



# **Modulhandbuch**

## **Masterstudiengang International Software Systems Science**

### **Wirtschaftsinformatik und Angewandte Informatik**

**Gemäß der geltenden Fassung der Studien- und Fachprüfungsordnung vom  
06.03.2015 für den Masterstudiengang International Software Systems Science an  
der Otto-Friedrich-Universität Bamberg. Gültig ab Wintersemester 2019/20.**

---

## **Hinweis zur Weitergeltung älterer Fassungen eines Modulhandbuchs:**

### **1. Geltungsbeginn**

Die im vorliegenden Modulhandbuch enthaltenen Modulbeschreibungen gelten erstmals für das Semester, das auf dem Deckblatt angegeben ist.

### **2. Übergangsbestimmung**

- a. Studierende, die gemäß bisher geltendem Modulhandbuch ein Modul bereits in Teilen absolviert haben (vgl. Nr. 2b), schließen das Modul nach der bisher geltenden Fassung des Modulhandbuchs ab.

Diese Übergangsbestimmung gilt ausschließlich für den dem versäumten/nicht bestandenen/nicht absolvierten regulären Prüfungstermin unmittelbar folgenden Prüfungstermin. Auf Antrag der oder des Studierenden kann der Prüfungsausschuss in begründeten Fällen eine Verlängerung der Übergangsfrist festlegen.

- b. Ein Modul ist in Teilen absolviert, wenn die Modulprüfung nicht bestanden oder versäumt wurde. Gleiches gilt für den Fall, dass zumindest eine Modulteilprüfung bestanden, nicht bestanden oder versäumt wurde.

Ferner gilt ein Modul als in Teilen absolviert, sofern sich die oder der Studierende gemäß bisher geltendem Modulhandbuch zu einer dem jeweiligen Modul zugeordneten Lehrveranstaltung angemeldet hat.

### **3. Geltungsdauer**

Das Modulhandbuch gilt bis zur Bekanntgabe eines geänderten Modulhandbuchs auch für nachfolgende Semester.

**Äquivalenzliste MA International Software Systems Science,  
StuFPO vom 06.03.2015**

Im Folgenden finden Sie eine Auflistung von Modulen, deren Bezeichnung bzw. Kürzel geändert wurde, ohne dass damit eine wesentliche Änderung des Moduls verbunden ist. Sofern ein in der Spalte „bisheriges Modul“ aufgeführtes Modul erfolgreich absolviert wurde, kann das in der Spalte „neues Modul“ angegebene Modul nicht belegt werden.

bisheriges Modul			neues Modul		
Modulkürzel	Modulbezeichnung	bis (Semester)	Modulkürzel	Modulbezeichnung	ab (Semester)
MOBI-DSC	Data Streams and Complex Event Processing	SS 18	MOBI-DSC-M	Data Streams and Complex Event Processing	WS 1819



---

## Module

DSG-DSAM-M: Distributed Systems Architecture and Middleware.....	8
DSG-DistrSys-M: Distributed Systems.....	10
DSG-Project-M: Masterprojekt Verteilte Systeme.....	13
DSG-SOA-M: Service-Oriented Architecture and Web Services.....	15
DSG-SRDS-M: Selected Readings in Distributed Systems.....	18
EESYS-DAE-M: Data Analytics in der Energieinformatik.....	20
EESYS-ES-M: Energieeffiziente Systeme.....	23
Gdl-AFP-M: Advanced Functional Programming.....	26
Gdl-CSNL-M: Computational Semantics of Natural Language.....	28
Gdl-FP-M: Functional Programming.....	30
HCI-MCI-M: Mensch-Computer-Interaktion.....	32
HCI-US-B: Ubiquitäre Systeme.....	35
KTR-GIK-M: Grundbausteine der Internet-Kommunikation.....	38
KTR-MAKV-M: Modellierung und Analyse von Kommunikationsnetzen und Verteilten Systemen.....	41
KTR-MMK-M: Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen.....	44
KTR-Mobi-M: Mobilkommunikation.....	47
KTR-SSSProj-M: KTR Masterprojekt Software Systems Science.....	50
KogSys-ML-M: Lernende Systeme (Machine Learning).....	53
MOBI-ADM-M: Advanced Data Management.....	56
MOBI-DSC-M: Data Streams and Complex Event Processing.....	58
MOBI-PRS-M: Master Project Mobile Software Systems (SoSySc).....	60
PSI-AdvaSP-M: Advanced Security and Privacy.....	62
PSI-ProjectCAD-M: Project Complex Attacks and Defenses.....	65
PSI-ProjectSP-M: Project Security and Privacy.....	68
SME-STE-M: Introduction to Knowledge Representation: Space, Time, Events.....	70
SNA-OSN-M: Projekt zu Online Social Networks.....	72
SSS-PraktIntKon-M: Praktikum im International Kontext.....	74
SSS-SEM-M: Master-level Seminar in Software Systems Science.....	75
SSS-Thesis-M: Master Thesis in Software Systems Science.....	77

---

## Inhaltsverzeichnis

---

SWT-ASV-M: Applied Software Verification.....	79
SWT-PCC-M: Principles of Compiler Construction.....	81
SWT-PR2-M: SWT Masters Project in Software Systems Science.....	83

---

# Übersicht nach Modulgruppen

## 1) A1 Software Systems Science (Modulgruppe) ECTS: 30 - 48

In den Modulgruppen A1 und A2 sind Module im Gesamtvolumen von 48 ECTS-Punkten unter Einhaltung der für die Modulgruppen geltenden Mindest- und Höchstgrenzen zu absolvieren.

### a) Pflichtbereich (Teilmodulgruppe) ECTS: 30

DSG-DSAM-M: Distributed Systems Architecture and Middleware (6 ECTS, WS, jährlich).....	8
KTR-GIK-M: Grundbausteine der Internet-Kommunikation (6 ECTS, SS, jährlich).....	38
MOBI-DSC-M: Data Streams and Complex Event Processing (6 ECTS, WS, jährlich).....	58
PSI-AdvaSP-M: Advanced Security and Privacy (6 ECTS, SS, jährlich).....	62
SWT-PCC-M: Principles of Compiler Construction (6 ECTS, WS, jährlich).....	81

### b) Wahlpflichtbereich (Teilmodulgruppe) ECTS: 0 - 18

DSG-DistrSys-M: Distributed Systems (6 ECTS, SS, jährlich).....	10
DSG-SOA-M: Service-Oriented Architecture and Web Services (6 ECTS, SS, jährlich).....	15
DSG-SRDS-M: Selected Readings in Distributed Systems (3 ECTS, WS, SS).....	18
Gdl-AFP-M: Advanced Functional Programming (6 ECTS, SS, jährlich).....	26
Gdl-FP-M: Functional Programming (6 ECTS, WS, jährlich).....	30
KTR-MAKV-M: Modellierung und Analyse von Kommunikationsnetzen und Verteilten Systemen (6 ECTS, SS, jährlich).....	41
KTR-MMK-M: Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen (6 ECTS, SS, jährlich).....	44
KTR-Mobi-M: Mobilkommunikation (6 ECTS, WS, jährlich).....	47
MOBI-ADM-M: Advanced Data Management (6 ECTS, SS, jährlich).....	56
SWT-ASV-M: Applied Software Verification (6 ECTS, SS, jährlich).....	79

## 2) A2 Domain-specific Software Systems Science (Bereich) ECTS: 0 - 18

In den Modulgruppen A1 und A2 sind Module im Gesamtvolumen von 48 ECTS-Punkten unter Einhaltung der für die Modulgruppen geltenden Mindest- und Höchstgrenzen zu absolvieren.

EESYS-DAE-M: Data Analytics in der Energieinformatik (6 ECTS, WS, jährlich).....	20
EESYS-ES-M: Energieeffiziente Systeme (6 ECTS, SS, jährlich).....	23
Gdl-CSNL-M: Computational Semantics of Natural Language (6 ECTS, SS, jährlich).....	28
HCI-MCI-M: Mensch-Computer-Interaktion (6 ECTS, SS, jährlich).....	32
HCI-US-B: Ubiquitäre Systeme (6 ECTS, WS, jährlich).....	35

---

KogSys-ML-M: Lernende Systeme (Machine Learning) (6 ECTS, WS, jährlich).....	53
SME-STE-M: Introduction to Knowledge Representation: Space, Time, Events (6 ECTS, WS, jährlich).....	70
SNA-OSN-M: Projekt zu Online Social Networks (6 ECTS, WS, jährlich).....	72

### 3) A3 Seminar and Project (Modulgruppe) ECTS: 12

#### a) Seminar (Teilmodulgruppe) ECTS: 3

SSS-SEM-M: Master-level Seminar in Software Systems Science (3 ECTS, WS, SS).....	75
---	----

#### b) Projekt (Teilmodulgruppe) ECTS: 9

**Attention: The module SWT-PR2-M cannot take place in winter semester 19/20!**

MOBI-PRS-M: Master Project Mobile Software Systems (SoSySc) (9 ECTS, SS, jährlich).....	60
DSG-Project-M: Masterprojekt Verteilte Systeme (9 ECTS, WS, SS).....	13
SWT-PR2-M: SWT Masters Project in Software Systems Science (9 ECTS, WS, SS).....	83
KTR-SSSProj-M: KTR Masterprojekt Software Systems Science (9 ECTS, WS, SS).....	50
PSI-ProjectCAD-M: Project Complex Attacks and Defenses (9 ECTS, WS, SS).....	65
PSI-ProjectSP-M: Project Security and Privacy (9 ECTS, WS, SS).....	68

### 4) A4 Master Thesis (Modulgruppe) ECTS: 30

SSS-Thesis-M: Master Thesis in Software Systems Science (30 ECTS, WS, SS).....	77
--	----

### 5) A5 International Experience (Modulgruppe) ECTS: 30

In der Modulgruppe A5 sind Module im Umfang von 30 ECTS-Punkten zu absolvieren.

Zur Auswahl stehen die im Folgenden genannten Module der Wahlpflichtbereiche 5a bis 5c. Soweit in diesen Wahlpflichtbereichen insgesamt weniger als 30 ECTS-Punkte erbracht werden, sind weitere, noch nicht absolvierte Module der Modulgruppen A1, A2 oder A3 zu absolvieren.

#### a) Module aus dem gelenkten Auslandsstudium (Teilmodulgruppe) ECTS: 0 - 30

Im Wahlpflichtbereich a können Module im Umfang von 0 bis 30 ECTS-Punkten eingebracht werden, die im Rahmen eines gelenkten Auslandsstudiums an einer ausländischen Hochschule absolviert werden, sofern sie sich wesentlich von den nach Vorgabe der vorliegenden Ordnung zu absolvierenden Modulen unterscheiden und fachsystematisch den Modulgruppen A1, A2 oder A3 zugeordnet werden können.

#### b) Praktikum im internationalen Kontext (Teilmodulgruppe) ECTS: 0 - 12

SSS-PraktIntKon-M: Praktikum im International Kontext (12 ECTS, WS, SS).....	74
--	----

#### c) Fremdsprachen (Teilmodulgruppe) ECTS: 0 - 18

---



Im Wahlpflichtbereich 5c können Module im Umfang von 0 bis 18 ECTS-Punkten aus dem Angebot des Sprachenzentrums Bamberg absolviert werden. Ausgenommen sind Module der englischen Sprache sowie Module der Sprache, in der die Hochschulzugangsberechtigung erworben wurde.

Einzelheiten, insbesondere die zur Auswahl stehenden Module sowie die jeweils abzulegenden Modulprüfungen und Modulteilprüfungen sind in der Prüfungsordnung und dem Modulhandbuch für sprachpraktische Module der Otto-Friedrich-Universität Bamberg festgelegt.

---

<b>Modul DSG-DSAM-M Distributed Systems Architecture and Middleware</b> <i>Distributed Systems Architectures and Middleware</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS19/20) Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
<b>Inhalte:</b> <p>Distributed Systems are at the heart of modern computing. Web storage, web applications, cross-organizational information systems, enterprise information systems, ... almost anything is a distributed system. Even simple tasks such as the synchronization of a mobile device with a desktop machine is subject to distributed systems rules. While DSG-IDistrSys focuses on the fundamental principles of distributed systems and some low-level implementation technologies this course investigates enterprise-level distributed computing.</p> <p>This course introduces students to the ideas, benefits, technologies and issues related to server-centric distributed systems and middleware in general. The core topics are centered around component technologies such as Java EJBs, Business-to-Business technologies like EDI and ebXML, and Cloud Computing facilities like Google App Engine and Windows Azure. Thus the course introduces and discusses in-depth topics concerning distributed middleware and its practical use:</p> <ul style="list-style-type: none"> <li>• Characteristics and Foundations of Distributed Systems</li> <li>• Classical Middleware and Services</li> <li>• Concurrency and Synchronization</li> <li>• Component Technologies</li> <li>• Cloud Computing, in particular platform as a service</li> <li>• Business-to-Business Technologies</li> </ul> <p>The selection of topics and teaching method of this course reflects the Distributed Systems Group's (DSG) dedication to integrate business and IT, theory and practice, research and teaching. You not only will be taught the classical way, but you will have hands-on experience on middleware development and middleware tools. Also, you will get the chance to discuss selected publications with your lecturers.</p>		
<b>Lernziele/Kompetenzen:</b> Students are able to evaluate, plan, design and implement server-centric distributed systems. Students are familiar with recent approaches and standards for building and managing such systems, know about the central problems involved as well as ways to overcome these issues. Students have hands-on experience with up-to-date middleware and tools for building server-centric systems.		
<b>Sonstige Informationen:</b> The main language of instruction in this course is English.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> Basic knowledge in software engineering and in distributed systems as introduced, e.g., in the module DSG-IDistrSys-B (or DSG-DistrSys-M).		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge in software engineering and in distributed systems as introduced, e.g., in the module DSG-IDistrSys-B or DSG-DistrSys-M.		<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. Distributed Systems Architecture and Middleware</b></p> <p><b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Dr. Guido Wirtz  <b>Sprache:</b> Englisch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b>  c.f. overall module description</p> <hr/> <p><b>Inhalte:</b>  c.f. overall module description</p> <hr/> <p><b>Literatur:</b>  This is a fast emerging field with new insights every year. So, up-to-date literature will be provided at the beginning of each course.</p>	<b>2,00 SWS</b>
<p><b>2. Distributed Systems Architecture and Middleware</b></p> <p><b>Lehrformen:</b> Übung  <b>Dozenten:</b> Mitarbeiter Praktische Informatik  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>  Introduction to and discussion of tools and practical issues closely related to the topics discussed in the lecture as well as solutions of problems that come up during working on the practical assignment.</p> <hr/> <p><b>Literatur:</b>  see lecture</p>	<b>2,00 SWS</b>
<p><b>Prüfung</b>  Hausarbeit mit Kolloquium / Prüfungsdauer: 15 Minuten  Bearbeitungsfrist: 3 Monate</p> <p><b>Beschreibung:</b>  Oral examination concerning the topics discussed in the lecture, exercises and assignment. <b>Students may choose English or German as the language for the oral examination.</b> Examinations will take place at the end of the winter term or at the begin of the summer term (students may choose one of them).</p> <p>Students are assumed to work on a programming assignment ('schriftliche Hausarbeit') during the semester that is introduced at the beginning of the semester and uses the most important technologies discussed during the semester.</p> <p><b>Note:</b> Without working on the programming assignment over the term students may run into problems during their oral examination (Kolloquium) as we discuss questions concerning topics from the lectures as well as from the assignment; questions about the assignment are based on the assignment solution programmed by the students.</p>	

