Die individuelle Gesamtleistung muss mit der Note "ausreichend" bewertet werden, um die Prüfung zu bestehen.

Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstaltung.

<b>Modul Mathe-B-01 (BWL) Mathematik für Wirtschaftswissenschaftlerinnen und Wirtschaftswissenschaftler I (BWL)</b>	3 ECTS / 90 h
(seit WS16/17) Modulverantwortliche/r: N.N.	
<p><b>Inhalte:</b></p> <ul style="list-style-type: none"> <li>0 Grundlagen</li> <li>0.1 Kartesische Produkte und Relationen</li> <li>0.2 Abbildungen</li> <li>1 Folgen und Reihen <ul style="list-style-type: none"> <li>1.1 Folgen</li> <li>1.2 Reihen</li> <li>1.3 Finanzmathematik <ul style="list-style-type: none"> <li>1.3.1 Einfache Zinsrechnung</li> <li>1.3.2 Zinseszinsrechnung</li> <li>1.3.3 Rentenrechnung</li> <li>1.3.4 Tilgungsrechnung</li> <li>1.3.5 Investitionsrechnung</li> </ul> </li> </ul> </li> <li>2 Differenzialrechnung <ul style="list-style-type: none"> <li>2.1 Funktionen einer und mehrerer Variablen <ul style="list-style-type: none"> <li>2.1.1 Beispiele, grafische Darstellung und Eigenschaften von Funktionen einer und mehrerer Variablen</li> <li>2.1.2 Polynome, gebrochen rationale und algebraische Funktionen</li> <li>2.1.3 Transzendente Funktionen (Exponential-, Logarithmus- und Winkelfunktionen)</li> <li>2.1.4 Grenzwerte und Stetigkeit von Funktionen</li> </ul> </li> <li>2.2 Differenzialrechnung für Funktionen einer Variablen <ul style="list-style-type: none"> <li>2.2.1 Differenzialquotient und Ableitungsregeln</li> <li>2.2.2 Differenziation der Grundfunktionen</li> <li>2.2.3 Monotonie, Konvexität/Konkavität und Extremstellen differenzierbarer Funktionen einer Variablen</li> <li>2.2.4 Rechnen mit dem Symbol #, die Regeln von de l'Hospital</li> <li>2.2.5 Approximation differenzierbarer Funktionen durch Polynome, Differenziale und der Satz von Taylor</li> <li>2.2.6 Elastizitäten</li> </ul> </li> <li>2.3 Differenzialrechnung für Funktionen mehrerer Variablen <ul style="list-style-type: none"> <li>2.3.1 Partielle und totale Ableitungen</li> <li>2.3.2 Die Kettenregel für Funktionen mehrerer Variablen</li> <li>2.3.3 Partielle Ableitungen höherer Ordnung</li> <li>2.3.4 Partielle und totale Differenziale, partielle Elastizitäten</li> <li>2.3.5 Implizite Funktionen</li> <li>2.3.6 Extremstellen differenzierbarer Funktionen mehrerer Variablen (ohne Nebenbedingungen)</li> <li>2.3.7 Extremstellen differenzierbarer Funktionen mehrerer Variablen (mit Nebenbedingungen)</li> <li>2.3.8 Differenziation vektorwertiger Funktionen</li> </ul> </li> </ul> </li> <li>3 Integralrechnung <ul style="list-style-type: none"> <li>3.1 Das unbestimmte Integrale</li> <li>3.2 Das bestimmte Integrale</li> <li>3.3 Uneigentliche Integrale</li> <li>3.4 Integralrechnung für Funktionen mehrerer Variablen</li> </ul> </li> </ul>	

<b>Lernziele/Kompetenzen:</b> Vermittlung von mathematischen Grundkenntnissen aus dem Gebiet der Analysis. Die Teilnehmer dieser Vorlesung/Übung sollen in die Lage versetzt werden, die mathematischen Verfahren und Konzepte der weiterführenden (wirtschafts-)informatischen Veranstaltungen zu verstehen und zu beherrschen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Mathematik für Wirtschaftswissenschaftler I</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Dr. rer. pol. Reinhard Dobbener <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS	3,00 SWS
<b>Literatur:</b> <ul style="list-style-type: none"> <li>• Chiang A.C.: Fundamental Methods of Mathematical Economics, McGraw-Hill, New York, 1967.</li> <li>• Dobbener R.: Analysis - Studienbuch für Ökonomen, 2. Auflage, Oldenbourg, München, Wien, 1993.</li> <li>• Gal T., Kruse H.J., Vogeler B., Wolf H.: Mathematik für Wirtschaftswissenschaftler, Band 1-3, Springer, Berlin, Heidelberg, New York, Tokio, 1983.</li> <li>• Opitz O.: Mathematik, Oldenbourg, München, Wien, 1989.</li> <li>• Schwarze J.: Mathematik für Wirtschaftswissenschaftler, Band 1-3, Neue Wirtschaftsbriefe, Herne, Berlin, 1981.</li> <li>• Sydsaeter K., Hammond P.: Mathematik für Wirtschaftswissenschaftler, Pearson Studium, München 2004.</li> </ul>	

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Zulassungsvoraussetzung zur Modulprüfung:</b> keine	
---	--

<b>Modul MI-AuD-B Algorithmen und Datenstrukturen</b> <i>Algorithms and Data Structures</i>		6 ECTS / 180 h
(seit SS07) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Grundlegende Algorithmen (z. B. Suchen, Sortieren, einfache Graphalgorithmen) und Datenstrukturen (z. B. Listen, Hashtabellen, Bäume, Graphen) werden vorgestellt. Konzepte der Korrektheit, Komplexität und Algorithmenkonstruktion werden betrachtet.		
<b>Lernziele/Kompetenzen:</b> Das Modul vermittelt die Kompetenz, die Qualität von Datenstrukturen und Algorithmen im Hinblick auf konkrete Anforderungen einzuschätzen und ihre Implementierung in einem Programm umzusetzen. Daneben sollen grundlegende Kompetenzen im Bereich der Algorithmenkonstruktion erworben werden. Durch die Übung soll auch Sicherheit im Umgang mit objektorientierten Entwicklungsmethoden und Standardbibliotheken erworben und Teamarbeit geübt werden.		
<b>Sonstige Informationen:</b> Der <b>Arbeitsaufwand</b> für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung)</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Semesterbegleitendes Üben, Bearbeiten alter Klausuraufgaben, ... zum Vorlesungsstoff: ca. 30 Stunden (inkl. 7,5 Stunden [= 1/3] der 2 SWS Übungsbetrieb)</li> <li>• Bearbeiten der 6 Teilleistungen: insgesamt ca. 60 Stunden (inkl. 15 Stunden [= 2/3] der 2 SWS Übungsbetrieb)</li> <li>• Klausurvorbereitung und Klausur: ca. 37,5 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Kenntnisse in Informatik und Programmierung, wie sie z. B. im Modul DSG-EiAPS-B vermittelt werden.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Algorithmen und Datenstrukturen</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Andreas Henrich <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	2,00 SWS
<b>Inhalte:</b> Die Vorlesung betrachtet die klassischen Bereiche des Themengebiets Algorithmen und Datenstrukturen:	

<ul style="list-style-type: none"> <li>• Einleitung</li> <li>• Listen</li> <li>• Hashverfahren</li> <li>• Bäume</li> <li>• Graphen</li> <li>• Sortieren</li> <li>• Algorithmenkonstruktion</li> </ul>	
<p><b>Literatur:</b>          Als begleitende Lektüre wird ein Standardlehrbuch über Algorithmen und Datenstrukturen empfohlen. Beispiele wären:</p> <ul style="list-style-type: none"> <li>• Saake, Gunter; Sattler, Kai-Uwe: Algorithmen und Datenstrukturen: Eine Einführung mit Java, ISBN: 978-3-89864-385-6, 3. Aufl. 2006, 512 Seiten, Dpunkt Verlag</li> <li>• Ottmann, Thomas; Widmayer, Peter: Algorithmen und Datenstrukturen, ISBN: 978-3-8274-1029-0, 4. Aufl. 2002, 736 Seiten, Spektrum, Akedemischer Verlag</li> </ul>	
<p><b>2. Übung Algorithmen und Datenstrukturen</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Mitarbeiter Medieninformatik  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p>	<p>2,00 SWS</p>
<p><b>Inhalte:</b>          In der Übung werden folgende Aspekte betrachtet:</p> <ul style="list-style-type: none"> <li>• Verständnis und Nutzung von Algorithmen</li> <li>• Aufwandsbestimmung für Algorithmen</li> <li>• Implementierung von Algorithmen und Datenstrukturen</li> <li>• Nutzung von Bibliotheken</li> <li>• Anwendung von Prinzipien zur Algorithmenkonstruktion</li> </ul>	
<p><b>Literatur:</b>          siehe Vorlesung</p>	

<p><b>Prüfung</b>          schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten  <b>Beschreibung:</b>  <b>Gegenstand</b> der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich der Teilleistungen; siehe unten).          In der <b>Klausur</b> können 90 Punkte erzielt werden.          Zusätzlich zur Prüfungsdauer wird eine <b>Lesezeit</b> von 15 Minuten gewährt, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.          Im Semester werden studienbegleitend 6 <b>Teilleistungen</b> (schriftliche Hausarbeiten) in der Übung ausgegeben und besprochen, deren Abgabe freiwillig ist. Für jede Teilleistung stehen in der Regel 2 Wochen als Bearbeitungszeit</p>	
---	--