<p><b>Modul DSG-DistrSys-M Distributed Systems</b> <i>Distributed Systems</i></p>	<p>6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium</p>
<p>(seit SS20) Modulverantwortliche/r: Prof. Dr. Guido Wirtz</p>	
<p><b>Inhalte:</b> Nowadays infrastructure and business relies more or less on distributed systems of various flavors. Most of our civilization would not work any more if all distributed systems would fail. So, that should be a good reason for anyone planning to work in the context of IT to be fluent with the characteristics and conceptual as well as technical issues of such systems. The course discusses and compares the different flavors of and problems with distributed systems, presents solutions to the most common problems arising with this kind of systems and teaches general techniques and architectures that are essential to make distributed systems work. These aspects are discussed with a focus on both, conceptual (using selected research papers) as well as practical, technical and algorithmic topics. Additionally, the course also addresses state-of-the-art techniques how to build distributed systems in practice using Java-based technologies like process interaction, synchronization, remote procedure calls and web service infrastructure as well as basic cloud techniques. Students are required to work (in groups) on an assignment using different technologies in order to combine the theoretical concepts with practical experience and ... Yes, we program!</p>	
<p><b>Lernziele/Kompetenzen:</b> Students know about the characteristics and different flavors of distributed systems, understand the essential differences compared to monolithic, centralized systems as well as their consequences when designing, architecting and building distributed systems. Students are able to apply the common algorithmic techniques and programming paradigms in order to build distributed systems themselves. Students know about the different architectural approaches to build and manage small- to medium-scale distributed systems and are capable of making well-founded architectural decisions for different use-cases. Students have gained experience with practically building and running distributed systems using state-of-the-art technologies.</p>	
<p><b>Sonstige Informationen:</b> The language of instruction in this course is English.  The overall workload of 180h for this module consists of:</p> <ul style="list-style-type: none"> <li>• weekly classes: 22.5h</li> <li>• tutorials: 7.5h</li> <li>• Work on assignment: 90h</li> <li>• Literature study 40h</li> <li>• preparation for and time of the final exam: 20h</li> </ul> <p>This course is intended for master students which have not enrolled in a similar course during their bachelor studies. In case of questions don't hesitate to contact the person responsible for this module.</p>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b> Knowledge of the basics of computer science in general, esp. operating systems, as well as practical experience in sequential and</p>	<p><b>Besondere Bestehensvoraussetzungen:</b> keine</p>

parallel Java programming, esp. knowledge about multithreading and synchronization like, e.g., the subject-matters of DSG-PKS-B.		
Modul Programmierung komplexer interagierender Systeme (DSG-PKS-B) - empfohlen		
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>1. Lecture Distributed Systems</b></p> <p><b>Lehrformen:</b> Vorlesung</p> <p><b>Dozenten:</b> Prof. Dr. Guido Wirtz</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b> c.f. module description</p> <hr/> <p><b>Inhalte:</b> c.f. module description</p> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair: Distributed Systems - Concepts and Design. Pearson Education UK, 2011 (5. edition);</li> <li>• Andrew Tanenbaum, Marten van Steen: Distributed Systems - Principles and Paradigms, 2017 (3rd edition)</li> <li>• Additional research literature will be provided during the term for selected readings and discussions</li> </ul>	<b>2,00 SWS</b>
<p><b>2. Tutorial Distributed Systems</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Mitarbeiter Praktische Informatik</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b> c.f. module description</p> <hr/> <p><b>Inhalte:</b> Introduction to and discussion of tools and practical issues closely related to the topics discussed in the lecture as well as solutions of problems that come up during working on the practical assignment.</p>	<b>2,00 SWS</b>

<p><b>Prüfung</b></p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 15 Minuten</p> <p>Bearbeitungsfrist: 3 Monate</p> <p><b>Beschreibung:</b></p> <p>Oral examination concerning the topics discussed in the lecture, exercises and assignment. <b>Students may choose English or German as the language for the written assignment and oral examination.</b> Examinations will take place</p>	
--	--

at the end of the summer term or at the begin of the winter term (students may choose one of them).

Students are assumed to work on an advanced programming assignment ('schriftliche Hausarbeit') during the semester that is introduced at the beginning of the semester and uses the most important technologies discussed during the semester.

**Note:** Without working on the programming assignment over the term students may run into problems during their oral examination (Kolloquium) as we discuss questions concerning topics from the lectures as well as from the assignment; questions about the assignment are based on the assignment solution programmed by the students.

<b>Modul DSG-Project-M Masterprojekt Verteilte Systeme</b>		9 ECTS / 270 h
<i>Master Project Distributed Systems</i>		
(seit SS19) Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
<b>Inhalte:</b> Im Rahmen des Projekts wird ein umfangreiches Thema im Bereich der Systementwicklung in Gruppenarbeit bearbeitet. Studierende arbeiten dabei an auch in der Praxis auftretenden Problematiken, die mit der möglichst selbständigen Lösung einer größeren, nur noch bedingt von einem Einzelnen lösbaren Aufgabe in zum Teil konkret vorgegebenen Rahmenbedingungen verbunden sind. Das Master-Projekt unterscheidet sich dabei von der Projektarbeit im Bachelorstudiengang durch die Komplexität der Aufgabe und den direkten Bezug zu aktuellen wissenschaftlichen Arbeiten des Lehrstuhls.		
<b>Lernziele/Kompetenzen:</b> Fähigkeit zur selbständigen Erarbeitung von Problemlösungen 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>Sonstige Informationen:</b> Verfassen eines schriftlichen Projektberichts, eines Posters zum Projektergebnis sowie Bestehen des mündlichen Testats zum Projektergebnis.  Der Arbeitsaufwand von insgesamt 270 Std. (als Block nach dem jeweiligen SoSe) gliedert sich in etwa in: <ul style="list-style-type: none"> <li>• 35 Std. Einführung, Vorstellen von Werkzeugen, Kurzvorträge</li> <li>• 30 Std. Recherchen zu und Einarbeitung in Thematik des Projekts inkl. Vorbereiten von Kurzvorträgen</li> <li>• 180 Std. praktische Projektarbeit (Softwareentwicklung)</li> <li>• 15 Std. Abfassen des Projektberichts und Erstellen des gemeinsamen Posters</li> <li>• 10 Std. Vorbereitung auf das Testat (unter o.g. schon erbrachten Aufwänden)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Die Veranstaltung baut auf der Veranstaltung DSG-EiDistrSys 'Einführung in verteilte Systeme' auf. Je nach Themenstellung ist auch der vorherige Besuch einer der Veranstaltungen DSG-SOA-M oder DSG-DSAM-M zu empfehlen (Bekanntgabe jeweils bei Themenankündigung).  Von den Studierenden des Faches wird die Beherrschung einer höheren (objektorientierten) Programmiersprache sowie die Bereitschaft zur praktischen Arbeit am Rechner erwartet.  Modul Einführung in Verteilte Systeme (DSG-EiDistrSys) -		<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>Verteilte Systeme</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik  <b>Sprache:</b> Deutsch/Englisch  <b>Angebotshäufigkeit:</b> WS, SS</p> <hr/> <p><b>Inhalte:</b>  Aufbauend auf den in den Vorlesungen und Übungen im Bereich Verteilte Systeme erworbenen Kenntnissen und Fertigkeiten wird in dieser Veranstaltung ein kleineres Projekt mit wissenschaftlichem Bezug in einer Gruppe umgesetzt. Nach einer kurzen Einführung in die jeweils verwendeten Technologien und Werkzeuge wird in einer als Projekt mit verschiedenen Arbeitsgruppen/paketen organisierten Form ein zusammenhängendes Problem aus dem Bereich der verteilten und mobilen Systeme praktisch bearbeitet. In der Regel wird dabei ein Prototyp eines komplexen verteilten Softwaresystems oder Werkzeugs aus diesem Bereich erstellt.</p> <hr/> <p><b>Literatur:</b>  - je nach Projektthema -</p>	<b>6,00 SWS</b>
<p><b>Prüfung</b>  Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten  Bearbeitungsfrist: 3 Monate  <b>Zulassungsvoraussetzung zur Modulprüfung:</b>  regelmäßige Teilnahme an der Lehrveranstaltung  <b>Beschreibung:</b>  Bericht über den im Projekt erbrachten Eigenanteil als klar gekennzeichnete Teil des Gesamtberichts der Projektgruppe; Mitarbeit bei der Erstellung einer Posterdemonstration zu den Projektergebnissen.   Mündliches Prüfungsgespräch über die Inhalte des Projekts, insbesondere die vom jeweiligen Studierenden erbrachten konzeptionellen und praktischen Leistungen.</p>	



<b>Modul DSG-SOA-M Service-Oriented Architecture and Web Services</b> <i>Service-Oriented Architecture and Web Services</i>	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit SS20) Modulverantwortliche/r: Prof. Dr. Guido Wirtz	
<p><b>Inhalte:</b></p> <p>Building enterprise-scale IT systems requires sound concepts for integrating software. <b>Service-oriented architectures (SOAs)</b> have been the number one answer to this integration challenge for years. Indeed, service orientation is and will be a cornerstone in modularizing large IT landscapes and alignment with business needs is the driving factor for service engineering. A SOA composes an IT system from services in a loosely-coupled manner. Each service implements a business task and therefore have a clear value attribution. When business needs change, the loose coupling of services allows for quick adjustment of the SOA. In recent years, Microservices have been put forward as a new paradigm for organizing software-intensive systems as a set of small services that communicate using lightweight communication technologies and are <i>independently deployable by fully automated deployment machinery</i>. Conceptually, Microservices and SOA share a lot, but the Microservices paradigm puts a lot more emphasis on automation in development and therefore is a better fit for modern development practices.</p> <p>When moving beyond company boundaries and opening up the solution space is necessary, <b>software ecosystems (SECOs)</b> come into play. Software ecosystems integrate software contributions from independent organizational entities and enable software products and solutions that a single company cannot realize alone. Prominent representatives of software ecosystems are Android and the Playstore or iOS and the AppStore. But the paradigm of software ecosystems goes far beyond mobile platforms and also covers application areas in the cloud domain or the embedded domain.</p> <p>Skilled software architects therefore reconcile the business views and technical views for the benefit of the enterprise and therefore need both, advanced knowledge in business process and workflow management as well as a rock-solid understanding of service engineering and distributed computing.</p> <p>This course will introduce you to the world of architectures for large-scale software by giving a brief overview on distributed systems and software architecture in general. Then SOAs as an architectural paradigm and Web Services (WSDL + REST) as SOA implementation technology will be treated in detail. SOA will be contrasted to Microservices and the development aspects that Microservices focuses on will be discussed. Software ecosystems then will be introduced as a paradigm for organizing software systems and container technology (Linux Containers (LXC) and Docker) as a frequent implementation means for software ecosystems will be introduced. In particular, we will investigate what building industry-grade ecosystems based on container technology means in practice.</p> <ul style="list-style-type: none"> <li>• Conceptual Foundations of SOA</li> <li>• SOA Characteristics</li> <li>• Microservices</li> <li>• WSDL and Basic Web Services</li> <li>• REST-ful Services</li> <li>• Software Ecosystems</li> <li>• Container technology</li> </ul> <p>The selection of topics and teaching method of this course reflects the Distributed Systems Group's (DSG) dedication to integrate business and IT, theory and practice, research and teaching. You not only will be taught the classical way, but you will have hands-on experience on service development and SOA tools.</p>	

Also, you will get a grasp of current services research and you will get the chance to discuss selected publications with your lecturers.

**Lernziele/Kompetenzen:**

Students know about the different aspects of service-oriented architectures and their practical use.

Students

- Understand the characteristics of SOAs, Microservices and SECOs and its implications on IT systems.
- Know relevant technologies and standards in the field and being able to combine some of these to develop basic Web Services and service compositions
- Being able to compare WSDL Web Services to REST Web Services
- Being able to use container technology for integrating software
- Being able to judge IT architectures from a SOA/Microservices/SECO perspective.
- Being able to understand and discuss scientific work in the area

**Sonstige Informationen:**

The main language of instruction in this course is English.

The overall workload of 180h for this module consists of:

- weekly classes: 22.5h
- tutorials: 22.5h
- Work on the assignment: 75h
- Literature study 30h
- preparation for and time of final exam: 30h

**Zulassungsvoraussetzung für die Belegung des Moduls:**

Basic knowledge in software engineering and in distributed systems as introduced, e.g., in the modules DSG-IDistrSys-B or DSG-DistrSys-M.

**Empfohlene Vorkenntnisse:**

Basic knowledge in software engineering and in distributed systems as introduced, e.g., in the modules DSG-IDistrSys-B or DSG-DistrSys-M.

Modul Introduction to Distributed Systems (DSG-IDistrSys) - empfohlen

**Besondere**

**Bestehensvoraussetzungen:**

keine

**Angebotshäufigkeit:** SS, jährlich

**Empfohlenes Fachsemester:**

**Minimale Dauer des Moduls:**

1 Semester

**Lehrveranstaltungen**

**1. Service-Oriented Architecture and Web Services**

**2,00 SWS**

**Lehrformen:** Vorlesung

**Dozenten:** Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik

**Sprache:** Englisch

**Angebotshäufigkeit:** SS, jährlich

**Lernziele:**

cf. module description

**Inhalte:**

cf. module description

**Literatur:**

<p>SOA and SECOS are still fast emerging fields - most recent version of standards and up-to-date literature will be provided at the beginning of each course.</p>	
<p><b>2. Service-Oriented Architecture and Web Services</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Mitarbeiter Praktische Informatik  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b>  c.f. overall module description</p> <hr/> <p><b>Inhalte:</b>  Introduction to and discussion of tools and practical issues closely related to the topics discussed in the lecture as well as solutions of problems that come up during working on the practical assignment.</p> <hr/> <p><b>Literatur:</b>  (see lecture)</p>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b>  Hausarbeit mit Kolloquium / Prüfungsdauer: 15 Minuten  Bearbeitungsfrist: 3 Monate</p> <p><b>Beschreibung:</b>  Oral examination concerning the topics discussed in the lecture, exercises and assignment. <b>Students may choose English or German as the language for the oral examination.</b> Examinations will take place at the end of the summer term or at the begin of the winter term (students may choose one of them).  Students are assumed to work on a programming assignment ('schriftliche Hausarbeit') during the semester that is introduced at the beginning of the semester and uses the most important technologies discussed during the semester.</p> <p><b>Note:</b> Without working on the programming assignment over the term students may run into problems during their oral examination (Kolloquium) as we discuss questions concerning topics from the lectures as well as from the assignment; questions about the assignment are based on the assignment solution programmed by the students.</p>	

<b>Modul DSG-SRDS-M Selected Readings in Distributed Systems</b> <i>Selected Readings in Distributed Systems</i>		3 ECTS / 90 h 23 h Präsenzzeit 67 h Selbststudium
(seit WS18/19) Modulverantwortliche/r: Prof. Dr. Guido Wirtz		
<b>Inhalte:</b> This module is intended to offer an in-depth study of specific topics in distributed systems that go well beyond the topics discussed in DSG-IDistrSys, DSG-SOA-M or DSG-DSM-M. We try to close the gap between 'standard' lecture topics often dealing with the (required) basics and the state-of-the-art related to a specific research question regarding distributed systems in general, complex systems architecture, SOC and SOA, server-side middleware, process languages, as well as questions w.r.t. standard conformance, interoperability and correctness based on 'ground-breaking' as well as up-to-date research papers from international journals and/or conferences.		
<b>Lernziele/Kompetenzen:</b> Students will learn how to read and work on recent research papers and how to present their essence as an outline talk to colleagues (students). Students will be able to classify and compare results from papers in the context of a specific research question. Moreover, students will become proficient in the developments of the specialized research area that is the topic of the particular course.		
<b>Sonstige Informationen:</b> The main language of instruction in this course is English.  The overall work load for the course is 90 hours: <ul style="list-style-type: none"> <li>• 22.5 h classes</li> <li>• 55 h work on assigned readings, essay and presentations</li> <li>• 12.5 h preparation for and time of final exam</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge about distributed systems as offered, e.g., by the course DSG-IDistrSys or similar knowledge. Dependend on the topic of the specific course, additional knowledge as discussed in DSG-SOA-M or DSG-DSAM-M will be required (ask if in doubt before enrolling in the course)  Modul Introduction to Distributed Systems (DSG-IDistrSys) - 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>Selected Readings in Distributed Systems</b> <b>Lehrformen:</b> Vorlesung/Seminar <b>Dozenten:</b> Prof. Dr. Guido Wirtz, Mitarbeiter Praktische Informatik <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, SS		<b>2,00 SWS</b>

**Inhalte:**

The course discusses recent topics and research questions concerning distributed systems and related areas like, e.g.,

- Components and Component systems (SoSe 2010)
- Service Engineering Challenges in a B2Bi world (WiSe 2010/2011)
- Visual Process Description Languages (SoSe 2011)
- Services, SOA and Orchestrations - State of the Art (SoSe 2012, SoSe 2013)
- Services, SOA, Process Languages, ... Clouds - State of the Art (SoSe 2014)
- Enterprise Architecture from EAI to the IoT (WS 2016/2017)

This module is intended to offer an in-depth study of specific topics in distributed systems that go well beyond the topics discussed in DSG-IDistrSys, DSG-SOA-M or DSG-DSAM-M. We try to close the gap between 'standard' lecture topics often dealing with the (required) basics and the state-of-the-art related to a specific research question regarding distributed systems, SOA, middleware and so on.

**Literatur:**

As the concrete topics change each semester, pointers to literature are given during the preparation of each specific course using the vc-uni-bamberg.de learning platform.

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten

Bearbeitungsfrist: 4 Monate

**Beschreibung:**

Oral examination about the topics discussed during the term with a special emphasis on those topics, the examinee has presented during the course in her or his short presentations or their essay. The language for the oral examination is English.