<p>zur Verfügung. Die abgegebenen Lösungen zu den Teilleistungen werden bewertet. Pro Teilleistung können maximal 2 Punkte erzielt werden. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die bei der Bearbeitung der Teilleistungen erreichten Punkte (maximal 12 Punkte) als Bonuspunkte angerechnet. Eine 1,0 ist dabei auch ohne Punkte aus der Bearbeitung der Teilleistungen erreichbar.</p>	
--	--

<b>Modul MI-EMI-B Einführung in die Medieninformatik</b> <i>Introduction to Media Informatics</i>		6 ECTS / 180 h
(seit WS11/12) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Neben Grundkonzepten der Digitalisierung werden die Medientypen Bild, Audio, Text, Video, 2D-Vektorgrafik sowie 3D-Grafik behandelt. Dabei wird jeweils auf die Erstellung und Bearbeitung entsprechender Medienobjekte sowie deren Kodierung eingegangen.		
<b>Lernziele/Kompetenzen:</b> Studierende sollen zu den verschiedenen Medientypen Beispielformate kennenlernen. Sie sollen die eingesetzten Kompressionsverfahren sowie die dahinter stehenden Philosophien verstehen und die praktischen Einsatzmöglichkeiten einschätzen können. Ferner sollen sie konzeptuelle Kenntnisse und praktische Erfahrungen im Umgang mit Medienobjekten sammeln und z. B. die Erstellung und Bearbeitung von Medientypen wie Text, Bild, Audio und Video selbständig durchführen können.		
<b>Sonstige Informationen:</b> Der <b>Arbeitsaufwand</b> für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung)</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Semesterbegleitendes Üben, Bearbeiten alter Klausuraufgaben, ... zum Vorlesungsstoff: ca. 30 Stunden (inkl. 7,5 Stunden [= 1/3] der 2 SWS Übungsbetrieb)</li> <li>• Bearbeiten der 3 Teilleistungen: insgesamt ca. 60 Stunden (inkl. 15 Stunden [= 2/3] der 2 SWS Übungsbetrieb)</li> <li>• Prüfungsvorbereitung und Prüfung: ca. 37,5 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse in Informatik (können auch durch den parallelen Besuch eines einführenden Moduls zur Informatik erworben werden)		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Einführung in die Medieninformatik</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Andreas Henrich <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	2,00 SWS
<b>Inhalte:</b> Im Rahmen dieser Vorlesung werden nach einer Einführung in das Thema grundlegende Medien und Medienformate betrachtet. Hierzu zählen Bilder, Audio, Texte und Typografie, Video, 2D- und 3D-Grafik.	

<p>Neben den Formaten werden die entsprechenden Grundlagen wie Farbmodelle und Wahrnehmungsmodelle betrachtet. Ziel ist dabei, praktische Fähigkeiten im Umgang mit den genannten Formaten zu vermitteln und die Konzepte von Kodierungs- und Kompressionsverfahren zu erarbeiten. Hierzu geht die Veranstaltung, die einen breiten Überblick über das Gebiet geben soll, an einzelnen ausgewählten Stellen stärker in die Tiefe. Zu nennen sind dabei insbesondere die Medientypen Text, Bild, Audio, Video und 2D-Vektorgrafik.</p> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Malaka, Rainer; Butz, Andreas; Hussmann, Heinrich: Medieninformatik: Eine Einführung. Pearson Studium; 1. Auflage, 2009</li> <li>• Chapman, Nigel; Chapman Jenny: Digital Multimedia (2nd Edition), John Wiley &amp; Sons, Ltd, 2004</li> <li>• Henning, Peter A.: Taschenbuch Multimedia , 3. Auflage, Fachbuchverlag Leipzig im Carl Hanser Verlag, 2003</li> <li>• weitere Literatur wird in der Veranstaltung bekannt gegeben</li> </ul>	
<p><b>2. Übung Einführung in die Medieninformatik</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Mitarbeiter Medieninformatik</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <p>Die Inhalte der Vorlesung Einführung in die Medieninformatik werden in den Übungen vertieft und praktisch umgesetzt. Insbesondere werden Kodierungs- und Kompressionsverfahren nachvollzogen, Medienobjekte erstellt und bearbeitet und der Umgang mit einfachen Werkzeugen (z. B. zur Bildbearbeitung) eingeübt.</p> <hr/> <p><b>Literatur:</b></p> <p>siehe Vorlesung</p>	2,00 SWS
<p><b>Prüfung</b></p> <p>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b></p> <p><b>Gegenstand</b> der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich der Teilleistungen; siehe unten).</p> <p>In der <b>Klausur</b> können 90 Punkte erzielt werden.</p> <p>Zusätzlich zur Prüfungsdauer wird eine <b>Lesezeit</b> von 15 Minuten gewährt, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.</p> <p>Im Semester werden studienbegleitend 3 <b>Teilleistungen</b> (schriftliche Hausarbeiten) in der Übung ausgegeben und besprochen, deren Abgabe freiwillig ist. Für jede Teilleistung stehen in der Regel 4 Wochen als Bearbeitungszeit zur Verfügung. Die abgegebenen Lösungen zu den Teilleistungen werden bewertet. Pro Teilleistung können maximal 4 Punkte erzielt werden. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die bei der Bearbeitung der Teilleistungen erreichten Punkte (maximal 12</p>	

---

Punkte) als Bonuspunkte angerechnet. Eine 1,0 ist dabei auch ohne Punkte aus der Bearbeitung der Teilleistungen erreichbar.	
---	--

<b>Modul MI-Proj-B Projekt zur Medieninformatik [Bachelor]</b>		6 ECTS / 180 h
<i>Media Informatics Project [Bachelor]</i>		
(seit WS16/17)		
Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b>		
Aufbauend auf den in den Vorlesungen und Übungen des Faches Medieninformatik erworbenen Kenntnissen und Fertigkeiten wird in diesem Projekt für ein Anwendungsszenario ein System konzipiert und implementiert. Die Arbeit erfolgt im Team. Die Themen werden den Bereichen Web-Anwendungen bzw. Multimediale Systeme entnommen.		
<b>Lernziele/Kompetenzen:</b>		
Die Kompetenz zur systematischen Entwicklung von Systemen in einem arbeitsteiligen Team wird vertieft. Kompetenzen in den Bereichen Anforderungsermittlung, Systemdesign, Implementierung, Evaluation und Dokumentation werden vermittelt. Ferner werden durch die Arbeit im Team Kompetenzen im Bereich Teamfähigkeit gestärkt.		
<b>Sonstige Informationen:</b>		
Die Lehrveranstaltung wird in <b>Deutsch</b> durchgeführt. Zahlreiche Quellen und Dokumentationen sind aber auf <b>Englisch</b> verfasst.		
Der <b>Arbeitsaufwand</b> für dieses Modul gliedert sich in folgende Bereiche:		
<ul style="list-style-type: none"> <li>• Teilnahme an einführenden Präsenzveranstaltungen</li> <li>• Teilnahme an (Gruppen-)Besprechungen und Zwischenpräsentationen</li> <li>• Bearbeitung der Projektaufgabenstellung allein und im Team</li> <li>• Vor- und Nachbereitung von Projektbesprechungen und -präsentationen</li> <li>• Prüfungsvorbereitung und Prüfung</li> </ul>		
Die Aufwände können dabei in Abhängigkeit von der Aufgabenstellung und der in der Gruppe abgestimmten Aufgabenverteilung unter den Gruppenmitgliedern sehr unterschiedlich auf die Bereiche verteilt sein.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
keine		keine
Modul Einführung in die Medieninformatik (MI-EMI-B) - empfohlen		
Modul Web-Technologien (MI-WebT-B) - empfohlen		
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester
<b>Lehrveranstaltungen</b>		
<b>Projekt zur Medieninformatik</b>		4,00 SWS
<b>Lehrformen:</b> Übung		
<b>Dozenten:</b> Prof. Dr. Andreas Henrich, Mitarbeiter Medieninformatik		
<b>Sprache:</b> Deutsch		
<b>Angebotshäufigkeit:</b> WS, jährlich		

<p><b>Inhalte:</b> Im Projekt werden wechselnde Projektthemen zu den Inhalten der Lehrveranstaltungen im Bereich der Medieninformatik bearbeitet. Dabei sind im Regelfall Aspekte mehrerer Lehrveranstaltungen relevant, so dass sich Teams mit Studierenden, die unterschiedliche Lehrveranstaltungen besucht haben, gut ergänzen. Die in einem Projektpraktikum bearbeitete Aufgabenstellung geht dabei deutlich über den Umfang einer normalen Übungsaufgabe hinaus und wird in kleinen Gruppen bearbeitet. Das erarbeitete Ergebnis wird dokumentiert und in einer Abschlusspräsentation vorgestellt.</p> <hr/> <p><b>Literatur:</b> wird in der Veranstaltung bekannt gegeben</p>	
--	--

<p><b>Prüfung</b> Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten Bearbeitungsfrist: 6 Monate</p> <p><b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung</p> <p><b>Beschreibung:</b> Hausarbeit (Dokumentation und Reflexion des Projektes und des Projektverlaufes) sowie ca. 20 Min. Kolloquium zum Projektergebnis und zum Projektverlauf (in der Regel im Rahmen eines Gruppenkolloquiums)</p>	
--	--