Students are assumed to read a set of papers during the semester which are introduced at the beginning of the semester and present the content of at least two papers in a short outline talk (10 minutes maximum) as basis for the discussion among the participants during class. Additionally, each student writes an essay (8 pages) that describes the essentials of one of the research topics discussed during class and relates this topic to the overall theme of the selected readings course.

<b>Modul EESYS-DAE-M Data Analytics in der Energieinformatik</b> <i>Data Analytics in Energy Informatics</i>	6 ECTS / 180 h
(seit WS19/20) Modulverantwortliche/r: Prof. Dr. Thorsten Staake	
<p><b>Inhalte:</b></p> <p>Der Kurs vermittelt theoretische Grundlagen und praktische Fertigkeiten in den Bereichen Data Analytics und maschinelles Lernen. Zudem erfolgt eine Einführung in die Statistik-Software GNU R. Dabei helfen konkrete Beispiele und reale Datensätzen aus den Themenfeldern Energie, Umwelt und Konsumentenverhalten, um die Kursteilnehmerinnen und Kursteilnehmer auf die Herausforderungen in der Praxis und in der angewandten Forschung vorzubereiten.</p> <p>Nach einer Zusammenfassung ausgewählter Statistik-Grundlagen umfasst der Kurs</p> <ul style="list-style-type: none"> <li>• eine Einführung in die Statistik-Software GNU R,</li> <li>• den Entwurf von Feldexperimenten und die Erhebung von Verhaltensdaten mit Informationssystemen,</li> <li>• Verfahren der linearen und logistischen Regressionen,</li> <li>• Verfahren der Cluster-Analyse,</li> <li>• Techniken aus dem Bereich des Maschinellen Lernens einschließlich KNN, Regressionen und Support-Vektor-Maschinen und</li> <li>• Aspekte der Ethik und des Datenschutzes bei fortgeschrittenen Datenerhebungs- und analyseverfahren.</li> </ul>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Nach erfolgreichem Abschluss des Kurses können die Teilnehmerinnen und Teilnehmer</p> <ul style="list-style-type: none"> <li>• neue Praxis- und Forschungsfragen, die sich mit empirischen Methoden ergründen lassen, in Experimente übersetzen,</li> <li>• entsprechende Experimente planen und durchführen,</li> <li>• aus den in der Veranstaltung vorgestellten Methoden geeignete auswählen, um erhobene Daten korrekt zu analysieren,</li> <li>• ihre Methodenwahl begründen und die Analyseschritte erklären,</li> <li>• Analysen in R korrekt und effizient umsetzen,</li> <li>• die Methoden, sofern erforderlich, für spezifische Probleme theoretisch fundiert anpassen,</li> <li>• die Ergebnisse eigener Analysen und Analysen Dritter interpretieren und deren Vorteile und Grenzen benennen und</li> <li>• Aspekte des Datenschutzes und der Ethik, die bei der Anwendung leistungsstarker Erhebungs- und Analyseverfahren zu beachten sind, benennen und kritisch reflektieren.</li> </ul>	
<p><b>Sonstige Informationen:</b></p> <p>Die Vorlesung wird als nichtzeitgebundene, videobasierte Online-Vorlesung durchgeführt.</p> <p>Die Übung findet wöchentlich als Präsenzveranstaltung statt.</p> <p>Die Online-Veranstaltung umfasst Videos mit Untertiteln, Lesematerial, Übungsdatensätze sowie zahlreiche Online- und Offline-Aufgaben. Ein Online-Diskussionsforum ist verfügbar.</p> <p>Die Online-Vorlesung wird durch drei Präsenzveranstaltungen (zusätzlich zu den Präsenz-Übungen) unterstützt:</p>	

1. Präsenzveranstaltung als Einführungsveranstaltung. Hier werden der genaue Ablauf erläutert und Zugangsinformationen zu den Online-Ressourcen bekanntgegeben. Termin: Erste Vorlesungswoche.
2. Präsenzveranstaltung zur Vertiefung der Inhalte und zur eigenen Fortschrittskontrolle. Termin: Wird in der ersten Vorlesungswoche bekanntgegeben.
3. Präsenzveranstaltung zur Prüfungsvorbereitung und für Fragen & Antworten zur Klausur. Termin: Letzte Vorlesungswoche.

Eine Einführung in die Statistik-Software GNU R findet im Rahmen der ersten Tutorien statt.

**Zulassungsvoraussetzung für die Belegung des Moduls:**

keine

**Empfohlene Vorkenntnisse:**

Grundlegende Statistik-Kenntnisse (z.B. aus dem Bachelor-Studium). Eine Wiederholung der Statistik-Grundlagen ist Teil des zur Verfügung gestellten Online-Materials sowie der ersten Übungen und sollte, wenn erforderlich, durch ein Selbststudium ergänzt werden.

Grundlegende Kenntnisse einer Programmiersprache.

**Besondere**

**Bestehensvoraussetzungen:**

keine

**Angebotshäufigkeit:** WS, jährlich

**Empfohlenes Fachsemester:**

**Minimale Dauer des Moduls:**

1 Semester

**Lehrveranstaltungen**

**1. Data Analytics in der Energieinformatik**

**2,00 SWS**

**Lehrformen:** Vorlesung

**Dozenten:** Prof. Dr. Thorsten Staake

**Sprache:** Deutsch/Englisch

**Angebotshäufigkeit:** WS, jährlich

**Inhalte:**

Die videobasierte Online-Vorlesung ist in zwei Teile untergliedert. Teil 1 umfasst eine Wiederholung und Vertiefung der für das Modul erforderlichen Grundlagen aus der Statistik. Teil 2 behandelt die im Abschnitt „Modul EESYS-DAE-M“ unter „Inhalte“ genannten Themen sowie die den verwendeten Konzepten zugrundeliegenden Theorien. Sowohl Teil 1 als auch Teil 2 nutzen reale Daten sowie aktuelle Beispiele und Aufgaben aus der Unternehmenspraxis und der aktuellen Forschung (insbes. aus den Bereichen Energie, Nachhaltigkeit und Konsumentenverhalten) zur Verdeutlichung der Konzepte und zur Vorbereitung auf neue Problemstellungen und Anwendungsfälle. Zahlreiche Aufgabenstellungen werden in GNU R gelöst.

**Literatur:**

Weiterführende Unterlagen werden in der Veranstaltung bekanntgegeben.

**2. Data Analytics in der Energieinformatik**

**2,00 SWS**

**Lehrformen:** Übung

**Sprache:** Deutsch/Englisch

**Angebotshäufigkeit:** WS, jährlich

**Inhalte:**

In der Übung werden die in der Vorlesung behandelten Inhalte auf exemplarische Praxisprobleme angewendet, auf neue Fragestellungen übertragen und kritisch diskutiert. Besonderen Raum nehmen kleinere Fallstudien und die Analyse von Datensätzen ein. Es sind Aufgaben mit der Statistik-Software GNU R zu lösen. Dies erfolgt in Teilen in Einzelarbeit und in Teilen in Kleingruppen. Die Übung transportiert auch vereinzelt neue Inhalte, insbesondere, wenn eine enge Verknüpfung mit deren Anwendung didaktisch sinnvoll ist. In einzelnen Übungen findet eine freiwillige, selbst zu korrigierende Lernfortschrittskontrolle statt. Zu Beginn findet eine Einführung in GNU R statt.

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Beschreibung:**

Durch die freiwillige Abgabe von semesterbegleitenden Studienleistungen können Punkte zur Notenverbesserung gesammelt werden, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus Studienleistungen bestanden ist. Zu Beginn der Lehrveranstaltung wird bekannt gegeben, ob Studienleistungen angeboten werden. Falls Studienleistungen angeboten werden, wird zu diesem Zeitpunkt auch die Anzahl, die Art, der Umfang und die Bearbeitungsdauer der Studienleistungen sowie die Anzahl an erreichbaren Punkten pro Studienleistung und in der Modulprüfung bekannt gegeben. Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.



<b>Modul EESYS-ES-M Energieeffiziente Systeme</b> <i>Energy Efficient Systems</i>	6 ECTS / 180 h
(seit WS19/20) Modulverantwortliche/r: Prof. Dr. Thorsten Staake	
<p><b>Inhalte:</b></p> <p>Die Veranstaltung behandelt die Gestaltung und den Einsatz von Informationssystemen zur Steigerung der Energie- und Ressourceneffizienz und zur Reduktion von Treibhausgas-Emissionen. Sie richtet sich insbesondere an Studierende der Informatik und Wirtschaftsinformatik, die mit ihren Kompetenzen Anwendungsfelder in den Bereichen Energie, Mobilität, Produktion und nachhaltiger Konsum bewerten und erschließen möchten.</p> <p>Die Veranstaltung stellt dazu Methoden und Theorien aus den Bereichen der Verhaltensökonomie, des Operations-Managements und der Simulation vor, die dabei helfen, umweltrelevante Sachverhalte, Prozesse und Verhaltensweisen zu bewerten und zu beeinflussen. Die Themen werden ergänzt durch Kosten-/Nutzen-Betrachtungen auf Mikro- und Makro-Ebene (einschließlich Rebound-Effekte) und einer Diskussion der wirtschaftlichen und gesellschaftlichen Implikationen der vorgestellten Ansätze.</p> <p>Zu Beginn der Veranstaltung werden einfache physikalische und energietechnische Grundlagen vermittelt, sodass Studierenden ohne Vorkenntnisse im Bereich Energie eine erfolgreiche Kursteilnahme ermöglicht wird.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Das Modul soll die Teilnehmenden dazu befähigen,</p> <ul style="list-style-type: none"> <li>• die im Kurs vorgestellten physikalischen und technischen Grundprinzipien, die für ein Verständnis der behandelten Themen erforderlich ist, zu erklären und in Analysen anzuwenden,</li> <li>• die zentralen Komponenten, Stellgrößen, Anforderungen und Herausforderungen der Elektromobilität zu erläutern und darzulegen, welchen Beitrag Informationssysteme zur Lösung der Herausforderungen leisten können; darüber hinaus sollen Teilnehmende in der Lage sein, datenbasierte Simulationen zu wichtigen Kenngrößen von Elektrofahrzeugen (z.B. Erreichbarkeit, Einfluss auf lokale Stromnetze) aufzubauen,</li> <li>• die Möglichkeiten von Informationssystemen zur Reduktion des Energieverbrauchs im Bereich Raumklima/Heizung darzulegen und konzeptionell sowie datenbasiert zu bewerten,</li> <li>• die Eigenschaften von Umweltinformationssystemen und deren Wirkmechanismen detailliert zu erläutern,</li> <li>• die vorgestellten Theorien (z.B. die Prospect Theory) und Konzepte aus der Verhaltensökonomie zu erklären, auf Informationssysteme zur Entscheidungsunterstützung und -beeinflussung zu übertragen, entsprechende Systeme zu konzipieren und deren Wirkung zu bewerten, und</li> <li>• die Effekte der Maßnahmen zu bewerten (direkte Effekte ebenso wie indirekte und Makroökonomische Effekte) bzw. Bewertungsansätze kritisch zu diskutieren.</li> </ul> <p>Darüber hinaus soll das Modul die Teilnehmenden dazu befähigen, die im Kurs erworbenen Fähigkeiten auch in neuen Situationen anzuwenden und geeignet anzupassen und zu erweitern.</p> <p>Schlussendlich sollen Studierende ihre Gestaltungsmöglichkeiten, die sich aus ihrem IT-Studium im Bereich der Nachhaltigkeit ergeben, erkennen und umsetzen können.</p>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine</p>	

<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>	
<p><b>1. Energieeffiziente Systeme</b>  <b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Dr. Thorsten Staake  <b>Sprache:</b> Deutsch/Englisch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>                      Die Vorlesung behandelt die im Abschnitt „Modul EESYS-ES-M“ unter „Inhalte“ genannten Themen. Die Erarbeitung der Kompetenzen wird durch Lehrvorträge und Diskussionen unterstützt. Besonderen Raum nehmen Fallstudien und die Analyse von Fachbeiträgen ein. Methoden und Konzepte werden regelmäßig anhand praktischer Beispiele eingeführt und in Beispielaufgaben angewendet. Für einzelne Themen enthält die Vorlesung „Flipped-Classroom-Elemente“, bei denen erwartet wird, dass sich die Studierenden mit dem Lesen von Fachbeiträgen auf eine Veranstaltung vorbereiten, in der die Inhalte dann reflektiert und erweitert werden.</p> <hr/> <p><b>Literatur:</b>                      Weiterführende Unterlagen werden in der Veranstaltung bekanntgegeben.</p>	<b>2,00 SWS</b>
<p><b>2. Energieeffiziente Systeme</b>  <b>Lehrformen:</b> Übung  <b>Sprache:</b> Deutsch/Englisch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>                      In den ersten Übungsveranstaltungen werden die erforderlichen physikalischen und technischen Grundlagen zusammengefasst, um einen direkten Einstieg auch ohne energiespezifische Vorkenntnisse zu ermöglichen. Dazu behandelt die Übung insbesondere Grundbegriffe der Energietechnik und der Elektrotechnik. Darauf aufbauend werden die in der Vorlesung behandelten Inhalte auf exemplarische Praxisprobleme angewendet, auf neue Fragestellungen übertragen und kritisch diskutiert. Übungen umfassen auch Analysen von Fachbeiträgen und kleinere Fallstudien. Die Bearbeitung erfolgt in Teilen in Einzelarbeit und in Teilen in Kleingruppen.                      Die Übung transportiert auch vereinzelt neue Inhalte, insbesondere, wenn eine enge Verknüpfung mit deren Anwendung didaktisch sinnvoll ist. In einzelnen Übungen findet eine freiwillige, selbst zu korrigierende Lernfortschrittskontrolle statt.</p>	<b>2,00 SWS</b>

<b>Prüfung</b>	
----------------	--

---

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Beschreibung:**

Durch die freiwillige Abgabe von semesterbegleitenden Studienleistungen können Punkte zur Notenverbesserung gesammelt werden, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus Studienleistungen bestanden ist. Zu Beginn der Lehrveranstaltung wird bekannt gegeben, ob Studienleistungen angeboten werden. Falls Studienleistungen angeboten werden, wird zu diesem Zeitpunkt auch die Anzahl, die Art, der Umfang und die Bearbeitungsdauer der Studienleistungen sowie die Anzahl an erreichbaren Punkten pro Studienleistung und in der Modulprüfung bekannt gegeben. Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.