<b>Modul MI-WebT-B Web-Technologien</b> <i>Web Technologies</i>		6 ECTS / 180 h
(seit SS12) Modulverantwortliche/r: Prof. Dr. Andreas Henrich		
<b>Inhalte:</b> Nach einer Betrachtung der Grundlagen werden die verschiedenen Ebenen der Entwicklung von Web-Anwendungen von HTML und CSS über JavaScript und entsprechende Bibliotheken bis hin zur Serverseite und Frameworks oder Content Management Systemen betrachtet. Aspekte der Sicherheit von Web-Anwendungen werden ebenfalls angesprochen.		
<b>Lernziele/Kompetenzen:</b> Studierende sollen methodische, konzeptuelle und praktische Fähigkeiten und Fertigkeiten zur Erstellung von Web-Applikationen erwerben. Besonderes Augenmerk wird dabei auf Web 2.0 Technologien gelegt. Die Studierenden sind nach Abschluss des Moduls in der Lage, Web-Anwendungen selbständig mit gängigen Frameworks und Techniken zu entwickeln.		
<b>Sonstige Informationen:</b> Die Lehrveranstaltungen werden in <b>Deutsch</b> durchgeführt. Zahlreiche Quellen und Dokumentationen der Systeme sind aber auf <b>Englisch</b> . Der <b>Arbeitsaufwand</b> für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Vorlesung: 22,5 Stunden (entspricht den 2 SWS Vorlesung)</li> <li>• Vor- und Nachbereitung der Vorlesung (inkl. Recherche und Studium zusätzlicher Quellen): ca. 30 Stunden</li> <li>• Semesterbegleitendes Üben, Bearbeiten alter Klausuraufgaben, ... zum Vorlesungsstoff: ca. 30 Stunden (inkl. 7,5 Stunden [= 1/3] der 2 SWS Übungsbetrieb)</li> <li>• Bearbeiten der 3 Teilleistungen: insgesamt ca. 60 Stunden (inkl. 15 Stunden [= 2/3] der 2 SWS Übungsbetrieb)</li> <li>• Prüfungsvorbereitung und Prüfung: ca. 37,5 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundkenntnisse der Informatik und zu Dateiformaten, wie Sie z. B. in den unten angegebenen Modulen erworben werden können. Insbesondere sind auch Kenntnisse in einer imperativen oder objektorientierten Programmiersprache erforderlich. Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen Modul Einführung in die Medieninformatik (MI-EMI-B) - empfohlen		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
1. Vorlesung Web-Technologien		

<p><b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Dr. Andreas Henrich  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>                  Die Veranstaltung betrachtet die Aufgabenfelder, Konzepte und Technologien zur Entwicklung von Web-Anwendungen. Folgende Bereiche bilden dabei die Schwerpunkte der Veranstaltung:</p> <ul style="list-style-type: none"> <li>• Das Web: Einführung, Architektur, Protokoll ...</li> <li>• Sprachen zur Beschreibung von Webseiten: HTML &amp; CSS</li> <li>• Client-Side Scripting: die Basics &amp; AJAX</li> <li>• Server-Side Scripting: CGI + PHP</li> <li>• Frameworks</li> <li>• Sicherheit von Web-Anwendungen</li> <li>• CMS, LMS, SEO &amp; Co.</li> </ul> <hr/> <p><b>Literatur:</b>                  aktuelle Literatur wird in der Veranstaltung bekannt gegeben</p>	2,00 SWS
<p><b>2. Übung Web-Technologien</b></p> <p><b>Lehrformen:</b> Übung  <b>Dozenten:</b> Mitarbeiter Medieninformatik  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>                  praktische Aufgaben zum Stoff der Vorlesung</p> <hr/> <p><b>Literatur:</b>                  siehe Vorlesung</p>	2,00 SWS

<p><b>Prüfung</b>                  schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b>  <b>Gegenstand</b> der Klausur sind alle Inhalte von Vorlesung und Übung (einschließlich der Teilleistungen; siehe unten).</p> <p>In der <b>Klausur</b> können 90 Punkte erzielt werden.</p> <p>Zusätzlich zur Prüfungsdauer wird eine <b>Lesezeit</b> von 15 Minuten gewährt, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können.</p> <p>Im Semester werden studienbegleitend 3 <b>Teilleistungen</b> (schriftliche Hausarbeiten) in der Übung ausgegeben und besprochen, deren Abgabe freiwillig ist. Für jede Teilleistung stehen in der Regel 4 Wochen als Bearbeitungszeit zur Verfügung. Die abgegebenen Lösungen zu den Teilleistungen werden bewertet. Pro Teilleistung können maximal 4 Punkte erzielt werden. Ist die Klausur bestanden (in der Regel sind hierzu 50 % der Punkte erforderlich), so werden die bei der Bearbeitung der Teilleistungen erreichten Punkte (maximal 12</p>	
---	--



---

Punkte) als Bonuspunkte angerechnet. Eine 1,0 ist dabei auch ohne Punkte aus der Bearbeitung der Teilleistungen erreichbar.	
---	--

<b>Modul MOBI-IMP-B Implementation of Data Management Systems</b>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
<i>Implementation of Data Management Systems</i>		
(seit WS14/15) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
<b>Inhalte:</b> This modul covers the realization of database systems: <ul style="list-style-type: none"> <li>· Architecture of Database Systems</li> <li>· Storage Structures, Files and Segments</li> <li>· DB Buffer Management</li> <li>· Various indexes and access path methods (hash-based, hierarchical, multi-dimensional)</li> <li>· DB Interfaces (record-oriented, set-oriented)</li> <li>· Implementation of table operations</li> </ul>		
<b>Lernziele/Kompetenzen:</b> The students will know how a database system can retrieve large amounts of structured data in low latency, and they will be able to understand and apply various indexing strategies for other data management related tasks.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Grundlegende Kenntnisse über relationale Datenbanken, relationale Algebra und SQL, z.B. erworben im Modul SEDA-DMS-B: Datenmanagementsysteme		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Vorlesung Implementation of Data Management Systems</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, jährlich		2,00 SWS
<b>Inhalte:</b> The lecture covers the realization of database systems: <ul style="list-style-type: none"> <li>· Architecture of Database Systems</li> <li>· Storage Structures, Files and Segments</li> <li>· DB Buffer Management</li> <li>· Various indexes and access path methods (hash-based, hierarchical, multi-dimensional)</li> <li>· DB Interfaces (record-oriented, set-oriented)</li> </ul>		

· Implementation of table operations	
<b>2. Übung Implementation of Data Management Systems</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	2,00 SWS
<b>Inhalte:</b> Siehe Vorlesung	
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b> Die Prüfungssprache ist Englisch.	

<b>Modul MOBI-MSS-B Mobility in Software Systems</b> <i>Mobility in Software Systems</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS15/16) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
<b>Inhalte:</b> This modul covers architectures, implementation techniques and algorithms for mobile software systems and software systems that manage mobility. This includes client-side aspects (mobile applications like location-based services), server-side aspects (data management of moving objects), and aspects of distribution (data communication). In addition, since many mobile software systems deal with sensitive information like the location of users, aspects of location privacy are covered.		
<b>Lernziele/Kompetenzen:</b> The students will understand the challenges of mobility in software systems, and will be able to apply techniques and methods to realize such systems.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge about relational databases, relational algebras and SQL (e.g. from module SEDA-DMS-B: Datenmanagementsysteme)		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Mobility in Software Systems</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	4,00 SWS
<b>Lernziele:</b> The students will understand the challenges of mobility in software systems, and will be able to apply techniques and methods to realize such systems.	
<b>Inhalte:</b> This lecture covers architectures, implementation techniques and algorithms for mobile software systems and software systems that manage mobility. This includes client-side aspects (mobile applications like location-based services), server-side aspects (data management of moving objects), and aspects of distribution (data communication). In addition, since many mobile software systems deal with sensitive information like the location of users, aspects of location privacy are covered.	

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b>	
--	--

---

Die Prüfungssprache ist Englisch.	
-----------------------------------	--

<b>Modul MOBI-PRAI-B Bachelor Project Mobile Software Systems (AI)</b>		6 ECTS / 180 h
<i>Bachelor Project Mobile Software Systems (AI)</i>		
(seit WS14/15)		
Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
<b>Inhalte:</b>		
<p>Applications of in mobile software systems, which are taken from current research activities in mobile, context-aware systems and data stream management, are carried out in part individually and in part in small teams of students, from conception, via theoretical and/or practical realization, to evaluation. In particular, the project concerns the development of sound concepts pertaining to the task to be addressed under the given project constraints. This requires studying the current research literature and relevant approaches on the project's topic.</p> <p>An example of a project task would be the conceptual development, the prototypic implementation, and the case-study-driven evaluation of a small sensor-based, mobile system, which would require knowledge from the modul MOBI-DSC Data streams and event processing.</p> <p>The tasks in the project will be tailored to Bachelor level.</p>		
<b>Lernziele/Kompetenzen:</b>		
<p>Studierende sollen ein vertieftes Verständnis der bei der Durchführung von praktischen, arbeitsteilig organisierten, Softwareprojekten auftretenden Probleme wie auch von erfolgversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich der praktischen Informatik geschieht, gewinnen die TeilnehmerInnen wichtige Erfahrungen mit der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in einem wissenschaftlich ausgerichteten Arbeitsbericht.</p>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
<p>Programmierkenntnisse sowie grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z. B. erworben im Modul "Software Engineering Lab" (SWT-SWL-B), und zum wissenschaftlichen Arbeiten, z. B. erworben im Modul "Wissenschaftliches Arbeiten in der Informatik" (IAIWAI-B).</p>		keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester
<b>Lehrveranstaltungen</b>		
<b>Übung Bachelor project Mobile Software Systems (AI)</b>		4,00 SWS
<b>Lehrformen:</b> Übung		
<b>Dozenten:</b> Prof. Dr. Daniela Nicklas		
<b>Sprache:</b> Englisch/Deutsch		
<b>Angebotshäufigkeit:</b> SS, jährlich		
<b>Inhalte:</b>		
Projektdurchführung		

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten

Bearbeitungsfrist: 12 Wochen

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Prüfung Hausarbeit mit Kolloquium. Die Prüfungssprache wird zu Beginn der Lehrveranstaltung bekannt gegeben.