<b>Modul Gdl-AFP-M Advanced Functional Programming</b>		6 ECTS / 180 h
<i>Advanced Functional Programming</i>		
(seit WS19/20)		
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b>		
Die Veranstaltung beschäftigt sich vertiefend mit ausgewählten fortgeschrittenen Konzepten der funktionalen Programmierung. Sie setzt Grundkenntnisse der Funktionalen Programmierung voraus, die zum Beispiel durch die vorherige Teilnahme an der einführenden Veranstaltung Gdl-IFP erworben wurden.		
<b>Lernziele/Kompetenzen:</b>		
Aufbauend auf einem elementaren und anwendungsorientierten Vorwissen des Bachelorstudiums erwerben die Studierenden ein breites und wesentlich vertieftes Verständnis der Besonderheiten der funktionalen Programmierung als nichtdeklaratives Programmierprinzip. Sie können existierende Sprachkonzepte analysieren, kritisch bewerten und daraus eigenständig neue Ideen für die Weiterentwicklung in Forschungs- und Anwendungskontexten ableiten.		
Die Veranstaltung trainiert die Fähigkeit zur selbständigen Entwicklung und Evaluierung neuer funktionaler Modellierungsmethoden unter Einsatz von fortgeschrittenen Strukturierungskonzepten, wie etwa polymorphe Typsystemen erster und höherer Ordnung in der statischen Spezifikation von Programmen, Monaden und Koroutinen (continuation passing), Induktive und Koinduktive Datentypen; Konstruktion von Übersetzungsverfahren für Synchronen Programmierung sowohl als Datenfluss- als auch Kontrollflussparadigma; Beherrschung und Evaluierung komplexer Konzepte der nebenläufigen Programmierung in Haskell aus der aktuellen Forschung.		
<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 ggf. Teilnahme an Rechnerübungen: 45 Stunden</li> <li>• Prüfungsvorbereitung und schriftliche Prüfung: 30 Stunden</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
gundlegende Kenntnisse in funktionaler Programmierung, z.B. aus dem Modul Gdl-IFP-B, gute Englischkenntnisse		keine
Modul Introduction to Functional Programming (Gdl-IFP) - empfohlen		
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester
<b>Lehrveranstaltungen</b>		
<b>1. Advanced Functional Programming</b>		<b>2,00 SWS</b>
<b>Lehrformen:</b> Vorlesung		
<b>Dozenten:</b> Prof. Ph.D. Michael Mendler		

<p><b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> SS, 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>• S. Marlow: The Haskell 2010 Language Report. <a href="https://www.haskell.org/onlinereport/haskell2010/">https://www.haskell.org/onlinereport/haskell2010/</a></li> <li>• V. Zsó�, Z. Horváth, R. Plasmeijer: Central European Functional Programming School. Springer 2012.</li> <li>• S. Marlow: Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming, O'Reilly 2013.</li> <li>• B. O'Sullivan, J. Goerzen, D. Stewart: Real World Haskell. O'Reilly 2009.</li> <li>• Ch. Okasaki: Purely Functional Data Structures, CUP 1998</li> <li>• F. Rabhi, G. Lapalme: Algorithms - A Functional Approach.</li> <li>• D. Syme, A. Granicz, A. Cisternino: Expert F#4.0, Apress 2015.</li> <li>• B. Pierce: Types and Programming Languages. MIT Press 2002. (esp. Chapters 23+25)</li> <li>• H. Barendregt, W. Dekkers, R. Statman: Lambda Calculus with Types. CUP 2013.</li> </ul>	
<p><b>2. Advanced Functional Programming</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Ph.D. Michael Mender  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> SS, 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>	<b>2,00 SWS</b>
<p><b>Prüfung</b>  schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten  <b>Beschreibung:</b>  Die Prüfungssprache ist Englisch.</p>	

<p><b>Modul Gdl-CSNL-M Computational Semantics of Natural Language</b></p> <p><i>Computational Semantics of Natural Language</i></p>	<p>6 ECTS / 180 h</p>
<p>(seit SS19)</p> <p>Modulverantwortliche/r: Prof. Ph.D. Michael Mendler</p> <p>Weitere Verantwortliche: Luke Burke</p>	
<p><b>Inhalte:</b></p> <p>The formal study of natural language syntax and semantics has developed as a very lively sub-field of linguistics in the past 50 years, with the typed lambda calculus in particular providing a way of giving compositional analyses of meanings in natural language. Recently, monads and continuations have been employed as tools in natural language syntax and semantics. The aim of this module is to introduce the use of monads and continuations in natural language semantics and to discuss different approaches to the formal representation of quantifier scope ambiguities in natural language. The basics of natural language semantics (typed lambda calculus) will be briefly introduced, before discussing a continuation-based approach to quantification in natural language, which will be contrasted with other approaches. Monads representing focus, intensionality and non-determinism in natural language will be discussed. We will look at how analyses of the meaning of sentences can be represented in Haskell.</p> <p>Importantly, the course may differ slightly from other courses in that assessment will not concentrate on technical exercises; rather, we require careful reading and dissection of relevant literature on the topic, since the primary mode of assessment will be via seminar presentations and essays, and you will be assessed on your understanding of, and your independent analysis of, relevant literature discussed in lectures. Independent reading of this literature will in fact be essential.</p> <p>This course may also be of interest to students in philosophy and linguistics.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>At the end of this course students should be familiar with different approaches to the formal representation of quantifier scope ambiguities in natural language; be familiar with how monads and continuations have been used in natural language semantics; be familiar with the use of Haskell to formalise analyses in natural language semantics; be able to produce and manipulate terms of the typed lambda calculus to represent how meanings combine; have an understanding of how both logics and trees have been used to represent natural language syntax; be acquainted with logics such as Montague's "Intensional Logic" and Gallin's Ty2.</p>	
<p><b>Sonstige Informationen:</b></p> <p>The workload for this module consists of:</p> <ul style="list-style-type: none"> <li>• participation in lectures and seminar sessions: 45 Stunden</li> <li>• individual preparation and reading: 45 Stunden</li> <li>• essay writing and preparation of presentation: 60 Stunden</li> <li>• exam preparation and oral exam: 30 Stunden</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p> <p>keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b></p> <p>Willingness to read relevant literature, critically discuss and analyse it and write about it. Basic logic (Gdl-Mfl-1: Mathematik für Informatik or an equivalent level of understanding). Some knowledge of</p>	<p><b>Besondere Bestehensvoraussetzungen:</b></p> <p>English language skills at Level B2 (UniCert II) or above.</p>

<p>modal logic more basic than that required for (Gdl-MTL: Modal and Temporal Logic). Knowledge of the typed lambda calculus (abstraction and application) and elementary Haskell (Gdl-IFP: Introduction to Functional Programming) would be very useful, though not essential.</p>		
<p><b>Angebotshäufigkeit:</b> SS, jährlich</p>	<p><b>Empfohlenes Fachsemester:</b></p>	<p><b>Minimale Dauer des Moduls:</b> Semester</p>

<p><b>Lehrveranstaltungen</b></p>	
<p><b>Computational Semantics of Natural Language</b>  <b>Sprache:</b> Englisch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>                  Through prepared class presentations, essay writing, and direct interactions with the students the lecturer introduces the topics of the course in detail. The seminars deepen the students' understanding of the theoretical concepts and constructions covered in the lectures through presentations, which involve comparing alternative analyses of linguistic phenomena.</p> <hr/> <p><b>Literatur:</b>                  • van Eijck, J. And Unger, Christina, "Computational Semantics with Functional Programming", Cambridge University Press 2010                  • Barker, C. and Shan, C.-C., "Continuations and natural language", Volume 53. Oxford studies in Theoretical Linguistics, Oxford University Press, 2014                  • Carpenter, Bob, "Type-Logical Semantics", MIT Press (1997)                  • Keenan, Edward, and Stabler, Edward, "Mathematical structures in Language", CSLI publications, Stanford, 2016                  • Gallin, Daniel, "Intensional and Higher-Order Modal logic. North Holland, 1975.</p>	<p><b>4,00 SWS</b></p>

<p><b>Prüfung</b>                  Portfolio  <b>Beschreibung:</b>                  The portfolio assessment consists of</p> <ul style="list-style-type: none"> <li>• written essay (4000-5000 words)</li> <li>• class presentation (30 mins)</li> <li>• final oral exam (20 min)</li> </ul> <p>Each portfolio part is graded individually. The grades are weighted as follows:                  30% for essay, 20% for class presentation, 50% for final oral exam.</p>	
--	--

<b>Modul Gdl-FP-M Functional Programming</b> <i>Functional Programming</i>	6 ECTS / 180 h
(seit WS19/20) Modulverantwortliche/r: Prof. Ph.D. Michael Mendler	
<b>Inhalte:</b> Die Veranstaltung beschäftigt sich mit den Grundlagen der funktionalen Programmierung als bedeutende Alternative 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> Aufbauend auf einem elementaren und anwendungsorientierten Vorwissen des Bachelorstudiums erwerben die Studierenden ein breites und wesentlich vertieftes Verständnis der Besonderheiten der funktionalen Programmierung als nichtdeklaratives Programmierprinzip. Sie können existierende Sprachkonzepte analysieren, kritisch bewerten und daraus eigenständig neue Ideen für die Weiterentwicklung in Forschungs- und Anwendungskontexten ableiten.  Die Veranstaltung trainiert die selbständige Entwicklung algorithmischer Problemlösungen in der funktionalen Programmiersprache Haskell sowie zur Evaluierung neuer funktionaler Modellierungsmethoden. Hierzu zählen insbesondere: Bewertung wichtiger Strukturprinzipien der funktionalen Programmierung, wie referenzielle Transparenz, Rekursion, induktive und coinduktive algebraische Datentypen, Monaden; Fähigkeit diese Strukturkonzepte auf Basis unvollständiger oder begrenzter Information adäquat in der Programmierung konkreter Aufgabenstellungen einzusetzen; Beherrschung komplexer deklarativer Modelle interaktiver Software; Kenntnis des typisierten Lambdakalküls und seiner mathematischen Eigenschaften; Fähigkeit zur kritischen Bewertung des Lambdakalküls für die semantische Fundierung und Implementierung funktionaler Programmiersprachen.	
<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 + schriftliche Prüfung (90 min.): 30 Stunden</li> </ul>	
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine	
<b>Empfohlene Vorkenntnisse:</b> grundlegende Programmierkenntnisse, gute Englischkenntnisse	<b>Besondere Bestehensvoraussetzungen:</b> keine



Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen 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. Functional Programming</b>  <b>Lehrformen:</b> Vorlesung  <b>Dozenten:</b> Prof. Ph.D. Michael Mendler  <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>• Pierce, B. C.: Types and Programming Languages, MIT Press, 2002</li> <li>• Thompson, S.: Haskell – The Craft of Functional Programming, Addison-Wesley 1999.</li> </ul>	<b>2,00 SWS</b>
<p><b>2. Functional Programming</b>  <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>Prüfung</b>  schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten  <b>Beschreibung:</b>  Die Prüfungssprache ist Englisch.</p>	<b>2,00 SWS</b>

<b>Modul HCI-MCI-M Mensch-Computer-Interaktion</b> <i>Human-Computer Interaction</i>		6 ECTS / 180 h
(seit WS17/18) Modulverantwortliche/r: Prof. Dr. Tom Gross		
<b>Inhalte:</b> Vertiefende theoretische, methodische und praktische Grundlagen der Mensch-Computer-Interaktion.		
<b>Lernziele/Kompetenzen:</b> Ziel ist die Vermittlung fundierter Kenntnisse und Fähigkeiten im Bereich der Mensch-Computer-Interaktion sowie eines breiten theoretischen und praktischen Methodenwissens zum Entwurf, zur Konzeption und zur Evaluierung interaktiver 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> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Mensch-Computer-Interaktion</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> Im Rahmen der Vorlesung werden nach einer Einführung in das Thema die folgenden Themen konzeptionell, technisch und methodisch behandelt: <ul style="list-style-type: none"> <li>• Mobile Mensch-Computer-Interaktion</li> <li>• Adaptivität und Adaptierbarkeit</li> </ul>	

<ul style="list-style-type: none"> <li>• Informationsvisualisierung</li> <li>• Tangible User Interaction</li> <li>• Usability Engineering</li> <li>• Gebrauchstauglichkeit und Ökonomie</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>• Jacko, J.A. und Sears, A., (Hrsg.). Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications. Lawrence Erlbaum, Hillsdale, NJ, 2002.</li> <li>• Hammond, J., Gross, T. und Wesson, J., (Hrsg.). Usability: Gaining a Competitive Edge. Kluwer Academic Publishers, Dordrecht, 2002.</li> </ul>	
<p><b>Prüfung</b> mündliche Prüfung / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b> In Abhängigkeit von der Teilnehmerzahl wird die Modulprüfung entweder in Form einer Klausur <b>oder</b> in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben.</p> <p>In der mündlichen Prüfung 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 mündl. Prüfung 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 mündl. Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	
<p><b>Lehrveranstaltungen</b></p>	
<p><b>Mensch-Computer-Interaktion</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</p> <hr/> <p><b>Literatur:</b> siehe Vorlesung</p>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten</p> <p><b>Beschreibung:</b> In Abhängigkeit von der Teilnehmerzahl wird die Modulprüfung</p>	

entweder in Form einer Klausur **oder** in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben.

In der Klausur über 90 min. können 90 Punkte erzielt werden.

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.

<b>Modul HCI-US-B Ubiquitäre Systeme</b> <i>Ubiquitous Systems</i>		6 ECTS / 180 h
(seit WS17/18) 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>Ubiquitäre Systeme</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Tom Gross <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> Im Rahmen der Vorlesung werden nach einer Einführung in das Thema Ubiquitous Computing - also der allgegenwärtigen Rechner, die verschwindend klein, teilweise in Alltagsgegenständen eingebaut, als Client und Server fungieren	

<p>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>Prüfung</b> mündliche Prüfung / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b> In Abhängigkeit von der Teilnehmerzahl wird die Modulprüfung entweder in Form einer Klausur <b>oder</b> in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben.</p> <p>In der mündlichen Prüfung mit einer Prüfungsdauer von 30 Minuten 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, die Punkte pro optionaler Studienleistung sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die mündl. Prüfung 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 mündl. Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.</p>	
<p><b>Lehrveranstaltungen</b></p>	
<p><b>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>	<p><b>2,00 SWS</b></p>

**Prüfung**

schriftliche Modulprüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Beschreibung:**

In Abhängigkeit von der Teilnehmerzahl wird die Modulprüfung entweder in Form einer Klausur **oder** in Form einer mündlichen Prüfung durchgeführt. Die Festlegung erfolgt zu Semesterbeginn und wird im ersten Lehrveranstaltungstermin bekannt gegeben.

In der Klausur über 90 min. können 90 Punkte erzielt werden.

Es besteht die Möglichkeit, optionale Studienleistungen zu erbringen. Diese umfassen insgesamt 12 Punkte. Die Art der optionalen Studienleistungen, die Punkte pro optionaler Studienleistung 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.

<b>Modul KTR-GIK-M Grundbausteine der Internet-Kommunikation</b> <i>Foundations of Internet Communication</i>	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS16/17) Modulverantwortliche/r: Prof. Dr. Udo Krieger	
<p><b>Inhalte:</b></p> <p>Die Lehrveranstaltung vermittelt eine Einführung in die theoretischen Grundlagen wichtiger kommunikationstechnischer Problemstellungen zu den Themengebieten Grundlagen der Internet-Kommunikation, Verbindungssegmente und Routing in IP-Netzen, Transportprotokolle in IP-Netzen bzw. fortgeschrittener Module wie Echtzeit-Kommunikation und Sicherheit in IP-Netzen und die eigenständige praktische Umsetzung des erworbenen Wissens durch vorgegebene Laborübungen zur Internet-Kommunikation in Kleingruppen. Dabei werden weitere Hilfsmittel und Anleitungen sowie die Laborumgebung bereitgestellt.</p> <p>Zur Implementierung soll ein Rechnernetz im Labor konfiguriert und getestet werden. Die Betriebssystem-Grundausrüstung und erforderliche Software-Werkzeuge wie Wireshark und Atheris werden bereitgestellt. Grundlagen der Handhabung werden von den Studierenden im Projekt selbst erarbeitet.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Wichtige Fertigkeiten zur Bewertung aktueller Kommunikationstechnologien sind nur durch die Vermittlung praktischer Fähigkeiten und Erfahrungen in teamorientierten Prozessen unter Zeit- und Zielvorgaben industrienah erlernbar. Die Studierenden werden in der Vorlesung Grundbausteine der Internet-Kommunikation und den begleitenden Laborübungen zu eigenverantwortlichem, team-orientierten Arbeiten angeleitet. Ziel ist der Erwerb praktischer Fertigkeiten auf dem Gebiet der IP-gestützten Datenkommunikation und die Fähigkeit, Lösungsvorschläge der modernen Internet-Kommunikation sicher beurteilen zu können.</p> <p>Die Lehrveranstaltung "Grundbausteine der Internet-Kommunikation" hat folgende Zielsetzungen:</p> <ul style="list-style-type: none"> <li>• Fortführung der Vorlesung Datenkommunikation des Bachelorprogrammes als Profilbildungsstudium auf Masterniveau</li> <li>• praktisches Erarbeiten der Grundlagen der Internet- und Multimedia-Kommunikation</li> <li>• Aufbau und Verkehrsanalyse von TCP/IP-basierten Rechnernetzen mit modernen Echtzeit- und Web-Anwendungen</li> <li>• Angebot einer Prüfungsalternative zur Lehrveranstaltung Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen (KTR-MMK-M) oder Mobilkommunikation (KTR-Mobi-M) im Prüfungsfach Kommunikationssysteme und Rechnernetze</li> <li>• Ergänzung der Lehrangebote in Verteilten Systemen und Medieninformatik zur Bildung eines Studienschwerpunktes "Mobile verteilte Systeme" bzw. Next Generation Internet</li> </ul> <p>Die Lehrveranstaltung ist für Bachelorstudierende im Profilbildungsstudium zur Stärkung ihrer Arbeitsmarktchancen, für Masterstudierende sowie für Austauschstudenten/innen besonders empfehlenswert.</p>	
<p><b>Sonstige Informationen:</b></p> <p>Der Arbeitsaufwand gliedert sich grob wie folgt:</p> <ul style="list-style-type: none"> <li>• Teilnahme an Präsenzveranstaltungen (Vorlesung, Laborübungen, Laborbesprechungen): 45 Stunden</li> <li>• Vorbereitung, Ausführung und Nachbereitung von Vorlesungen und Laborübungen: 100 Stunden</li> <li>• Prüfungsvorbereitung: 35 Stunden</li> </ul>	



The module can be selected by exchange students and master students speaking only English.