Production of a written report on the software project carried out (Assignment/Hausarbeit). Discussion of this project report and of the developed artefacts in the context of the wider project topic (Colloquium/Kolloquium).

<b>Modul SEDA-DMS-B Datenmanagementsysteme</b> <i>Datenmanagementsysteme</i>		6 ECTS / 180 h
(seit WS09/10) Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
<b>Inhalte:</b> Das Modul vermittelt eine systematische Einführung in das Gebiet der Datenmanagementsysteme.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden verstehen die Datenverwaltung auf der Basis des Relationenmodells und kennen grundlegende Architekturkonzepte für Datenmanagementsysteme. Sie erlernen methodische Grundlagen der konzeptuellen Datenmodellierung und verstehen dadurch in vertiefter Weise die Modellierung mit ERM und SERM. Die Studierenden erlernen die Grundlagen der Sprache SQL und können mit SQL Datenbankschemata generieren sowie zugehörige Datenbanken aufbauen und manipulieren. Schließlich sammeln sie erste Erfahrungen im Umgang mit realen Datenbankverwaltungssystemen.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Kenntnisse aus dem Modul Grundlagen betrieblicher Informationssysteme sind wünschenswert, jedoch nicht Voraussetzung.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Datenmanagementsysteme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Elmar J. Sinz <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	2,00 SWS
<b>Inhalte:</b> Datenmanagementsysteme sind zentrale Teilsysteme betrieblicher Anwendungssysteme. Ihre Entwicklung und ihr Betrieb stellen Kernaufgaben der Wirtschaftsinformatik dar. Das Modul vermittelt eine systematische Einführung in diesen Themenbereich. Der Fokus liegt dabei auf der Analyse, der Gestaltung und der Nutzung von Datenmanagementsystemen, nicht etwa auf der Implementierung von Datenbankverwaltungssystemen. Inhaltliche Schwerpunkte bilden das Relationenmodell, die Sprache SQL, Architekturen von Datenmanagementsystemen, der Entwurf von Datenbankschemata, theoretische Grundlagen der Datenmodellierung, Transaktionen und Transaktionsverwaltung sowie der Betrieb von Datenmanagementsystemen. Praktische Fertigkeiten werden insbesondere in Bezug auf den Entwurf von Datenbankschemata und SQL vermittelt. SQL wird anhand von konkreten	



<p>Datenbankverwaltungssystemen beübt. Fertigkeiten werden insbesondere in Bezug auf SQL vermittelt.</p> <p>Inhalte:</p> <ul style="list-style-type: none"> <li>• Einführung</li> <li>• Das Relationenmodell</li> <li>• Die Sprache SQL</li> <li>• Architekturen von Datenmanagementsystemen</li> <li>• Entwurf von Datenbankschemata</li> <li>• Fallstudie: Entwicklung eines Datenmanagementsystems</li> <li>• Theoretische Grundlagen der Datenmodellierung</li> <li>• Transaktionen und Transaktionsverwaltung</li> <li>• Betrieb von datenbankbasierten AWS</li> <li>• Alternative Entwicklungen im Bereich Datenbanken</li> </ul>	
<p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Date C.J.: An Introduction to database systems. 8th Edition, Addison-Wesley, Reading, Massachusetts 2003</li> <li>• Ferstl O.K., Sinz E.J.: Grundlagen der Wirtschaftsinformatik. 7. Auflage, Oldenbourg, München 2012, Kapitel 9.2</li> <li>• Kemper A., Eickler A.: Datenbanksysteme. Eine Einführung. 8. Auflage, Oldenbourg, München 2011</li> <li>• Pernul G., Unland R.: Datenbanken im Unternehmen. Analyse, Modellbildung und Einsatz. 2. Auflage, Oldenbourg, München 2003</li> <li>• Coronel C., Morris S., Rob P.: Database Systems. Design, Implementation, and Management. 9th Edition, Course Technology, Thomson Learning, Boston 2009</li> <li>• Vossen G.: Datenbankmodelle, Datenbanksprachen und Datenbankmanagement-Systeme. 5. Auflage, Oldenbourg, München 2008</li> </ul>	
<p><b>2. Übung Datenmanagementsysteme</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Mitarbeiter Wirtschaftsinformatik, insb. Systementwicklung und Datenbankanwendung</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <p>Die Inhalte der Vorlesung werden anhand von Übungsaufgaben und Fallbeispielen vertieft. Praktische Übungen werden unter Verwendung eines gängigen Datenbankverwaltungssystems durchgeführt.</p> <hr/> <p><b>Literatur:</b></p> <p>siehe Vorlesung</p>	2,00 SWS
<p><b>Prüfung</b></p> <p>schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	

<b>Modul SEDA-EuU-B Entrepreneurship und Unternehmensgründung</b> <i>Entrepreneurship and Foundation of an Enterprise</i>		3 ECTS / 90 h
Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
<b>Inhalte:</b> Die Veranstaltung vermittelt einen Überblick zum Themengebiet Unternehmensgründung. Schwerpunkt der Veranstaltung ist die Erstellung eines eigenen Businessplans im Verlauf des Semesters. Der Businessplan und insbesondere die eingeschlossene Finanzplanung dienen als Entscheidungsgrundlage pro oder contra Gründung des Unternehmens, indem sie die geplante wirtschaftliche Entwicklung und somit die Tragfähigkeit des Vorhabens aufzeigen.		
<b>Lernziele/Kompetenzen:</b> <ul style="list-style-type: none"> <li>• Studierende können Grundzüge eines Business Plans beschreiben und darstellen.</li> <li>• Studierende können einige nationale Fördermöglichkeiten für unternehmerische Selbständigkeit charakterisieren.</li> <li>• Studierende können ihre Geschäftsidee in einem Business Plan zusammenfassen.</li> <li>• Studierende übernehmen Verantwortung für Prozesse und Produkte des Arbeitens und Lernens in Kleingruppen.</li> <li>• Studierende reflektieren ihre Vorgehensweise bei Lehren und Lernen alleine und in einem gruppenbezogenen Kontext.</li> <li>• Studierende reflektieren ihre Fähigkeiten zur unternehmerischen Selbständigkeit.</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Entrepreneurship und Unternehmensgründung</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Dr. Markus Wolf <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS		2,00 SWS
<b>Inhalte:</b> Es werden folgende Punkte eines Businessplans diskutiert: <ul style="list-style-type: none"> <li>• Executive Summary</li> <li>• Kundennutzen und Alleinstellungsmerkmal</li> <li>• Markt &amp; Wettbewerbsanalysen</li> <li>• Marketing &amp; Vertrieb</li> <li>• Geschäftsmodell</li> <li>• Chancen &amp; Risiken</li> <li>• Realisierungsfahrplan</li> </ul>		

- |   |  |
|---|--|
| <ul style="list-style-type: none"><li>• Das Unternehmerteam</li><li>• Finanzplanung und Finanzierung des Unternehmens</li></ul> |  |
|---|--|

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten	
--	--

<b>Modul SEDA-PT-B Methoden der Präsentation, Gesprächsführung und Diskussion</b> <i>Methoden der Präsentation, Gesprächsführung und Diskussion</i>		3 ECTS / 90 h
Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
<b>Inhalte:</b> An Beispielen von Präsentationen, Einzelgesprächen und Diskussionen sollen <ul style="list-style-type: none"> <li>• persönliche Wirkung auf einzelne und Gruppen</li> <li>• formale und gruppensdynamische Abläufe und</li> <li>• inhaltliche Darstellungsformen</li> </ul> bewusst gemacht und zielbezogen für Präsentationen, für Gespräche und für Diskussionen geübt werden.		
<b>Lernziele/Kompetenzen:</b> Die persönliche Wirkung auf Einzelpersonen und Gruppen kennen lernen und verbessern; Inhalte sachlich verständlich, didaktisch ansprechend und adressatengerecht präsentieren; Kurzvorträge, Gespräche und Diskussionen führen und trainieren.		
<b>Sonstige Informationen:</b> Das Modul wird als Blockveranstaltung abgehalten.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Methoden der Präsentation, Gesprächsführung und Diskussion</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Dr. Ulrich Jentzsch <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS		2,00 SWS
<b>Inhalte:</b> Das Seminar ist als Training konzipiert. Methodisch kommen Einzel- und Gruppenübungen sowie Gruppenarbeiten zur Anwendung. Die persönlichen Verhaltensaspekte werden durch Videoaufzeichnungen dokumentiert und anschließend kommentiert.		
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten		