**Zulassungsvoraussetzung für die Belegung des Moduls:**

keine

**Empfohlene Vorkenntnisse:**

- Datenkommunikation im Umfang KTR-Datkomm-B
- Programmierkenntnisse in JAVA (oder C++)
- der Erwerb von LINUX-Kenntnissen wird empfohlen, ist aber keine Voraussetzung

Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen

Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen

Modul Algorithmen und Datenstrukturen (MI-AuD-B) - empfohlen

**Besondere**

**Bestehensvoraussetzungen:**

keine

**Angebotshäufigkeit:** SS, jährlich

**Empfohlenes Fachsemester:**

**Minimale Dauer des Moduls:**

1 Semester

**Lehrveranstaltungen**

**Grundbausteine der Internet-Kommunikation**

**Lehrformen:** Vorlesung und Übung

**Dozenten:** Prof. Dr. Udo Krieger

**Sprache:** Englisch/Deutsch

**Angebotshäufigkeit:** SS, jährlich

**4,00 SWS**

**Inhalte:**

Die Lehrveranstaltung vermittelt eine Einführung in die theoretischen Grundlagen wichtiger kommunikationstechnischer Problemstellungen zu den Themengebieten Grundlagen der Internet-Kommunikation, Verbindungssegmente und Routing in IP-Netzen, Transportprotokolle in IP-Netzen bzw. fortgeschrittener Module wie Echtzeit-Kommunikation und Sicherheit in IP-Netzen und die eigenständige praktische Umsetzung des erworbenen Wissens durch vorgegebene Laborübungen zur Internet-Kommunikation in Kleingruppen. Dabei werden weitere Hilfsmittel und Anleitungen sowie die Laborumgebung bereitgestellt. Zur Implementierung soll ein Rechnernetz im Labor konfiguriert und getestet werden. Die Betriebssystem-Grundausrüstung und erforderliche Software-Werkzeuge wie Wireshark und Atheris 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:

- Unterteilung der Arbeiten in Arbeitspakete (laboratories/work packages),
- ihre Untergliederung in Aufgaben (tasks) und Teilaufgaben (subtasks) mit Meilensteinen
- und der Darlegung von Zwischenergebnissen bzw.
- einem Abschlussbericht mit Abschlusspräsentation

Weitere Laboraufgaben zu aktuellen Forschungsfragen im "Future Generation Internet" werden bei Bedarf in die Lehrveranstaltung integriert. Details werden in der Vorlesung 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.

**Literatur:**

Grundlagen:

- J. Liebeherr, M. Elzarki: Mastering Networks, An Internet Lab Manual, Pearson Education, Boston, 2004.

weitere Literatur zu einzelnen Arbeitspaketen:

- Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2014 .
- Tanenbaum, A. S.: Computernetzwerke, Pearson Studium, München, 6. Aufl., 2013.
- Sikora, A.: Technische Grundlagen der Rechnerkommunikation, Fachbuchverlag Leipzig, 2003.
- Leon-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2nd ed. 2004.
- Badach, A.: Voice over IP - Die Technik, Carl Hanser Verlag, München, 2. Aufl., 2005.
- Flaig, G., u.a.: Internet-Telefonie, Open source Press, München, 2006.

Eine aktualisierte Liste wird in der Vorlesung bereitgestellt.

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten

Bearbeitungsfrist: 4 Monate

**Beschreibung:**

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

- Auswertung der von einem Studierenden individuell bearbeiteten Teilaufgaben, die aufgrund einer Kennzeichnung der Urheberschaft im gemeinsam erstellten schriftlichen Projektbericht im Rahmen einer Gruppenarbeit dokumentiert werden
- Vorführung und Erläuterungen der Zusammenhänge einzelner Aufgaben und Ergebnisse im Rahmen einer individuellen Kolloquiumsprüfung im Umfang von 30 Minuten

Die Bewertungsregeln dieser einzelnen Komponenten werden in der ersten Sitzung der Lehrveranstaltung bekanntgegeben. 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 KTR-MAKV-M Modellierung und Analyse von Kommunikationsnetzen und Verteilten Systemen</b> <i>Modeling and Analysis of Communication Networks and Distributed Systems</i>	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS17/18) Modulverantwortliche/r: Prof. Dr. Udo Krieger	
<p><b>Inhalte:</b></p> <p>Gegenstand der Lehrveranstaltung ist die Analyse und Leistungsbewertung komplexer verteilter Systeme, z.B. von Telekommunikationssystemen und Rechnernetzen bzw. komplexen Netzen und Cloud Computing Systemen, die als Ergebnis eines abstrakten systemtheoretischen Modelles und seiner relevanten Modellparameter durchgeführt wird. Diese Modelle dienen der Systemanalyse und Vorhersage von Leistungsmerkmalen, z.B. von Nutzungsgrad, Durchsatz, Warte-, Antwortzeiten von Nachfrage-, Personen- oder Datenflüssen in verteilten technischen Systemen oder sozialen Netzen. Solche Vorhersagen sind z.B. in wirtschaftlichen und technischen Entscheidungsprozessen der System- und Netzgestaltung eines Future Generation Internet und seiner verteilten Dienste von großer strategischer Bedeutung.</p> <p>Die Lehrveranstaltung beschäftigt sich mit der Modellierung verteilter Systeme und stellt entsprechende Beschreibungsmethoden wie Lastmodelle und Systemmodelle vor. Zur systemtheoretischen Beschreibung und Analyse dieser Modelle und ihrer Betriebsmittelverwaltungs- und -verteilungsprozesse werden anschließend elementare Methoden und Verfahren der Systemtheorie, z.B. Markov-Ketten und algebraische sowie numerischen Lösungsverfahren, bereitgestellt.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Das Hauptziel der Veranstaltung besteht in der Vermittlung von Grundkenntnissen zur Messung, Analyse und Leistungsbewertung von Rechnernetzen, modernen Kommunikationssystemen und anderen verteilten Systemen mit Hilfe systemtheoretischer Messungs-, Modellierungs- und Analysemethoden.</p> <p>Die Anwendung der vorgestellten Modelle und Methoden wird anhand von Übungsaufgaben realitätsnaher Systemausschnitte veranschaulicht. Die Studierenden sollen befähigt werden, bekannte Verfahren auf neue Sachverhalte anzuwenden.</p>	
<p><b>Sonstige Informationen:</b></p> <p>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> <p>Die vermittelten Kenntnisse aus den Modulen</p> <ul style="list-style-type: none"> <li>• Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (GDI-Mfi-1)</li> <li>• Mathematik für Informatik 2 (Lineare Algebra) (KTR-Mfi-2)</li> <li>• Methoden der Statistik I und II ( Stat-B-01, Stat-B-02)</li> </ul> <p>werden dringend empfohlen.</p>	<p><b>Besondere Bestehensvoraussetzungen:</b> keine</p>

<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester
---	----------------------------------	---

<b>Lehrveranstaltungen</b>	
<p><b>Modellierung und Analyse von Kommunikationsnetzen und Verteilten Systemen</b>  <b>Lehrformen:</b> Vorlesung und Übung  <b>Dozenten:</b> Prof. Dr. Udo Krieger  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Inhalte:</b>                      Gegenstand der Lehrveranstaltung ist die Analyse und Leistungsbewertung komplexer verteilter Systeme, z.B. von Telekommunikationssystemen und Rechnernetzen bzw. komplexen Netzen und Cloud-Computing Systemen, die als Ergebnis eines abstrakten systemtheoretischen Modelles und seiner relevanten Modellparameter durchgeführt wird. Diese Modelle dienen der Systemanalyse und Vorhersage von Leistungsmerkmalen, z.B. von Nutzungsgrad, Durchsatz, Warte-, Antwortzeiten von Nachfrage-, Personen- oder Datenflüssen in verteilten technischen Systemen oder sozialen Netzen. Solche Vorhersagen sind z.B. in wirtschaftlichen und technischen Entscheidungsprozessen der System- und Netzgestaltung eines Future Generation Internet und seiner verteilten Dienste von großer strategischer Bedeutung.                      Die Lehrveranstaltung beschäftigt sich mit der Modellierung verteilter Systeme und stellt entsprechende Beschreibungsmethoden wie Lastmodelle und Systemmodelle vor. Zur systemtheoretischen Beschreibung und Analyse dieser Modelle und ihrer Betriebsmittelverwaltungs- und -verteilungsprozesse werden anschließend elementare Methoden und Verfahren der Systemtheorie, z.B. Markov-Ketten und algebraische bzw. numerische Lösungsverfahren, bereitgestellt.                      Die Anwendung der vorgestellten Modelle und Methoden anhand realitätsnaher Systemausschnitte in den Übungen dient dem Erwerben der im heutigen industriellen Umfeld erforderlichen Kenntnisse und Fähigkeiten zur effizienten Systemanalyse, Systemmessung und Systembewertung.                      Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.</p> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• G. Bolch, S. Greiner, H. de Meer, K. S. Trivedi: Queueing Networks and Markov Chains. Wiley, 2nd ed., 2006.</li> <li>• R. Nelson: Probability, Stochastic Processes, and Queueing Theory. Springer, 1995.</li> </ul> <p>Weitere Literatur wird in der Vorlesung benannt.</p>	<p><b>4,00 SWS</b></p>
<p><b>Prüfung</b>                      mündliche Prüfung / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b></p>	

---

Die Inhalte der Vorlesung und Übung werden in Form einer mündlichen Prüfung geprüft.

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

<p><b>Modul KTR-MMK-M Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen</b>  <i>Multimedia Communication in High Speed Networks</i></p>	<p>6 ECTS / 180 h          45 h Präsenzzeit          135 h Selbststudium</p>
<p>(seit WS17/18)          Modulverantwortliche/r: Prof. Dr. Udo Krieger</p>	
<p><b>Inhalte:</b>          Ausgehend von den Grundlagen der Datenkommunikation werden in dieser weiterführenden Lehrveranstaltung des Masterprogrammes die Netzwerkarchitektur, der vermittlungstechnische Entwurf, die Protokollstrukturen, die Dienstgütearchitekturen und das Verkehrsmanagement moderner Hochgeschwindigkeitsnetze für neueste Echtzeit- und Multimedia-Anwendungen besprochen. Die zur Abwicklung derartiger Kommunikationsbeziehungen mit ihrer Zusicherung von Dienstgüte-Merkmalen erforderlichen neuen Übermittlungsarchitekturen sowie die Erweiterungen des TCP/IP-Protokollstapels werden in der Veranstaltung vorgestellt.</p> <p>Im Mittelpunkt stehen neben leistungsfähigen Anschlusstechnologien auf leitungsgebundenen Medien, Transport- und Dienstgüte-Architekturen im Kernnetz, wie Intserv, Diffserv sowie MPLS und GMPLS. Außerdem werden die Fortentwicklung des IPv4 durch IPv6 sowie die Steuerungsalgorithmen von TCP, Multipath-TCP und SCTP vorgestellt. Ferner werden die schnelle Paketvermittlung in IP-Netzen mit Dienstgüte-Unterstützung und der Einsatz bekannter Betriebsmittel- und Verkehrsmanagement-Verfahren, z.B. Speicherverwaltungsalgorithmen wie RED, RIO und Schedulingalgorithmen wie WFQ, angesprochen. Darüber hinaus erfolgt eine Darlegung der Grundprinzipien Software-definierter Netze mit der Virtualisierung von Netzfunktionen. Außerdem werden die Grundlagen Informationszentrierter Netze erläutert.</p> <p>Darüber hinaus werden typische Anwendungen des Multimedia-Internet der 3. und 4. Generation wie Webanwendungen auf Basis von HTML5 und HTTP 2.0, WebRTC, Voice-over-IP und Medien-Streaming skizziert.</p> <p>Die Vertiefung durch die Lehrveranstaltung Grundbausteine der Internet-Kommunikation mit ihren praktischen Übungen sowie die Fortführung durch Hauptseminare und Masterarbeiten ist möglich und eine wichtige Zielsetzung dieser Lehrveranstaltung.</p> <p>Die Lehrveranstaltung eignet sich zur Kombination mit entsprechenden Lehrveranstaltungen zur Architektur verteilter Systeme und Middleware von Prof. Wirtz und entsprechender Module der Medieninformatik von Prof. Henrich, z.B. Information Retrieval I/II bzw. Multimedia-Technik oder Web-Engineering, zur Gestaltung eines entsprechenden Studienschwerpunktes in Wirtschaftsinformatik, Software Systems Science oder Angewandter Informatik.</p>	
<p><b>Lernziele/Kompetenzen:</b>          Die Studierenden sollen zu eigenständigem Arbeiten befähigt werden. Es werden Grundkenntnisse der Multimediakommunikation in Hochgeschwindigkeitsnetzen und die systematische Analyse der verwendeten Algorithmen mit Hilfe eines interaktiven Übungskonzeptes vermittelt. Die Studierenden lernen, gegebene Implementierungen der vorgestellten Kommunikationsverfahren zu analysieren und durch Messungen mit Wireshark und anderen Werkzeugen ihr Leistungsverhalten zu überprüfen. Die Bearbeitung von Aufgaben im Team ist Bestandteil der Ausbildung.</p>	
<p><b>Sonstige Informationen:</b>          Der Arbeitsaufwand gliedert sich grob wie folgt:</p>	

- Teilnahme an Präsenzveranstaltungen (Vorlesung, Übung, Laborbesprechungen): 45 Stunden
- Vor- und Nachbereitung von Vorlesungen und Übungen: 100 Stunden
- Prüfungsvorbereitung: 35 Stunden

The module can be selected by exchange students and master students speaking only English.

**Zulassungsvoraussetzung für die Belegung des Moduls:**

keine

**Empfohlene Vorkenntnisse:**

- erfolgreiche Teilnahme an der Lehrveranstaltung Datenkommunikation (im Umfang von KTR-Datkomm-B) bzw. Kenntnis der spezifizierten Inhalten
- gute Kenntnisse in JAVA (oder C++)

Modul Fortgeschrittene Java Programmierung (DSG-AJP-B) - empfohlen

Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen

**Besondere**

**Bestehensvoraussetzungen:**

keine

**Angebotshäufigkeit:** SS, jährlich

**Empfohlenes Fachsemester:**

**Minimale Dauer des Moduls:**

1 Semester

**Lehrveranstaltungen**

**Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen**

**Lehrformen:** Vorlesung und Übung

**Dozenten:** Prof. Dr. Udo Krieger

**Sprache:** Englisch/Deutsch

**Angebotshäufigkeit:** SS, jährlich

**4,00 SWS**

**Inhalte:**

Ausgehend von den Grundlagen der Datenkommunikation werden in dieser weiterführenden Lehrveranstaltung des Masterprogrammes die Netzwerkarchitektur, der vermittlungstechnische Entwurf, die Protokollstrukturen, die Dienstgütearchitekturen und das Verkehrsmanagement moderner Hochgeschwindigkeitsnetze für neueste Echtzeit- und Multimedia-Anwendungen besprochen. Die zur Abwicklung derartiger Kommunikationsbeziehungen mit ihrer Zusicherung von Dienstgüte-Merkmalen erforderlichen neuen Übermittlungsarchitekturen sowie die Erweiterungen des TCP/IP-Protokollstapels werden in der Veranstaltung vorgestellt.

Im Mittelpunkt stehen neben leistungsfähigen Anschlusstechnologien auf leitungsgebundenen Medien, neue Transport- und Dienstgütearchitekturen im Kernnetz, wie Intserv, Diffserv sowie MPLS und GMPLS. Außerdem werden die Fortentwicklung des IPv4 durch IPv6 sowie die Algorithmen von TCP, Multipath-TCP und SCTP vorgestellt. Ferner werden die schnelle Paketvermittlung in IP-Netzen mit Dienstgüteunterstützung und der Einsatz neuer Betriebsmittel- und Verkehrsmanagementverfahren, z.B. Speicherverwaltungsalgorithmen wie RED, RIO und Schedulingalgorithmen wie WFQ, angesprochen. Ferner werden neueste Architekturansätze für Next Generation Networks (NGN), wie z.B. Software-Definierte Netze und Informationszentrierte Netze, diskutiert.

Darüber hinaus werden typische Anwendungen des Multimedia-Internet der 3. und 4. Generation wie Webanwendungen auf Basis von HTML5 und HTTP 2.0, WebRTC, Voice-over-IP und Medien-Streaming skizziert.  
 Die Vertiefung durch die Lehrveranstaltung Grundbausteine der Internet-Kommunikation mit ihren praktischen Übungen sowie die Fortführung durch Hauptseminare und Masterarbeiten ist möglich und eine wichtige Zielsetzung dieser Lehrveranstaltung.  
 Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.

**Literatur:**

- Kurose, J., Ross, K.W.: Computernetzwerke – ein Top-Down-Ansatz mit Schwerpunkt Internet, Pearson Studium, München, 2013.
- Kurose, J.F., Ross, K.W.: Computer Networking, A Top-Down Approach Featuring the Internet, Pearson Addison-Wesley, 7th ed., 2017.
- Leon-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2nd ed. 2004.
- Comer, D.: Computernetzwerke und Internets, Pearson Studium, München, 2001.

Weitere Literatur wird in der Vorlesung benannt.

**Prüfung**

mündliche Prüfung / Prüfungsdauer: 30 Minuten

**Beschreibung:**

Bestehen einer mündlichen Prüfung über die Inhalte der Vorlesung und Übung.  
 Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstaltung.



<b>Modul KTR-Mobi-M Mobilkommunikation</b> <i>Mobile Communication</i>	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit WS17/18) Modulverantwortliche/r: Prof. Dr. Udo Krieger	
<p><b>Inhalte:</b></p> <p>Die Lehrveranstaltung stellt die grundlegenden Techniken der Mobilkommunikation vor. Es werden relevante Standards, Systemarchitekturen und Realisierungen sowie aktuelle Forschungs- und Entwicklungstrends diskutiert.</p> <p>Aufgrund des großen Umfangs des Themengebiets kann die Lehrveranstaltung nur exemplarisch die wichtigsten Aspekte jener drahtlosen mobilen Kommunikationssysteme darlegen, welche derzeit den stärksten Wachstumsmarkt darstellen und in heute alle Gesellschaftsbereiche durchdringen. In der Veranstaltung stehen die Systemaspekte der Netz- und Dienstarchitekturen mobiler Kommunikationssysteme im Vordergrund.</p> <p>Im Detail werden folgende Themen behandelt:</p> <ul style="list-style-type: none"> <li>• technische Grundlagen der drahtlosen Übertragung</li> <li>• Medienzugriffsverfahren</li> <li>• Betriebsmittelzuteilung in Mobilkommunikationsnetzen (Betriebsmittelzuteilungsstrategien auf der Funkebene, Verbindungsübergabe, Fehlersicherungsprotokolle, Schedulingverfahren u.a.)</li> <li>• Mobilitätsunterstützung auf der Vermittlungsschicht durch Mobile IP</li> <li>• Transportprotokolle und ihre Erweiterungen</li> <li>• drahtlose LANs und ihre Erweiterungen (IEEE802.11 Standards, WiMAX u.a.)</li> <li>• drahtlose Weitverkehrsnetze mit TDMA-Technologie (GSM Grundlagen und Protokolle, GPRS)</li> <li>• Datenkommunikation in drahtlosen Weitverkehrsnetzen (UMTS, HSPA, LTE, LTE-A u.a.)</li> <li>• Dienstarchitekturen für Mobilfunknetze</li> </ul>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Die Studierenden sollen zu eigenständigem Arbeiten befähigt werden. Es werden Grundkenntnisse der Mobilkommunikation und die systematische Analyse der verwendeten Algorithmen mit Hilfe eines interaktiven Übungskonzeptes vermittelt. Die Studierenden lernen, gegebene Implementierungen der vorgestellten Kommunikationsverfahren zu analysieren und durch Messungen mit Wireshark und anderen Werkzeugen ihr Leistungsverhalten zu überprüfen. Die Bearbeitung von Aufgaben im Team ist Bestandteil der Ausbildung.</p>	
<p><b>Sonstige Informationen:</b></p> <p>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>The module can be selected by exchange students and master students speaking only English.</p>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b></p> <p>keine</p>	

<b>Empfohlene Vorkenntnisse:</b> Solide Kenntnisse der Lehrveranstaltung Datenkommunikation (im Umfang von KTR-Datkomm-B oder einer Lehrveranstaltung mit vergleichbaren Inhalten) sowie gute Programmierkenntnisse in JAVA (und/oder C++) sollten dringend vorhanden sein.  Modul Fortgeschrittene Java Programmierung (DSG-AJP-B) - empfohlen Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen 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>Mobilkommunikation</b> <b>Lehrformen:</b> Vorlesung und Übung <b>Dozenten:</b> Prof. Dr. Udo Krieger <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>4,00 SWS</b>
<b>Inhalte:</b> Die Lehrveranstaltung stellt die grundlegenden Techniken der Mobilkommunikation vor. Es werden relevante Standards, Systemarchitekturen und Realisierungen sowie aktuelle Forschungs- und Entwicklungstrends diskutiert. Aufgrund des großen Umfanges des Themengebietes kann die Lehrveranstaltung nur exemplarisch die wichtigsten Aspekte jener drahtlosen mobilen Kommunikationssysteme darlegen, welche derzeit den stärksten Wachstumsmarkt darstellen und in heute alle Gesellschaftsbereiche durchdringen. In der Veranstaltung stehen die Systemaspekte der Netz- und Dienstarchitekturen mobiler Kommunikationssysteme im Vordergrund. Im Detail werden folgende Themen behandelt: <ul style="list-style-type: none"> <li>• technische Grundlagen der drahtlosen Übertragung</li> <li>• Medienzugriffsverfahren</li> <li>• Betriebsmittelzuteilung in Mobilkommunikationsnetzen (Betriebsmittelzuteilungsstrategien auf der Funkebene, Verbindungsübergabe, Fehlersicherungsprotokolle, Schedulingverfahren u.a.)</li> <li>• Mobilitätsunterstützung auf der Vermittlungsschicht durch Mobile IP</li> <li>• Transportprotokolle und ihre Erweiterungen</li> <li>• drahtlose LANs und ihre Erweiterungen (IEEE802.11 Standards, WiMAX u.a.)</li> <li>• drahtlose Weitverkehrsnetze mit TDMA-Technologie (GSM Grundlagen und Protokolle, GPRS)</li> <li>• Datenkommunikation in drahtlosen Weitverkehrsnetzen (UMTS, LTE u.a.)</li> </ul>	

<p>Die Inhalte der Vorlesung werden in den Übungen durch das eigenständige Bearbeiten von Aufgaben und das Vorstellen und Diskutieren der Lösungen im Gruppenrahmen sowie durch Laboraufgaben vertieft und weitergeführt. Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.</p>	
<p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Schiller, J.: Mobilkommunikation. Pearson-Education/Addison-Wesley, München, 2003.</li> <li>• Walke, B.: Mobilfunknetze und ihre Protokolle Bd. 1 &amp; 2. B.G. Teubner, 3. Aufl. 2001.</li> <li>• Pahlavan, K., Krishnamurthy, P.: Principles of Wireless Networks, A Unified Approach. Prentice Hall, 2002.</li> <li>• Pahlavan, K., Krishnamurthy, P.: Networking Fundamentals: Wide, Local and Personal Area Communications, Wiley, 2009.</li> <li>• Walke, B. u.a.: UMTS - Ein Kurs, Schlembach, 2002.</li> <li>• Holma, H., Toskala, A.: LTE for UMTS, Evolution to LTE-Advanced, 2. ed, Wiley, 2011.</li> </ul> <p>Weitere Literatur wird in der Vorlesung benannt.</p>	
<p><b>Prüfung</b> mündliche Prüfung / Prüfungsdauer: 30 Minuten</p> <p><b>Beschreibung:</b> Die Prüfung der Inhalte der Vorlesung und Übung erfolgt in Form einer mündlichen Prüfung.</p> <p>Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstaltung.</p>	

<p><b>Modul KTR-SSSProj-M KTR Masterprojekt Software Systems Science</b>  <i>KTR Master Project Software Systems Science</i></p>	<p>9 ECTS / 270 h            70 h Präsenzzeit            200 h Selbststudium</p>
<p>(seit WS17/18)            Modulverantwortliche/r: Prof. Dr. Udo Krieger</p>	
<p><b>Inhalte:</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 industrienahe 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.</p>	
<p><b>Lernziele/Kompetenzen:</b>            Die Lehrveranstaltung vermittelt Einblicke in die Entwicklung neuer Dienstarchitekturen und Netztechnologien aus dem Bereich des Internets der nächsten Generation.            Ziel ist der Erwerb praktischer Fertigkeiten auf dem Gebiet der IP-gestützten, qualitätsgesicherten Multimediakommunikation und die Fähigkeit, Lösungsvorschläge moderner Dienstarchitekturen im Internet der Zukunft sicher beurteilen zu können.            Studierende sollen ein vertieftes Verständnis der bei der Durchführung von Software-Projekten im Bereich Kommunikationsnetze und -dienste auftretenden konzeptionellen und praktischen Probleme wie auch von erfolgsversprechenden Lösungsansätzen dieser Probleme erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich der Professur für Informatik in Kleingruppen oder einzeln geschieht, gewinnen die Studierenden wichtige Erfahrungen in der Durchführung größerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in wissenschaftlich ausgerichteten Arbeitsberichten und in der professionellen Präsentation dieser Ergebnisse.</p>	
<p><b>Sonstige Informationen:</b>            Dieses Modul erstreckt sich über ein Semester. Der Arbeitsaufwand beträgt insgesamt 270 Std., welche sich grob wie folgt gliedern:</p> <ul style="list-style-type: none"> <li>• 30 Std. Recherche, Planung und Teilnahme am Planungsworkshop</li> <li>• 40 Std. Teilnahme an Projekttreffen, einschließlich Tutorien</li> <li>• 150 Std. Durchführung des Projekts (Projektarbeit)</li> <li>• 50 Std. Erstellung des Abschlussberichts, Erstellung und Präsentation der Projektergebnisse (Hausarbeit und Kolloquium)</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>            keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b>            Es werden empfohlen:</p> <ul style="list-style-type: none"> <li>• gute Kenntnisse in Mathematik für Informatiker 2 (KTR-MfI-2)</li> <li>• mindestens gute JAVA (oder C/C++) Kenntnisse</li> </ul>	<p><b>Besondere Bestehensvoraussetzungen:</b>            keine</p>

<ul style="list-style-type: none"> <li>• Kenntnisse der Datenkommunikation im Umfang von KTR-Datkomm-B oder vergleichbare Kenntnisse</li> <li>• grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z.B. im Umfang des Moduls "Software EngineeringLab" (SWT-SWL-B)</li> </ul> <p>Modul Programmierung komplexer interagierender Systeme (DSG-PKS-B) - empfohlen  Modul Datenkommunikation (KTR-Datkomm-B) - empfohlen  Modul Mathematik für Informatik 2 (Lineare Algebra) (KTR-Mfl-2) - empfohlen  Modul Software Engineering Lab (SWT-SWL-B) - empfohlen</p>		
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b> 2.	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>Masterprojekt Software Systems Science</b>  <b>Lehrformen:</b> Projekt  <b>Dozenten:</b> Prof. Dr. Udo Krieger  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> WS, SS</p> <hr/> <p><b>Lernziele:</b>  siehe oben</p> <hr/> <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 Fachgebiets 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 Lehrveranstaltung erstreckt sich über ein Semester. 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:</p> <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 Arbeitsergebnissen in einem Abschlussbericht mit Abschlusspräsentation der Arbeitsergebnisse in einem Kolloquium.</li> </ul> <p>Es werden Entwicklungsaufgaben zu aktuellen Forschungsfragen im "Future Generation Internet" bearbeitet. Details werden auf der Webseite der</p>	<b>6,00 SWS</b>

Lehrveranstaltung angekündigt. Eine aktuelle Liste der bearbeiteten Themen der Lehrveranstaltung wird in der 1. Besprechung bereitgestellt.  
Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstaltung.

**Literatur:**

Literatur wird in der 1. Besprechung bekanntgegeben.

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten

Bearbeitungsfrist: 4 Monate

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

**Beschreibung:**

Die Lehrveranstaltung erstreckt sich über ein Semester. Es werden die Leistungen der als Gruppen- oder Einzelarbeit ausgeführten individuellen schriftlichen Ausarbeitung der Projektaufgaben und ihrer Präsentation sowie die Ergebnisse einer abschließenden, individuellen Kolloquiumsprüfung bewertet.

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

<b>Modul KogSys-ML-M Lernende Systeme (Machine Learning)</b> <i>Machine Learning</i>		6 ECTS / 180 h
(seit WS19/20) Modulverantwortliche/r: Prof. Dr. Ute Schmid		
<b>Inhalte:</b> Die Veranstaltung vermittelt vertieftes Wissen und Kompetenzen im Bereich Maschinelles Lernen mit dem Fokus auf symbolischen, neuronalen und statistischen Algorithmen.		
<b>Lernziele/Kompetenzen:</b> Die Studierenden können: <ul style="list-style-type: none"> <li>• Forschungsziele des Maschinellen Lernens nennen</li> <li>• Forschungsfragen des Maschinellen Lernens erläutern</li> <li>• grundlegende Konzepte des Klassifikationslernens nennen und erläutern</li> <li>• symbolische, neuronale und statistische Algorithmen des Klassifikationslernens nennen und auf gegebene Daten anwenden</li> <li>• die Eignung gegebener Daten für Algorithmen des Klassifikationslernens beurteilen</li> <li>• spezielle Verfahren des Maschinellen Lernens nennen, erläutern und anwenden</li> <li>• Grundlegende Konzepte des menschlichen Lernens nennen und erläutern</li> <li>• Verbindungen zwischen menschlichem und maschinellern Lernen erörtern</li> </ul>		
<b>Sonstige Informationen:</b> Veranstaltung Deutsch (im Bedarfsfall Englisch). 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> Modul Mathematik für Informatiker 1 (Aussagen- und Prädikatenlogik) (Gdl-Mfl-1) . Modul Algorithmen und Datenstrukturen (MI-AuD-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. Lernende Systeme (Machine Learning)</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Ute Schmid <b>Sprache:</b> Deutsch/Englisch <b>Angebotshäufigkeit:</b> WS, jährlich <b>Lernziele:</b>		<b>2,00 SWS</b>

<p>s.o.</p> <hr/> <p><b>Inhalte:</b>                  In der Vorlesung werden wesentliche symbolische, statistische und neuronale Ansätze des Maschinellen Lernens mit Bezügen zum menschlichen Lernen vertiefend eingeführt. Typische behandelte Themengebiete sind:                  Entscheidungsbaumalgorithmen, Multilayer Perzeptrons, Instance-based Learning, Induktive Logische Programmierung, Genetische Algorithmen, Bayes'sches Lernen, Kernel Methods, Support Vector Machines, Induktive Programmsynthese und Reinforcement Learning.                  Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p> <hr/> <p><b>Literatur:</b>                  Mitchell, Machine Learning, McGraw-Hill, 1997;                  Goodfellow et al., Deep Learning, MIT Press, 2016;                  Bishop, Pattern Recognition and Machine Learning, Springer, 2006;</p>	
<p><b>2. Lernende Systeme (Machine Learning)</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Mitarbeiter Angewandte Informatik, insb. Kognitive Systeme</p> <p><b>Sprache:</b> Deutsch/Englisch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b>                  s.o.</p> <hr/> <p><b>Inhalte:</b>                  Vertiefung von in der Vorlesung eingeführten Methoden und Techniken, zum Teil mit Programmieraufgaben und Anwendungen in RapidMiner.                  Die Lehrsprache wird in der ersten Lehrveranstaltung bekanntgegeben.</p> <hr/> <p><b>Literatur:</b>                  siehe Vorlesung</p>	<p><b>2,00 SWS</b></p>
<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 freiwillige Studienleistungen (Übungsblätter) ausgegeben. Durch die freiwillige Bearbeitung der Studienleistungen können Punkte zur Notenverbesserung gesammelt werden, die auf die Klausur anrechenbar sind, sofern die Klausur auch ohne Punkte aus den optionalen Studienleistungen bestanden ist. Zu Beginn der Lehrveranstaltung wird bekannt gegeben:</p> <ul style="list-style-type: none"> <li>• Art und Anzahl der Studienleistungen</li> <li>• Umfang (Anzahl an erreichbaren Punkte) der Studienleistungen</li> <li>• Bearbeitungsdauer der Studienleistungen</li> </ul> <p>Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.</p>	



---

Erlaubte Hilfsmittel: Handschriftliche und gedruckte Materialien, Taschenrechner ohne vollständige alphanumerische Tastatur und Grafikdisplay.

Die Prüfungssprache wird in der ersten Lehrveranstaltung bekanntgegeben.