<b>Modul SEDA-TA-B Technikfolgeabschätzung / -bewertung</b>		3 ECTS / 90 h
<i>Technikfolgeabschätzung / -bewertung</i>		
Modulverantwortliche/r: Prof. Dr. Elmar J. Sinz		
<p><b>Inhalte:</b></p> <p>Das besondere Augenmerk liegt auf der untrennbaren Verflechtung von Naturwissenschaft, Technik und Wirtschaft mit ihren Entwicklungsimpulsen einerseits und der Bedeutung der zum Teil konträren weltanschaulichen Überzeugungen von Bevölkerungsgruppen andererseits. Dieses Spannungsverhältnis unterliegt der Technikfolgenbewertung vor allem:</p> <ul style="list-style-type: none"> <li>• durch das internationale Engagement der Unternehmen,</li> <li>• den immer weniger widerspruchlos akzeptierten Folgen der technisch-wirtschaftlichen Entwicklungen,</li> <li>• sowie dem Handikap, komplexe Prozesse mit weltanschaulichen Aspekten nicht durch streng wissenschaftliche Methoden erfassen zu können.</li> </ul>		
<p><b>Lernziele/Kompetenzen:</b></p> <p>Naturwissenschaften, Technik und Wirtschaft haben derzeit wohl den größten Einfluss auf das Denken, das Handeln und die Lebensbedingungen der Menschen in den Industrie- und Schwellenländern. Dieser Einfluss wirkt auf allen Ebenen der Gesellschaft bis auf das unternehmerische Verhalten mittelständischer Firmen. Daher wird anhand eines methodischen Rahmens versucht, aus der Sicht derer, die Naturwissenschaft, Technik und Wirtschaft für sich nutzbringend vorantreiben und aus der Sicht jener, die ohne Nutzen nur Betroffene sind, die wesentlichen Ziele, Kriterien und möglichen Folgen der daraus entstehenden Prozesse zu ermitteln, zu hinterfragen und nach festzulegenden Kriterien zu bewerten.</p>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
keine		keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester
<b>Lehrveranstaltungen</b>		
<p><b>Technikfolgeabschätzung / -bewertung</b></p> <p><b>Lehrformen:</b> Vorlesung und Übung</p> <p><b>Dozenten:</b> Dr. Ulrich Jentzsch</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p>		2,00 SWS
<p><b>Inhalte:</b></p> <p>Vermittlung von Grundkenntnissen:</p> <ul style="list-style-type: none"> <li>• zu den Begriffsinhalten einer TFA/TFB</li> <li>• zu zentralen Themenfeldern und Fragestellungen einer TFA/TFB</li> <li>• zu den Möglichkeiten und Grenzen prognostischer Aussagen im Rahmen einer TFA/TFB</li> </ul>		

<ul style="list-style-type: none"><li>• zum prinzipiellen inhaltlichen Aufbau und einer formalen Struktur einer TFA/TFB</li><li>• zu häufig verwendeten Methoden zur Problem- bzw. Entscheidungsanalyse innerhalb einer TFA/TFB - mit Übungen</li></ul>	
---	--

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten	
--	--

<b>Modul SME-Phy-B Physical Computing</b> <i>Physical Computing</i>		6 ECTS / 180 h
(seit WS14/15) Modulverantwortliche/r: Prof. Dr. Diedrich Wolter		
<p><b>Inhalte:</b></p> <p>Die in der physikalischen Umwelt eingebetteten Systeme können durch Sensoren ihre Umgebung erfassen und Wissen über ihren jeweiligen Kontext erlangen.</p> <p>Ziel dieses Kurses ist es, einen Überblick über die Möglichkeiten und Herausforderungen von Anwendungen im Bereich Physical Computing zu vermitteln. Physical Computing ist ein neues Gebiet an der Schnittstelle zu intelligenter Interaktion, eingebetteten Systemen und Smart Environments.</p> <p>Dieses Modul setzt zwei Schwerpunkte: Erstens, Kennenlernen von eingebetteten Sensorsystemen sowie Sammeln praktischer Erfahrung mit deren Programmierung und, zweitens, Algorithmen zur Verarbeitung von Sensordaten mit der Zielsetzung, Handlungen und Ereignisse zu erkennen und zu klassifizieren.</p> <p>Im Rahmen des Kurses programmieren Studierende in Kleingruppen ein eingebettetes System mit Sensorik und untersuchen Algorithmen zur Interpretation der Sensordaten:</p> <ul style="list-style-type: none"> <li>• Modellbasierte Interpretation unsicherer Information</li> <li>• Aktions- und Prozesserkennung mit Markov-Modellen</li> <li>• Sensorfusion mit dem Kalmanfilter</li> </ul>		
<p><b>Lernziele/Kompetenzen:</b></p> <ul style="list-style-type: none"> <li>• Einblick in die Programmierung eingebetteter Systeme</li> <li>• Erfahrung in der hardwarenahen Programmierung sammeln</li> <li>• Übersicht über Sensoren erlangen</li> <li>• Eignung von Sensoren zur Erkennung von Kontext und Umweltprozessen beurteilen</li> <li>• Kennenlernen von Algorithmen zur Interpretation von Sensordaten</li> </ul>		
<p><b>Sonstige Informationen:</b></p> <p>The main language of instruction in this course is German. Lectures and tutorials may be delivered in English on demand.</p>		
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p> <p>keine</p>		
<p><b>Empfohlene Vorkenntnisse:</b></p> <p>Elementare Grundkenntnisse der Programmierung sind dringend empfohlen (z.B. Modul MI-AuD-B), Grundkenntnisse in der Programmiersprache C können hilfreich sein.</p>		<p><b>Besondere Bestehensvoraussetzungen:</b></p> <p>keine</p>
<p><b>Angebotshäufigkeit:</b> SS, jährlich</p>	<p><b>Empfohlenes Fachsemester:</b></p>	<p><b>Minimale Dauer des Moduls:</b></p> <p>1 Semester</p>

<b>Lehrveranstaltungen</b>	
<p><b>1. Vorlesung Physical Computing</b></p> <p><b>Lehrformen:</b> Vorlesung</p> <p><b>Dozenten:</b> Prof. Dr. Diedrich Wolter</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p>	2,00 SWS

<p><b>Lernziele:</b> siehe Modulbeschreibung</p>	
<p><b>Inhalte:</b> siehe Modulbeschreibung</p>	
<p><b>Literatur:</b> wird in der ersten Vorlesung bekanntgegeben</p>	
<p><b>2. Übung zu Physical Computing</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Dr. Diedrich Wolter  <b>Sprache:</b> Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p>	<p>2,00 SWS</p>
<p><b>Inhalte:</b> Praktische Übungen zu den Inhalten der Vorlesung</p>	
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p>	



<b>Modul SME-Projekt-B Bachelorprojekt zu Smart Environments</b>		6 ECTS / 180 h 50 h Präsenzzeit 130 h Selbststudium
<i>Bachelor's project on Smart Environments</i>		
(seit WS15/16) Modulverantwortliche/r: Prof. Dr. Diedrich Wolter		
<b>Inhalte:</b> Das Modul behandelt die Umsetzung von Methoden aus dem Themenumfeld Smart Environments auf praktische Anwendungsprobleme im Rahmen eines Systementwicklungsprojektes mit Schwerpunkt auf der Softwareentwicklung. Dabei werden insbesondere Verfahren aus dem Umfeld der künstlichen Intelligenz (KI) eingesetzt. Folgende Aufgaben sind zu bewältigen: <ul style="list-style-type: none"> <li>• Literaturrecherche zu Lösungsansätzen für ein Anwendungsproblem</li> <li>• Entwurf eines Lösungsansatzes</li> <li>• Umsetzung des Entwurfs durch Implementation</li> <li>• Evaluation erzielter Ergebnisse und Dokumentation</li> </ul>		
<b>Lernziele/Kompetenzen:</b> <ul style="list-style-type: none"> <li>• Erlernen der Recherche, um für ein Anwendungsproblem relevante Lösungsansätze zu identifizieren</li> <li>• Fähigkeit erwerben, erlernte Methoden in einem konkreten Anwendungsproblem anzuwenden</li> <li>• ein Softwareentwicklungsprojekt unter Anleitung zu planen und selbständig durchzuführen</li> <li>• Methoden zur Lösung einer fachlichen Problemstellung beurteilen lernen</li> <li>• Kennenlernen des Spektrum von praktischen Problemen bei der Realisierung eines Systems</li> <li>• Ergebnisse in Form einer wissenschaftlichen Publikation und einem Vortrag darstellen können</li> </ul>		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand für dieses Modul gliedert sich grob wie folgt: <ul style="list-style-type: none"> <li>• Teilnahme an Gruppen- und Einzelbesprechungen: 50 Stunden</li> <li>• Entwurf und Umsetzung eines Lösungsansatzes: 100 Stunden</li> <li>• Anfertigung des Projektberichtes: 20 Stunden</li> <li>• Kolloquiumsvorbereitung: 10 Stunden</li> </ul> <p>The language of instruction in this course is German. However, all course materials are available in English. Term papers and presentations may be delivered in either German or English.</p>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Allgemeine Grundkenntnisse der Informatik, insbesondere der Programmierung sind dringend empfohlen.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> 1	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
<b>Lehrveranstaltungen</b>		
<b>Übung Bachelorprojekt zu Smart Environments</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Diedrich Wolter		4,00 SWS