<b>Modul MOBI-ADM-M Advanced Data Management</b> <i>Advanced Data Management</i>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(seit SS19 bis WS19/20) Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
<b>Inhalte:</b> This course covers recent trends in data management (e.g., so-called NOSQL databases) that go beyond the traditional relational data model. Such systems are designed to fulfil novel requirements (e.g., the ability to scale out or schema flexibility). They often relax requirements of traditional relational databases (e.g., consistency). In the course, we will discuss different approaches to model, manage, store, and retrieve data.		
<b>Lernziele/Kompetenzen:</b> Students understand the design goals, benefits and drawbacks of NOSQL database systems. They are able to decide which database system is appropriate for a given application depending on suitable criteria. They can design database structures for different NOSQL data models. They understand the implementation of internal components and storage structures of selected database systems.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Comprehension of the relational data model, relational algebra, and SQL language, obtained e.g. from the Module MOBI-DBS-B: Datenbanksysteme; Basic programming skills 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. Advanced Data Management</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> This course covers recent trends in data management (e.g., so-called NOSQL databases) that go beyond the traditional relational data model. Such systems are designed to fulfil novel requirements (e.g., the ability to scale out or schema flexibility). They often relax requirements of traditional relational databases (e.g., consistency). In the course, we will discuss different approaches to model, manage, store, and retrieve data.	
<b>Literatur:</b>	

L. Wiese, Advanced Data Management, For SQL, NoSQL, Cloud and Distributed Databases. Berlin, Boston: De Gruyter, 2015	
<b>2. Advanced Data Management</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> Practical exercises for lecture topics	
<b>Prüfung</b> mündliche Prüfung / Prüfungsdauer: 15 Minuten <b>Beschreibung:</b> Mündliche Prüfung oder Schriftliche Prüfung (wird zu Beginn des Semesters durch Ankündigung in der Veranstaltung festgelegt). Die Prüfungssprache ist Englisch.	

<b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Beschreibung:</b> Mündliche Prüfung oder Schriftliche Prüfung (wird zu Beginn des Semesters durch Ankündigung in der Veranstaltung festgelegt). Die Prüfungssprache ist Englisch.	
--	--

<b>Modul MOBI-DSC-M Data Streams and Complex Event Processing</b>		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
<i>Data Streams and Complex Event Processing</i>		
(seit WS19/20)		
Modulverantwortliche/r: Prof. Dr. Daniela Nicklas		
<b>Inhalte:</b> The management of data streams and foundations of event processing: Applications, systems, query languages, continuous query processing, and security in distributed data stream management systems. The modul covers the following topics: Architectures of data stream management systems; Query languages; Data stream processing; Complex event processing; Security in data stream management systems; Application of data stream management systems		
<b>Lernziele/Kompetenzen:</b> Understand the challenges of data stream management and complex event processing Recognize and link basic building blocks of data stream management tasks in different frameworks and systems Develop and program queries on data streams and event streams in different query languages to process data streams and detect event patterns Understand basic implementation techniques for data stream operators Understand the main security challenges and solutions in data stream management systems		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Foundations of relational databases, relational algebra and SQL; e.g. from Modul MOBI-DBS-B: Database Systems		<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>Data Streams and Complex Event Processing</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Lernziele:</b> Understand the challenges of data stream management and complex event processing Recognize and link basic building blocks of data stream management tasks in different frameworks and systems Develop and program queries on data streams and event streams in different query languages to process data streams and detect event patterns Understand basic implementation techniques for data stream operators	

Understand the main security challenges and solutions in data stream management systems	
<p><b>Inhalte:</b> The management of data streams and foundations of event processing: Applications, systems, query languages, continuous query processing, and security in distributed data stream management systems. The modul covers the following topics: Architectures of data stream management systems; Query languages; Data stream processing; Complex event processing; Security in data stream management systems; Application of data stream management systems</p>	
<p><b>Prüfung</b> mündliche Prüfung / Prüfungsdauer: 15 Minuten <b>Beschreibung:</b> Mündliche Prüfung oder Schriftliche Prüfung (wird zu Beginn des Semesters durch Ankündigung in der Veranstaltung festgelegt). Die Prüfungssprache ist Englisch.</p>	
<b>Lehrveranstaltungen</b>	
<p><b>Data Streams and Complex Event Processing</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Englisch <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b> siehe Vorlesung</p>	<b>2,00 SWS</b>
<p><b>Prüfung</b> schriftliche Prüfung (Klausur) / Prüfungsdauer: 60 Minuten <b>Beschreibung:</b> Mündliche Prüfung oder Schriftliche Prüfung (wird zu Beginn des Semesters durch Ankündigung in der Veranstaltung festgelegt). Die Prüfungssprache ist Englisch.</p>	

<p><b>Modul MOBI-PRS-M Master Project Mobile Software Systems (SoSySc)</b>  <i>Master Project Mobile Software Systems (SoSySc)</i></p>	<p>9 ECTS / 270 h</p>
<p>(seit WS19/20)                  Modulverantwortliche/r: Prof. Dr. Daniela Nicklas</p>	
<p><b>Inhalte:</b>                  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.                   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-M Data streams and event processing.                   The tasks in the project will be tailored to Master level.</p>	
<p><b>Lernziele/Kompetenzen:</b>                  Students will deepen their knowledge regarding the conceptual problems that arise when carrying out theoretical and/or practical research and software projects, and regarding approaches to possible solutions. Since this will be done by means of the intensive conduct of a research topic in Mobile Software Systems, students will gain important experience in carrying out research-oriented projects, from project planning, to the abstract and concrete design, to the realization, to the documentation of results in a scientific project report.</p>	
<p><b>Sonstige Informationen:</b>                  Bemerkung                   The main language of instruction is English. The module may be delivered in German if all participating students are fluent in German.                   The total workload of 270 hrs. is split approximately as follows:</p> <ul style="list-style-type: none"> <li>• 15 hrs. participating in introductions to and tutorials on methods, software tools, and giving presentations on the project status</li> <li>• 60 hrs. researching and familiarization with the project topic</li> <li>• 150 hrs. conducting the project work</li> <li>• 45 hrs. compiling a project report (Assignment/Hausarbeit) and preparation of the Colloquium (Kolloquium).</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>                  keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b>                  Programming skills (Java preferred), e.g. from the module "DSG-AJP-B"; Software project management, e.g. from the module "SWT-SWL-B Software Engineering Lab"; Scientific research and writing, e.g. from the module "IAIWAI-B Wissenschaftliches Arbeiten" or</p>	<p><b>Besondere Bestehensvoraussetzungen:</b>                  keine</p>

SSS-SRW-M Scientific Research on Writing for Master´s Students; Relational databases and SQL, e.g. from the module "SEDA-DBS-B Datenbanksysteme".		
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Master Project Mobile Software Systems (SoSySc)</b> <b>Lehrformen:</b> Übung <b>Dozenten:</b> Prof. Dr. Daniela Nicklas <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>6,00 SWS</b>
<b>Inhalte:</b> Conduct of the project, accompanied by regular meetings between students and lecturer.	
<b>Prüfung</b> Hausarbeit mit Kolloquium <b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regelmäßige Teilnahme an der Lehrveranstaltung <b>Beschreibung:</b> Als Prüfungsleistung ist eine Hausarbeit sowie ein Kolloquium zu erbringen. Die Bearbeitungsfrist der Hausarbeit und die Prüfungsdauer des Kolloquiums werden zu Beginn einer jeden Lehrveranstaltung von der Projektleiterin bzw. dem Projektleiter 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). The term of the project report and of the colloquium will be announced at the beginning of each course by the project leader.	

<p><b>Modul PSI-AdvaSP-M Advanced Security and Privacy</b>  <i>Advanced Security and Privacy</i></p>	<p>6 ECTS / 180 h          45 h Präsenzzeit          135 h Selbststudium</p>
<p>(seit SS18)          Modulverantwortliche/r: Prof. Dr. Dominik Herrmann</p>	
<p><b>Inhalte:</b>          Information security and privacy are relevant in almost all information systems today. Many real-world use cases have complex security and privacy requirements involving multiple parties. Often there are multiple stakeholders with different, sometimes even contradictory interests. For instance, some use cases call for a solution that allows a service provider to process sensitive data without learning its content. In other cases it is not the content but some meta information such as location and usage intensity that has to be protected. And then there are scenarios where seemingly harmless pieces of data can be used to disclose or infer very personal pieces of information about an individual.</p> <p>This module covers advanced techniques for information security and privacy that can be used to satisfy the complex requirements of practical systems. It builds upon the basic concepts in information security that are introduced in the module "Introduction to Security and Privacy" (PSI-IntroSP-B).</p>	
<p><b>Lernziele/Kompetenzen:</b>          This module is designed to bring students towards the research boundaries in the field of security and privacy technologies by covering a selection of contemporary topics in depth. The focus of the module is on technical safeguards that can be used by system designers and users to enforce properties such as confidentiality and integrity. Moreover, sophisticated attacks on security and privacy are explained.</p> <p>Successful students will be able to explain attack strategies and defenses discussed in recent research papers. They will also be able to analyze whether a particular attack or defense is relevant in a specific scenario. Finally, they will be able to implement selected attacks and defenses with a programming language of their choice.</p>	
<p><b>Sonstige Informationen:</b>          This module is taught in English. It consists of a lecture and tutorials. During the course of the tutorials there will be theoretical and practical assignments (task sheets). Assignments and exam questions can be answered in English or German.</p> <p>Lecture and tutorials are partially taught in form of a paper reading class. Participants are expected to read the provided literature in advance and participate in the discussions.</p> <p>Workload breakdown:</p> <ul style="list-style-type: none"> <li>• Lecture: 22.5 hours (2 hours per week)</li> <li>• Tutorials: 22.5 hours (2 hours per week)</li> <li>• Preparation and studying during the semester: 30 hours</li> <li>• Assignments: 67.5 hours</li> <li>• Preparation for the exam (including the exam itself): 37.5 hours</li> </ul>	
<p><b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>          keine</p>	
<p><b>Empfohlene Vorkenntnisse:</b>          Participants should be familiar with basic concepts in information security and privacy, which can be acquired, for instance, by taking the module "Introduction to Security and Privacy" (PSI-IntroSP-B).</p>	<p><b>Besondere Bestehensvoraussetzungen:</b>          keine</p>



This includes basic knowledge about the commonly used security terminology, common types of malware and attacks, buffer overflows and related attacks, cryptography, network security, web security, and concepts of privacy. Moreover, participants should have practical experience with at least one scripting or programming language such as Python or Java.		
<b>Angebotshäufigkeit:</b> SS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>1. Advanced Security and Privacy</b></p> <p><b>Lehrformen:</b> Vorlesung</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p> <hr/> <p><b>Lernziele:</b> cf. module description</p> <hr/> <p><b>Inhalte:</b> Selected topics:</p> <ul style="list-style-type: none"> <li>• Cryptographic methods and protocols, e.g., homomorphic encryption, attribute-based credentials, secure multi-party computation, zero-knowledge proofs, format-preserving and identity-based encryption, group signatures, and proxy re-encryption.</li> <li>• Attacks on privacy in datasets and communications (inference techniques, online tracking)</li> <li>• Privacy engineering and privacy enhancing technologies (e.g., Tor)</li> <li>• Usable security and privacy</li> <li>• Other current topics in privacy and security</li> </ul> <p>Some parts of the lecture are aligned with current events and recently published research. The selected topics are therefore subject to change.</p> <hr/> <p><b>Literatur:</b> Selected books:</p> <ul style="list-style-type: none"> <li>• R. Anderson: Security Engineering</li> <li>• A. Shostack: Threat Modelling</li> <li>• J.-P. Aumasson: Serious Cryptography</li> <li>• W. Stallings: Computer Security: Principles and Practice</li> <li>• B. Schneier et al.: Cryptography Engineering</li> <li>• J. Erickson: Hacking: The Art of Exploitation</li> <li>• J. Katz &amp; Y. Lindell: Introduction to Modern Cryptography</li> <li>• L. Cranor &amp; S. Garfinkel: Security and Usability</li> </ul>	<b>2,00 SWS</b>
<p><b>2. Tutorials for Advanced Security and Privacy</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> SS, jährlich</p>	<b>2,00 SWS</b>

---

**Prüfung**

schriftliche Prüfung (Klausur) / Prüfungsdauer: 90 Minuten

**Beschreibung:**

The content that is relevant for the exam consists of the content presented in the lecture and tutorials (including the assignments) as well as the content of the discussed papers. The maximum number of points that can be achieved in the exam is 100.

Participants that solve all assignments correctly can collect up to 20 bonus points. Details regarding the number of assignments, the number of points per assignment, and the type of assignments will be announced in the first lecture. If the points achieved in the exam are sufficient to pass the exam on its own (generally, this is the case when at least 50 points have been obtained), the bonus points will be added to the points achieved in the exam. The grade 1.0 can be achieved without the bonus points.

<b>Modul PSI-ProjectCAD-M Project Complex Attacks and Defenses</b> <i>Project Complex Attacks and Defenses</i>	9 ECTS / 270 h
(seit SS18) Modulverantwortliche/r: Prof. Dr. Dominik Herrmann	
<p><b>Inhalte:</b></p> <p>Breaking into information systems is exciting, but impractical due to ethical and legal concerns. However, offensive competences and adversarial thinking are essential to build secure systems. In this project students will get the opportunity to acquire practical security skills in a dedicated training environment.</p> <p>The goal of this project is to build and extend the "Insekta" platform. This web-based tool provides a frontend for virtual machines that can be used to study selected topics in security and privacy on one's own and at one's own pace.</p> <p>This project is offered together with PSI-ProjectPAD, which focuses on conceptually simpler attacks and defenses.</p> <p>The participants of the project familiarize themselves with security weaknesses in information systems and apply this knowledge to develop vulnerable services which others can use for training. To this end, participants form groups, read about attacks and defenses in textbooks and research papers, and discuss various options to implement them. Instructors will provide extensive and on-demand support to enable the participants to implement a vulnerable service that can be exploited to learn about a particular vulnerability.</p> <p>Besides implementing vulnerable services, the participants prepare training materials, which consist of questions and tasks to test one's knowledge as well as step-by-step instructions. These training materials may also contain interactive elements for an improved learning experience.</p> <p>The project also takes into account attacks on privacy, e.g., re-identifying individuals in anonymized datasets and communication networks, tracking users on the Internet, inferring sensitive attributes from seemingly harmless data traces, as well as mitigations, e.g., depersonalization strategies and differential privacy mechanisms. Here, practical activities consist in the preparation of datasets and scripts for analysis.</p>	
<p><b>Lernziele/Kompetenzen:</b></p> <p>Successful students will be able to describe attacks and defenses from textbooks and research papers in easily understandable form. They will also be able to carry out selected attacks in practice and implement defenses with a programming language of their choice.</p>	
<p><b>Sonstige Informationen:</b></p> <p>This project is taught in English, unless all participants are fluent in German. The workload of this project is equivalent to 270 hours.</p> <p>Workload breakdown:</p> <ul style="list-style-type: none"> <li>• 20 hrs: Getting familiar with the platform</li> <li>• 50 hrs: Reading papers and researching security vulnerabilities</li> <li>• 20 hrs: Preparing the talk (including time for attendance of other talks)</li> <li>• 90 hrs: Implementing the vulnerable service and defenses</li> <li>• 90 hrs: Writing training material and documentation</li> </ul> <p>Note that there is another project (PSI-ProjectPAD) with a workload equivalent to 180 hours.</p>	

<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
<p>This project is primarily intended for students in master programs. Students in bachelor programs can participate, if they are qualified.</p> <p>Participants should be familiar with basic concepts in information security and privacy, which can be acquired, for instance, by taking the module "Introduction to Security and Privacy" (PSI-IntroSP-B). This includes basic knowledge about the commonly used security terminology, common types of malware and attacks, buffer overflows and related attacks, cryptography, network security, web security, and concepts of privacy.</p> <p>Moreover, participants should have practical experience with at least one scripting or programming language such as Python or Java. Experience with Linux environments, web technologies, and network protocols is recommended.</p>		keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>Project Complex Attacks and Defenses</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, SS</p> <hr/> <p><b>Lernziele:</b></p> <p>cf. module description</p> <hr/> <p><b>Inhalte:</b></p> <p>Potential topics include:</p> <ul style="list-style-type: none"> <li>• web security (injection flaws and other issues mentioned in the OWASP Top 10)</li> <li>• network security (such as DNS cache poisoning and rebinding attacks)</li> <li>• security issues in C programs (buffer overflows, etc.)</li> <li>• cryptography (low-level attacks on ciphers, high-level attacks on protocols, e.g., TLS)</li> <li>• business logic failures</li> <li>• misconfigurations</li> <li>• attacks on availability (denial of service)</li> <li>• attacks on privacy (such as inference, tracking, re-identification, fingerprinting)</li> <li>• privacy defenses (such as k-anonymity, related concepts, differential privacy)</li> </ul> <hr/> <p><b>Literatur:</b></p> <p>Literature will be announced at the beginning of the project.</p>	<p><b>6,00 SWS</b></p>

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten

Bearbeitungsfrist: 3 Monate

**Zulassungsvoraussetzung zur Modulprüfung:**

Regular attendance at project meetings.

**Beschreibung:**

The module examination consists of two parts: Firstly, the participants submit a written report (in English) that includes the source code of the vulnerable service and the training material. Secondly, the participants give a talk in which they defend their work (in English; in German if all participants are fluent in German) by presenting theoretical and practical aspects of their vulnerable service as well as relevant mitigations. The maximum number of points that can be achieved in the module examination is 100.