<p><b>Sprache:</b> Deutsch/Englisch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b>          siehe Modulbeschreibung</p> <hr/> <p><b>Inhalte:</b>          Im Bachelorprojekt werden wechselnde Themen aus dem Gebiet Smart Environments in Kleingruppen bearbeitet. Problembasiert wird dabei wissenschaftliches Arbeiten und das Entwickeln eigener Lösungsansätze geübt. Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.</p> <hr/> <p><b>Literatur:</b>          wird in der ersten Veranstaltung bekanntgegeben</p>	
<p><b>Prüfung</b>          Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten          Bearbeitungsfrist: 4 Monate</p> <p><b>Zulassungsvoraussetzung zur Modulprüfung:</b>          Regelmäßige Teilnahme an der Lehrveranstaltung</p> <p><b>Beschreibung:</b>          Umsetzung der Projektaufgabe, Dokumentation in Form eines wissenschaftlichen Aufsatzes als Hausarbeit sowie Präsentation im Kolloquium. Die Gewichtung von Hausarbeit und Kolloquium wird zu Beginn der Lehrveranstaltung von der Dozentin bzw. dem Dozenten bekannt gegeben.</p> <p>Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstaltung.</p>	

<b>Modul Stat-B-01 Methoden der Statistik I</b> <i>Methoden der Statistik I</i>		6 ECTS / 180 h
(seit WS13/14) Modulverantwortliche/r: Prof. Dr. Susanne Rässler		
<b>Inhalte:</b> Im Rahmen der Vorlesung Methoden der Statistik I und der zugehörigen Übung werden die wichtigsten Grundlagen und Methoden der deskriptiven (beschreibenden) Statistik vermittelt. Dabei umfasst der Abschnitt zur deskriptiven Statistik Methoden, mit denen ein gegebenes Datenmaterial überschaubar dargestellt bzw. durch wenige aussagekräftige Zahlen wie Lageparameter, Streuungsmaße oder Korrelationskoeffizienten charakterisiert werden kann. Schließlich werden verschiedene Fragen der Datenerhebung angesprochen, denn eine noch so ausgefeilte statistische Methode ist nur so gut, wie die Daten, auf die sie angewendet wird.		
<b>Lernziele/Kompetenzen:</b> keine		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/stat-oek/leistungen/studium/infos-grundstudium-ba/">http://www.uni-bamberg.de/stat-oek/leistungen/studium/infos-grundstudium-ba/</a>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> ab dem 1.	<b>Minimale Dauer des Moduls:</b> 1 Semester Semester

<b>Lehrveranstaltungen</b>	
<b>Methoden der Statistik I</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS	4,00 SWS
<b>Lernziele:</b> Die Studierenden sollen mit den grundlegenden statistischen Methoden vertraut gemacht werden. Besondere Schwerpunkte bilden dabei die theoretischen Grundlagen dieser Methoden, die Voraussetzungen ihrer Anwendbarkeit, ihre Umsetzung in Statistiksoftware sowie die sinnvolle Interpretation der Ergebnisse.	
<b>Inhalte:</b> Im Rahmen der Vorlesung Methoden der Statistik I und der zugehörigen Übung werden die wichtigsten Grundlagen und Methoden der deskriptiven (beschreibenden) Statistik vermittelt. Dabei umfasst der Abschnitt zur deskriptiven Statistik Methoden, mit denen ein gegebenes Datenmaterial überschaubar dargestellt bzw. durch wenige aussagekräftige Zahlen wie Lageparameter, Streuungsmaße oder Korrelationskoeffizienten charakterisiert werden kann. Schließlich werden verschiedene Fragen der Datenerhebung angesprochen,	

denn eine noch so ausgefeilte statistische Methode ist nur so gut, wie die Daten, auf die sie angewendet wird.	
--	--

<b>Prüfung</b>	
----------------	--

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten	
--	--

<b>Zulassungsvoraussetzung zur Modulprüfung:</b>	
--	--

keine	
-------	--

<b>Modul Stat-B-02 Methoden der Statistik II</b> <i>Methoden der Statistik II</i>		6 ECTS / 180 h
(seit WS13/14) Modulverantwortliche/r: Prof. Dr. Susanne Rässler		
<p><b>Inhalte:</b> Im Rahmen der Vorlesung Methoden der Statistik II und der zugehörigen Übung werden die wichtigsten Grundlagen der Wahrscheinlichkeitsrechnung und der induktiven (schließenden) Statistik vermittelt. Im Einzelnen befasst sich die Vorlesung Methoden der Statistik II mit den grundlegenden Begriffen, Regeln und Gesetzmäßigkeiten der Wahrscheinlichkeitsrechnung, wobei vor allem Zufallsvorgänge, die sich durch sog. Zufallsvariablen beschreiben lassen, im Vordergrund des Interesses stehen. Viele aus der deskriptiven Statistik bekannte Größen, wie die Verteilungsparameter, können analog für Zufallsvariablen definiert werden. Außerdem werden mit dem Gesetz der großen Zahlen und dem zentralen Grenzwertsatz zwei für die induktive Statistik besonders wichtige Sätze der Wahrscheinlichkeitsrechnung vorgestellt.</p> <p>Des Weiteren stehen in der Vorlesung zur induktiven Statistik Methoden im Vordergrund, nach denen wahrscheinlichkeitstheoretisch fundierte Rückschlüsse von einer Stichprobe auf die betrachtete Grundgesamtheit möglich sind. Aufbauend auf den zuvor behandelten Grundlagen der Wahrscheinlichkeitstheorie werden Verfahren der Punktschätzung und der Intervallschätzung sowie wichtige Hypothesentests behandelt. Im Anschluss daran folgt ein Überblick über einige weitere interessante Teilgebiete der Statistik, wobei speziell die Methoden der Regressionsrechnung und der Analyse kategorialer Variablen ausführlicher besprochen werden.</p>		
<b>Lernziele/Kompetenzen:</b> keine		
<b>Sonstige Informationen:</b> <a href="http://www.uni-bamberg.de/stat-oek/leistungen/studium/infos-grundstudium-ba/">http://www.uni-bamberg.de/stat-oek/leistungen/studium/infos-grundstudium-ba/</a>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Die vorherige Absolvierung von Stat-B-01 (Methoden der Statistik I)		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> ab dem 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>Methoden der Statistik II</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> WS, SS</p>	4,00 SWS
<p><b>Lernziele:</b> Die Studierenden sollen mit den grundlegenden statistischen Methoden vertraut gemacht werden. Besondere Schwerpunkte bilden dabei die theoretischen Grundlagen dieser Methoden, die Voraussetzungen ihrer Anwendbarkeit, ihre Umsetzung in Statistiksoftware sowie die sinnvolle Interpretation der Ergebnisse.</p>	
<b>Inhalte:</b>	

Im Rahmen der Vorlesung Methoden der Statistik II und der zugehörigen Übung werden die wichtigsten Grundlagen der Wahrscheinlichkeitsrechnung und der induktiven (schließenden) Statistik vermittelt. Im Einzelnen befasst sich die Vorlesung Methoden der Statistik II mit den grundlegenden Begriffen, Regeln und Gesetzmäßigkeiten der Wahrscheinlichkeitsrechnung, wobei vor allem Zufallsvorgänge, die sich durch sog. Zufallsvariablen beschreiben lassen, im Vordergrund des Interesses stehen. Viele aus der deskriptiven Statistik bekannte Größen, wie die Verteilungsparameter, können analog für Zufallsvariablen definiert werden. Außerdem werden mit dem Gesetz der großen Zahlen und dem zentralen Grenzwertsatz zwei für die induktive Statistik besonders wichtige Sätze der Wahrscheinlichkeitsrechnung vorgestellt.

Des Weiteren stehen in der Vorlesung zur induktiven Statistik Methoden im Vordergrund, nach denen wahrscheinlichkeits-theoretisch fundierte Rückschlüsse von einer Stichprobe auf die betrachtete Grundgesamtheit möglich sind. Aufbauend auf den zuvor behandelten Grundlagen der Wahrscheinlichkeitstheorie werden Verfahren der Punktschätzung und der Intervallschätzung sowie wichtige Hypothesentests behandelt. Im Anschluss daran folgt ein Überblick über einige weitere interessante Teilgebiete der Statistik, wobei speziell die Methoden der Regressionsrechnung und der Analyse kategorialer Variablen ausführlicher besprochen werden.

**Literatur:**

Fahrmeir, L., Künstler, R., Pigeot, I., Tutz, G.: Statistik: Der Weg zur Datenanalyse, 5. Auflage, Springer, Heidelberg, 2004.  
 Agresti, A., Franklin, C. A.: Statistics: The Art and Science of Learning from Data, Prentice Hall, Upper Saddle River (New Jersey), 2006.  
 Krämer, W.: So lügt man mit Statistik, 8. Auflage, Piper, Frankfurt a.M., 2000.  
 Vogel, F.: Beschreibende und schließende Statistik - Formeln, Definitionen, Erläuterungen, Stichwörter und Tabellen, 13. Auflage, München, 2005.  
 Vogel, F.: Beschreibende und schließende Statistik - Aufgaben und Beispiele, 9. Aufl., München, 2001.

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Zulassungsvoraussetzung zur Modulprüfung:**

keine

<b>Modul SWT-FSE-B Foundations of Software Engineering</b>		6 ECTS / 180 h
<i>Foundations of Software Engineering</i>		
(seit WS15/16)		
Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b>		
This module teaches the foundations of software engineering that are applicable to various kinds of software systems – from information systems to embedded systems. It focusses on technologies, notations and processes for system specification, design, implementation, and verification and validation.		
<b>Lernziele/Kompetenzen:</b>		
Students will receive an introduction to the common problems and paradigms in, and foundations of, software development. They will also gather conceptual and practical knowledge in the analysis, design and testing of software, with an emphasis on technical aspects of specifying, designing, implementing, verifying and validating software.		
<b>Sonstige Informationen:</b>		
The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German.		
The total workload of 180 hrs. is split approximately as follows:		
<ul style="list-style-type: none"> <li>• 45 hrs. attending lectures (Vorlesungen)</li> <li>• 30 hrs. reviewing the lectures, including researching and studying material from additional sources</li> <li>• 45 hrs. attending practicals (Übungen)</li> <li>• 30 hrs. preparing and reviewing the practicals, including researching and studying material from additional sources</li> <li>• 30 hrs. preparing for the written exam (Klausur)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
Basic knowledge in Computer Science, as well as knowledge in programming in Java and in algorithms and data structures.		keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Vorlesung Foundations of Software Engineering</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Gerald Lüttgen <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	3,00 SWS
<b>Inhalte:</b>	
The lectures (Vorlesungen) provide an introduction to the foundations of software engineering, including commonly used technologies, notations and processes for all software engineering phases. In particular, conceptual and technical aspects	

<p>of software specification, architecture and design, and verification and validation are discussed, such as the Unified Modeling Language (UML) and its semantics, model-driven and pattern-based development, and software testing. Students are also introduced to specific aspects of agile software development.</p> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Sommerville, I. Software Engineering, 9th ed. Addison-Wesley, 2010.</li> <li>• Robertson, S. and Robertson, J. Mastering the Requirements Process, 2nd ed. Addison-Wesley, 2006.</li> <li>• Cohn, M. User Stories Applied. Addison-Wesley, 2004.</li> <li>• Stevens, P. and Pooley, R. Using UML - Software Engineering with Objects and Components, 2nd. ed. Addison-Wesley, 2008.</li> <li>• Freeman, E., Freeman, E., Sierra, K. and Bates, B. Head First Design Patterns. O'Reilly, 2004.</li> <li>• Gamma, E., Helm, R., Johnson, R. and Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Design. Addison-Wesley, 1994.</li> </ul> <p>Further literature will be announced in the lectures.</p>	
<p><b>2. Übung Foundations of Software Engineering</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <p>The practicals (Übungen) exercise and deepen the conceptual knowledge transferred via the lectures (Vorlesungen), and relay practical knowledge in software engineering.</p> <hr/> <p><b>Literatur:</b></p> <p>- see the corresponding lectures -</p>	<p>3,00 SWS</p>
<p><b>Prüfung</b></p> <p>schriftliche Prüfung (Klausur) / Prüfungsdauer: 120 Minuten</p> <p><b>Beschreibung:</b></p> <p>Written exam (Klausur) consisting of questions that relate to the contents of the lectures (Vorlesungen) and practicals (Übungen) of this module.</p> <p>The written exam is set in English, while answers may be provided in either English or German. The exam is passed if at least 50% of the available points are reached.</p>	



<b>Modul SWT-PR1-B Bachelorprojekt Softwaretechnik und Programmiersprachen</b> <i>Bachelors Project in Software Engineering and Programming Languages</i>	6 ECTS / 180 h
(seit WS13/14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen	
<p><b>Inhalte:</b></p> <p>Überschaubare Themen aus der aktuellen Forschungsarbeit der Softwaretechnik und Programmiersprachen werden in einer zum Teil individuell und zum Teil in einer arbeitsteilig arbeitenden Gruppe von Studierenden von der Konzeption bis zur praktischen Umsetzung durchgeführt. Dabei geht es nicht nur um die programmiertechnische Umsetzung, sondern insbesondere auch um die Entwicklung tragfähiger und mit den vorgegebenen Rahmenbedingungen kompatibler Konzepte zur Lösung der gestellten Aufgabe, sowie um die Sicherstellung der robusten und verlässlichen Funktionen der entwickelten Systeme. In der Regel ist dazu das Studium aktueller Literatur und die Auswahl, Umsetzung und/oder Adaption zum Thema vorgeschlagener Ansätze notwendig.</p> <p>Ein Beispiel für eine solche Aufgabe wäre die Entwicklung eines Compilers in der funktionalen Programmiersprache Haskell, für die Kenntnisse aus dem Modul "Principles of Compiler Construction" (SWT-PCC-M; auch für Bachelor-Studierende im Rahmen der Profilbildung) bzw. vergleichbare Kenntnisse erwartet werden. Ein weiteres Beispiel wäre eine experimentelle, auf Fallbeispielen basierende Studie von verschiedenen Werkzeugen zur Softwareanalyse bzw. -verifikation. Für eine derartige Aufgabe werden Kenntnisse aus dem Modul "Foundations of Software Analysis" (SWT-FSA-B) bzw. vergleichbare Kenntnisse erwartet.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Studierende sollen ein vertieftes Verständnis der bei der Durchführung von theoretischen und praktischen Softwareprojekten auftretenden konzeptioneller Probleme wie auch von erfolgversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich der Softwaretechnik und Programmiersprachen geschieht, gewinnen die Studierenden wichtige Erfahrungen mit der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in einem wissenschaftlich ausgerichteten Arbeitsbericht.</p>	
<p><b>Sonstige Informationen:</b></p> <p>Der Arbeitsaufwand beträgt 180 Std., welche sich grob wie folgt gliedern:</p> <ul style="list-style-type: none"> <li>• 30 Std. Einführung, Vorstellen von Werkzeugen und Vorträge zum Projektstand</li> <li>• 30 Std. Recherchen zu und Einarbeitung in die Thematik des Praktikums (inkl. Vorbereitung von Kurzvorträgen)</li> <li>• 80 Std. Projektarbeit (Softwareentwicklung)</li> <li>• 40 Std. Erstellung des Projektberichts (Hausarbeit) und Vorbereitung auf das Kolloquium</li> </ul> <p>Der Projektbericht darf wahlweise in Deutsch oder Englisch abgefasst sein. Eine regelmäßige Teilnahme an den Projekttreffen ist erforderlich.</p>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b></p>	<p><b>Besondere Bestehensvoraussetzungen:</b></p>

Grundlegende Kenntnisse in Softwaretechnik und Programmiersprachen, sowie Kenntnisse in den Grundlagen des im Projekt behandelten Themengebiets.		keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>Übung Bachelorprojekt Softwaretechnik und Programmiersprachen</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b>                  Werden zu Beginn des Projekts bekannt gegeben.</p> <hr/> <p><b>Inhalte:</b>                  Durchführung des Projekts, begleitet von regelmäßigen Treffen zwischen Teilnehmerinnen/Teilnehmern und Projektbetreuer.</p> <hr/> <p><b>Literatur:</b>                  Je nach Problematik; wird zu Beginn des Projekts bekannt gegeben.</p>	4,00 SWS

<p><b>Prüfung</b>                  Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten                  Bearbeitungsfrist: 12 Wochen  <b>Zulassungsvoraussetzung zur Modulprüfung:</b>                  Regelmäßige Teilnahme an den zugehörigen Lehrveranstaltungen  <b>Beschreibung:</b>                  Anfertigen eines schriftlichen Berichts in englischer oder deutscher Sprache über das durchgeführte Softwareprojekt (Hausarbeit). Diskussion des vorliegenden Projektberichts sowie der erstellten Artefakte vor dem Hintergrund des allgemeinen Themas der Projektarbeit (Kolloquium). Das Kolloquium kann wahlweise in englischer oder deutscher Sprache abgehalten werden.</p>	
---	--

<b>Modul SWT-RSD-B Reactive Systems Design</b> <i>Reactive Systems Design</i>		6 ECTS / 180 h
(seit SS17) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b> Reactive systems are digital systems that continuously react to their environment by reading sensor values, computing output values and emitting those values to actuators (sense-compute-emit loop). Such systems are quickly becoming parts of our daily lives: whether it is a home automation system, a driver's assistance system in a modern car, or sophisticated medical equipment at the hospital, we depend on their reliability and on the quality of their software.  This module discusses the theory and practice of designing and programming reactive systems software using so-called <i>synchronous</i> languages for expressing sense-compute-emit loops. The module's foci are on different synchronous language paradigms, on automatic code generation from system models, on techniques for verifying design properties and testing reactive software, and on modern development environments that are widely used in the aerospace, railway, automotive and energy industries.		
<b>Lernziele/Kompetenzen:</b> On completion of this module, students will be able to understand the context and concepts of reactive systems design. They will be able to employ state-of-the-art synchronous languages for programming reactive software, with the help of modern development environments that automatically generate code from models. Students will also be able to apply methods for testing and verifying reactive systems.		
<b>Sonstige Informationen:</b> The main language of instruction is English. The lectures and practicals may be delivered in German if all participating students are fluent in German.  The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> <li>• 30 hrs. attending lectures (Vorlesungen)</li> <li>• 30 hrs. attending practicals (Übungen)</li> <li>• 60 hrs. preparing and reviewing the lectures and practicals, including researching literature, studying material from additional sources</li> <li>• 60 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge in discrete mathematics and programming, e.g., acquired in the modules " <i>Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik)</i> " (GdI-MfI-1) and " <i>Einführung in Algorithmen, Programmierung und Software</i> " (DSG-EiAPS-B). Knowledge gained in program semantics and verification, e.g., in the module " <i>Foundations of Software Analysis</i> " (SWT-FSA-B), is beneficial but not necessary for following the module's content.		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b> 4.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>1. Vorlesung Reactive Systems Design</b>  <b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Dr. Gerald Lüttgen  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b>  – see the module's learning outcomes/competences (Lernziele/Kompetenzen) listed above –</p> <hr/> <p><b>Inhalte:</b>  Students are introduced to modern techniques and tools for designing and programming reactive systems. The lectures first motivate reactive systems and the synchronous hypothesis underlying their development, and present their basic design principles. The emphasis is on two distinctive paradigms of synchronous languages: the data-flow paradigm (as exemplified by, e.g., the language <i>Lustre</i>) and the control-flow <i>paradigm</i> (e.g., <i>Esterel</i> and <i>Statecharts</i>). In each case, the language's syntax and semantics is given in both an example-driven manner and a formal manner, before techniques for verifying design properties via model checking, for automatically generating running code from design models, and for automated testing are studied. The practicality of synchronous languages and techniques for reactive systems development is emphasised by exploring industrial development tools throughout.</p> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Potop-Butucaru, D., Edwards, S.A. and Berry, G. Compiling Esterel. Springer, 2007.</li> <li>• Halbwachs, N. Synchronous Programming of Reactive Systems. Springer, 1993.</li> <li>• Edwards, S. A. Languages for Digital Embedded Systems, Kluwer Academic Publishers, 2000.</li> <li>• Berry, G. The Constructive Semantics of Pure Esterel. Esterel Technologies, 1999.</li> <li>• Harel, D. and Politi, M. Modeling Reactive Systems with Statecharts. McGraw-Hill, 1998.</li> <li>• Broy, M., Jonsson, B., Katoen, J.-P., Leucker, M., Pretschner, A. (eds.). Model-Based Testing of Reactive Systems. Springer, 2005.</li> </ul>	2,00 SWS
<p><b>2. Übung Reactive Systems Design</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b>  – see the module's learning outcomes/competences (Lernziele/Kompetenzen) listed above –</p> <hr/> <p><b>Inhalte:</b></p>	2,00 SWS

<p>The practicals (Übungen) deepen the concepts and techniques taught in the lectures (Vorlesungen) and apply them to the development of reactive software. The latter involves a small programming project of a real reactive system with a modern development tool; for example, such a project could be the design and implementation of a controller for a model railway using <i>SCADE</i>.</p>	
<p><b>Literatur:</b> – see the corresponding lectures –</p>	
<p><b>Prüfung</b> Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten Bearbeitungsfrist: 3 Wochen</p> <p><b>Beschreibung:</b> The Assignment (Hausarbeit) consists of questions practicing, reviewing and deepening the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen); questions may also involve the practical use of the development tool introduced in the practicals. The assignment is set in English; students may answer in either English or German.</p> <p>The Colloquium (Kolloquium) consists of questions testing the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen), on the basis of the submitted solutions to the assignment (Hausarbeit). The examination language is either English or German and may be chosen by the student at the colloquium.</p>	

<b>Modul SWT-SSP-B Soft Skills in IT-Projekten</b> <i>Soft Skills for IT Projects</i>		3 ECTS / 90 h 30 h Präsenzzeit 60 h Selbststudium
(seit SS14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b> Ziel des Moduls ist es, die in der Praxis der IT-Projekte immer wichtiger werdenden Soft Skills wissenschaftlich und methodisch fundiert zu vermitteln. Die Studierenden lernen, dieses Wissen in der Praxis ziel- und lösungsorientiert anwenden zu können.		
<b>Lernziele/Kompetenzen:</b> Nach erfolgreicher Teilnahme an diesem Modul können Studierende insbesondere: <ul style="list-style-type: none"> <li>• Sich die Bedeutung menschlicher Faktoren in großen IT-Projekten bewusst machen;</li> <li>• Erfolgsfaktoren der Teamarbeit kennen und einschätzen;</li> <li>• Eigenkompetenzen und Kompetenzen anderer wahrnehmen, beurteilen und für die Teamorganisation nutzen;</li> <li>• Muster der Gruppendynamik - insbes. Kommunikationsmuster, Konfliktsituationen und Verantwortungsdiffusion - erkennen und managen.</li> </ul>		
<b>Sonstige Informationen:</b> Der Arbeitsaufwand beträgt 90 Std., welche sich grob wie folgt gliedern: <ul style="list-style-type: none"> <li>• 30 Std. Teilnahme an der Vorlesung und Übung</li> <li>• 45 Std. Vor- und Nachbereitung der Vorlesung und Übung</li> <li>• 15 Std. Vorbereitung auf die Klausur</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Keine		<b>Besondere Bestehensvoraussetzungen:</b> keine
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Soft Skills in IT-Projekten</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Norbert Seifert <b>Sprache:</b> Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	2,00 SWS 3 ECTS
<b>Inhalte:</b> Der Inhalt orientiert sich an der in der Praxis großer IT-Projekte erforderlicher Soft Skills: <ol style="list-style-type: none"> <li>1. Vorsprung durch Menschenkenntnis;</li> <li>2. Teamorganisation und -aufstellung;</li> <li>3. Kommunikation und Konfliktmanagement;</li> <li>4. Motivationsfaktoren und Selbstverantwortung;</li> </ol>	

5. Menschliche Spielregeln großer IT-Projekte.	
<b>Literatur:</b> Literatur wird zu Beginn der Lehrveranstaltung angegeben.	
<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten <b>Beschreibung:</b> Die Klausur prüft Wissen und Verständnis der in der Vorlesung und Übung vermittelten Lehrinhalte.	

<b>Modul SWT-SWL-B Software Engineering Lab</b> <i>Software Engineering Lab</i>		6 ECTS / 180 h
(seit WS13/14) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b> Small teams of students will conduct a software project, starting from a brief problem description. This involves the application of modern software engineering tools, skills in collaboration and team organisation, and knowledge of processes and techniques for producing software artefacts and associated documents.		
<b>Lernziele/Kompetenzen:</b> Students will develop a piece of medium-sized software in small teams, thereby acquiring practical expertise in software engineering and skills in working in a software development team. In addition, this module deepens the students' programming proficiency and their understanding of flexible software engineering processes and of software and process quality, and familiarises them with the deployment and use of modern software engineering tools.		
<b>Sonstige Informationen:</b> The main language of instruction is English. The practicals may be delivered in German if all participating students are fluent in German. A regular attendance of team meetings and active participation is required throughout.  The total workload of 180 hrs. is split approximately as follows: <ul style="list-style-type: none"> <li>• 15 hrs. attending meetings of the student's team with the lecturer (Dozent) on planning, coordination and feedback</li> <li>• 10 hrs. attending the accompanying practicals/tutorials (Übungen/Tutorials) on software tools</li> <li>• 130 hrs. conducting the team project</li> <li>• 25 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge in Computer Science and Software Engineering, as well as knowledge in Java programming and in programming in the small.		<b>Besondere Bestehensvoraussetzungen:</b>
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Übung Software Engineering Lab</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Gerald Lüttgen, Mitarbeiter Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich <b>Inhalte:</b>	4,00 SWS



Each team will carry out a software project. It will also regularly meet with their tutor (Dozent) in order to critically reflect on the team's work, and participate in tutorials that introduce the software engineering tools and some software engineering techniques to be used in this project.

**Literatur:**

- Tachiev, P., Leme, F., Massol, V. and Gregory, G. JUnit in Action, 2nd ed. Manning Publications, 2010.
- Loeliger, J. and McCullough, M. Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development, 2nd ed. O'Reilly, 2012.
- Vogel, L. Eclipse IDE. Lars Vogel, 2013. ISBN 3943747042.
- Schwaber, K. and Beedle, M. Agile Software Development with Scrum, Prentice Hall, 2001
- Cohn, M. User Stories Applied. Addison-Wesley, 2004.

See the description of the module "Foundations of Software Engineering (SWT-FSE-B)" for further literature.

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 45 Minuten

Bearbeitungsfrist: 2 Wochen

**Zulassungsvoraussetzung zur Modulprüfung:**

Regular participation in the associated practicals

**Beschreibung:**

Assignment (Hausarbeit) involving the compilation of a written project report in English or German language by each team, which shall cover the following topics:

- A description of the team's produced artefacts, plus the electronic submission of the artefacts themselves;
- A description, justification and critical reflection of the employed software engineering processes, methods and techniques in general and in each development phase;
- A description of the team's organisation, the distribution of work and the contributions of each team member.

The submission deadline and the details of the required content and format of this report will be announced at the beginning of the semester.

Colloquium (Kolloquium) consisting of a critical discussion of the team's produced software and project report with respect to the taken design decisions and possible alternatives, the quality of the produced artefacts and documentation, the project's status and completeness, the conduct of testing, and the appropriateness of the employed techniques and processes. The colloquium takes place in the presence of the team as a whole, but each question will be addressed to a specific student so that marks can be individualised. The colloquium can be held electively in English or German language.

Because this module involves a team effort, the examination can only be resit in a winter semester.