Optionally, participants can submit intermediary results (in English) to collect up to 20 bonus points. If the module examination is passed on its own (generally, this is the case when at least 50 points are obtained), the bonus points will be added to the points achieved in the module examination. The grade 1.0 can be achieved without the bonus points. Details regarding the number of optional submissions during the semester, their type, the points per submission, and the respective deadlines will be announced in the first session of the project.

<b>Modul PSI-ProjectSP-M Project Security and Privacy</b>		9 ECTS / 270 h
<i>Project Security and Privacy</i>		
(seit SS18) Modulverantwortliche/r: Prof. Dr. Dominik Herrmann		
<b>Inhalte:</b> In this project participants work independently on problems related to current research activities of the Privacy and Security in Information Systems Group. Instructors will provide guidance and supervision.		
<b>Lernziele/Kompetenzen:</b> Successful students will be able to independently work on research problems in security and privacy. They will also be able to implement tools and/or analyze data in order to answer a research question. Finally, they will be able to present their work in a talk and document their approach and results in a written report.		
<b>Sonstige Informationen:</b> This project is taught in English unless all participants are fluent in German. The workload of this project is equivalent to 270 hours.  Workload breakdown: <ul style="list-style-type: none"> <li>• 60 hrs: Getting familiar with the problem and preliminaries: reading related work, and understanding potentially existing source code</li> <li>• 20 hrs: Preparing the talk (including time for attendance of other talks)</li> <li>• 110 hrs: Implementing tools and/or analyzing data</li> <li>• 80 hrs: Writing final report with approach and methods</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Participants should have advanced knowledge and practical skills in information security and privacy, which can be acquired, for instance, in the module PSI-IntroSP-B and a security-related seminar or project. Depending on the actual topic participants may be expected to be familiar with commonly used security terminology, common types of malware and attacks, buffer overflows and related attacks, cryptography, network security, web security, and concepts of privacy.  Moreover, participants should have practical experience with at least one scripting or programming language such as Python or Java. Alternatively, participants should have strong skills in empirical data collection and data analytics (statistics and/or machine learning).  Experience with Linux environments, web technologies, and network protocols is recommended.		<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>Project Security and Privacy</b> <b>Lehrformen:</b> Übung <b>Sprache:</b> Englisch/Deutsch		<b>6,00 SWS</b>

<p><b>Angebotshäufigkeit:</b> WS, SS</p> <hr/> <p><b>Lernziele:</b> cf. module description</p> <hr/> <p><b>Inhalte:</b> Potential topics include</p> <ul style="list-style-type: none"> <li>• empirical studies, either manually (surveying security properties of systems) or automatically (e.g., web crawls),</li> <li>• creating scanning tools and platforms where results can be published in a meaningful way (e.g., PrivacyScore.org),</li> <li>• analyzing data sets for aspects of security and privacy, and</li> <li>• implementing cryptographic or anonymization techniques in a secure fashion, e.g., for encrypted storage in cloud services.</li> </ul> <hr/> <p><b>Literatur:</b> Literature will be announced at the beginning of the project.</p>	
--	--

<p><b>Prüfung</b> Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten Bearbeitungsfrist: 3 Monate</p> <p><b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regular attendance at project meetings.</p> <p><b>Beschreibung:</b> The module examination consists of two parts: Firstly, the participants submit a written report (in English) that includes the source code, datasets, and analysis scripts. Secondly, the participants give a talk in which they defend their work (in English; in German if all participants are fluent in German) by presenting related work, their approach, and results. The maximum number of points that can be achieved in the module examination is 100.</p> <p>Optionally, participants can submit intermediary results (in English) to collect up to 20 bonus points. If the module examination is passed on its own (generally, this is the case when at least 50 points are obtained), the bonus points will be added to the points achieved in the module examination. The grade 1.0 can be achieved without the bonus points. Details regarding the number of optional submissions during the semester, their type, the points per submission, and the respective deadlines will be announced in the first session of the project.</p>	
--	--

<b>Modul SME-STE-M Introduction to Knowledge Representation: Space, Time, Events</b> <i>Introduction to Knowledge Representation: Space, Time, Events</i>		6 ECTS / 180 h
(seit WS17/18) Modulverantwortliche/r: Prof. Dr. Diedrich Wolter		
<b>Inhalte:</b> This course gives an introduction to the area of knowledge representation, a sub-discipline of computer science in general and artificial intelligence in particular. Knowledge representation is involved with identifying means to represent practical problems and according background knowledge as data structures, and to develop reasoning algorithms to solve these problems. This course puts a spotlight on symbolic techniques to represent knowledge involving a spatio-temporal component as is typical for many practical real-world problems. Contents: <ul style="list-style-type: none"> <li>• fundamental concepts: knowledge, abstractions, relations, logics</li> <li>• syntax and semantics, formalization of knowledge</li> <li>• representation and reasoning</li> <li>• qualitative algebras and constraint calculi</li> <li>• constraint-based reasoning</li> <li>• spatial logics</li> <li>• complexity and tractable subclasses</li> </ul>		
<b>Lernziele/Kompetenzen:</b> <ul style="list-style-type: none"> <li>• gain overview of formalisms for representing spatio-temporal logics</li> <li>• gain skills to represent spatio-temporal knowledge symbolically</li> <li>• gain overview of reasoning problems and learn to identify approaches for solving them</li> <li>• learn to apply constraint-based reasoning methods</li> <li>• learn to identify computational complexity of reasoning problems</li> </ul>		
<b>Sonstige Informationen:</b> The main language of instruction in this course is English. The lectures and tutorials may be delivered in German if all participating students are fluent in German. The specification of the language will be made in the first week of lectures.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge in computer science is recommended, for example obtained in a computer science bachelor's curriculum.		<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. Introduction to Knowledge Representation: Space, Time, Events</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Diedrich Wolter		<b>2,00 SWS</b>



<p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Lernziele:</b> see description of module</p> <hr/> <p><b>Inhalte:</b> see description of module</p> <hr/> <p><b>Literatur:</b> will be announced in first lecture</p>	
<p><b>2. Introduction to Knowledge Representation: Space, Time, Events</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Dr. Diedrich Wolter</p> <p><b>Sprache:</b> Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b> practical excercises according to the lecture</p>	<b>2,00 SWS</b>
<p><b>Prüfung</b> mündliche Prüfung / Prüfungsdauer: 20 Minuten</p> <p><b>Beschreibung:</b> Exams may be taken in either English or German at the choice of the student.</p>	

<b>Modul SNA-OSN-M Projekt zu Online Social Networks</b>		6 ECTS / 180 h
<i>Project Online Social Networks</i>		
(seit WS13/14) Modulverantwortliche/r: Prof. Dr. Kai Fischbach Weitere Verantwortliche: Zylka, Matthäus, Dipl.-Wirt.-Inf.		
<b>Inhalte:</b> In der Veranstaltung werden aktuelle Forschungsthemen aus dem Bereich Online Social Networks (Digitale soziale Netzwerke) im Rahmen von Gruppenprojekten bearbeitet.		
<b>Lernziele/Kompetenzen:</b> Aufbauend auf den in den Vorlesungen und Übungen des Faches Soziale Netzwerke erworbenen Kenntnissen und Fertigkeiten wird in diesem Modul ein wissenschaftliches Projekt in einer Gruppe bearbeitet. Dabei werden die Fähigkeiten im Bereich Analyse sozialer Netzwerke ebenso weiterentwickelt wie die Kompetenzen in der Projektdurchführung und Gruppenarbeit. Die Projekte werden in nationaler und internationaler Zusammenarbeit mit Studierenden renommierter Universitäten umgesetzt. Bisherige Partneruniversitäten sind unter anderem das Massachusetts Institute of Technology (MIT), das Illinois Institute of Technology (IIT), die Aalto-Universität (Helsinki, Finnland), die Universität Tor Vergata (Rom, Italien) und die Universität zu Köln.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Teilnahme an mindestens einem der beiden folgenden Module: <ul style="list-style-type: none"><li>• Analyse sozialer Netzwerke (SNA-ASN-M)</li><li>• Netzwerktheorie (SNA-NET-M)</li></ul>		<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>Projekt zu Online Social Networks</b>  <b>Lehrformen:</b> Übung  <b>Dozenten:</b> Prof. Dr. Kai Fischbach  <b>Sprache:</b> Englisch/Deutsch  <b>Angebotshäufigkeit:</b> WS, jährlich</p> <hr/> <p><b>Inhalte:</b>  Die Methoden und Erkenntnisse der Analyse sozialer Netzwerke (SNA) haben innerhalb weniger Jahre einen erheblichen Bedeutungszuwachs in den Disziplinen Wirtschaftsinformatik, Informatik und Betriebswirtschaft erlangt. Ein wichtiger Grund für das stark angewachsene Interesse ist, dass die Erhebung und Untersuchung von Interaktionsstrukturen durch die zunehmende Verlagerung menschlicher Kommunikation auf elektronische Wege effektiver und effizienter geworden ist.  In der Veranstaltung werden wechselnde Projekte aus diesem Themenfeld bearbeitet.</p> <hr/> <p><b>Literatur:</b></p>	<b>4,00 SWS</b>

---

Literatur wird in der Veranstaltung bekannt gegeben.	
--	--

**Prüfung**

Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten

Bearbeitungsfrist: 4 Monate

**Zulassungsvoraussetzung zur Modulprüfung:**

Regelmäßige Teilnahme an der Lehrveranstaltung

<b>Modul SSS-PraktIntKon-M Praktikum im International Kontext</b>		12 ECTS / 360 h
<i>Internship in an International Context</i>		
(seit WS19/20)		
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> Als Praktikum im internationalen Kontext ist ein auf das Berufsfeld der Software Systems Science ausgerichtetes, fachspezifisches Praktikum nachzuweisen, welches im internationalen Kontext, vorzugsweise im Ausland, abzuleisten ist. Das Praktikum kann in einem ausländischen oder international agierenden, inländischen Unternehmen (bzw. einer Forschungseinrichtung) in privater oder öffentlicher Hand absolviert werden. Ein Praktikumsplatz ist so zu wählen, dass den Ausbildungszielen des § 39 Abs. 1 entsprochen wird.		
<b>Lernziele/Kompetenzen:</b> keine		
<b>Sonstige Informationen:</b> Der Nachweis des Praktikums ist durch ein Praktikumszeugnis der Organisationseinheit, bei der das Praktikum absolviert wurde, sowie durch einen schriftlichen Praktikumsbericht zu erbringen. Praktikumszeugnis und Praktikumsbericht sind zusammen beim Modulverantwortlichen einzureichen.		
<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>Prüfung</b> Praktikumsbericht, unbenotet	
<b>Beschreibung:</b> mind. 4 Seiten	

<b>Modul SSS-SEM-M Master-level Seminar in Software Systems Science</b>		3 ECTS / 90 h
<i>Master-Level Seminar in Software Systems Science</i>		
(seit WS14/15)		
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
<b>Inhalte:</b> Independent study and presentation of a topic on the chosen subject area, using scientific methods.		
<b>Lernziele/Kompetenzen:</b> keine		
<b>Sonstige Informationen:</b> Each student must choose one master-level seminar on a subject area of Software Systems Science.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Any recommended prerequisites will be determined and announced by the lecturer offering this module at the beginning of the seminar.		<b>Besondere Bestehensvoraussetzungen:</b> governed by the examination regulations (StuFPO)
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b> 1 Semester

<b>Lehrveranstaltungen</b>	
<b>Master-Level Seminar in a Subject Area of Software Systems Science</b>	<b>2,00 SWS</b>
<b>Lehrformen:</b> Hauptseminar	
<b>Sprache:</b> Englisch	
<b>Angebotshäufigkeit:</b> WS, SS	
<b>Inhalte:</b> Independent study and presentation of a topic on the chosen subject area using scientific methods.	
<b>Literatur:</b> All relevant literature will be announced at the beginning of the seminar by the lecturer offering this module.	

<b>Prüfung</b> Hausarbeit mit Referat	
<b>Zulassungsvoraussetzung zur Modulprüfung:</b> Regular participation in the course.	
<b>Beschreibung:</b> Essay (Hausarbeit) and presentation (Referat) - including a discussion - on the studied topic.  The hand-out and hand-in time for the essay, the expected duration of the presentation, and the weighting of the examination performances between	

---

essay and presentation will be announced by the lecturer at the beginning of the seminar.	
---	--

<b>Modul SSS-Thesis-M Master Thesis in Software Systems Science</b>		30 ECTS / 900 h
<i>Master Thesis in Software Systems Science</i>		
(seit SS19)		
Modulverantwortliche/r: Prof. Ph.D. Michael Mendler		
Weitere Verantwortliche: Professors of Computer Science		
<b>Inhalte:</b>		
Das Modul Masterarbeit hat einen Umfang von 30 ECTS-Punkten und beinhaltet eine schriftliche Prüfung in Form der Masterarbeit und eine mündliche Prüfung in Form des Kolloquiums. Das Thema der Masterarbeit ist einem der in Anhang 2a der Prüfungsordnung genannten Fächer zu entnehmen. Auf Antrag der Prüfungskandidatin bzw. des Prüfungskandidaten kann vom Prüfungsausschuss auch ein Thema aus einem Fach gemäß Anhang 2b zugelassen werden. In diesem Fall ist glaubhaft nachzuweisen, dass das gestellte Thema einen inhaltlichen Bezug zum Masterstudiengang International Software Systems Science aufweist.		
<b>Lernziele/Kompetenzen:</b>		
Durch die Bearbeitung der Abschlussarbeit soll der Nachweis erbracht werden, dass die Prüfungskandidatin oder der Prüfungskandidat eine umfangreiche Forschungsarbeit eigenständig gestalten kann, indem gelerntes Wissen unter Anwendung von Forschungsmethoden auf eine abgeleitete Forschungsfrage angewendet und deren Nutzen beurteilt wird.		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
Die Zulassung setzt voraus, dass dass Module im Umfang von mindestens 60 ECTS-Punkten erfolgreich absolviert wurden.		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
keine		keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
	4.	1 Semester

<b>Prüfung</b>	
schriftliche Hausarbeit / Bearbeitungsfrist: 6 Monate	
<b>Beschreibung:</b>	
Die Note der schriftlichen Hausarbeit wird bei der Ermittlung der Modulnote mit 67 % gewichtet.	

<b>Prüfung</b>	
Kolloquium	
<b>Beschreibung:</b>	
Die Note des Kolloquiums wird bei der Ermittlung der Modulnote mit 33 % gewichtet. Im Kolloquium werden die Hauptergebnisse der Abschlussarbeit verteidigt. Das Kolloquium findet nach Wahl der oder des Studierenden vor oder nach der Bewertung der Abschlussarbeit statt.	
Das Kolloquium hat eine Dauer von 20-60 min. Die genaue Dauer wird bei der	

---

Themenvergabe festgelegt.	
---------------------------	--



<b>Modul SWT-ASV-M Applied Software Verification</b> <i>Applied Software Verification</i>		6 ECTS / 180 h
(seit WS19/20) Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b> This module focuses on the increasingly important field of automated software verification, which aims at increasing the quality of today's complex computer systems. Students will be introduced to modern automated software verification and, in particular, to software model checking, and will be familiarised with a variety of important formal verification concepts, techniques and algorithms, as well as with state-of-the-art verification tools.		
<b>Lernziele/Kompetenzen:</b> On completion of this module, students will be able to thoroughly analyse software using modern software verification tools and understand the state-of-the-art techniques and algorithms that drive cutting-edge development environments offered by major software companies.		
<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 and applying software tools</li> <li>• 30 hrs. working on the assignment (Hausarbeit)</li> <li>• 30 hrs. preparing for the colloquium (Kolloquium)</li> </ul>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b> keine		
<b>Empfohlene Vorkenntnisse:</b> Basic knowledge in algorithms and data structures, mathematical logic and theoretical computer science. Knowledge of the module "Foundations of Software Analysis" (SWT-FSA-B) - or equivalent - is desirable.		<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. Applied Software Verification</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Gerald Lüttgen <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> SS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b> The lectures (Vorlesungen) will address the following topics in automated software verification: (i) state machines, assertions and algorithms for state	

<p>space exploration; (ii) temporal logics for specifying program properties; (iii) model checking using binary decision diagrams; (iv) SAT-based bounded model checking; (v) software model checking based on decision procedures; (vi) abstraction-based software model checking. In addition, state-of-the-art software verification tools will be introduced.</p> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Baier, C., Katoen, J.-P. Principles of Model Checking. MIT Press, 2008.</li> <li>• Biere, A., Heule, M., Van Maaren, H., Walsh, T. Handbook of Satisfiability. IOS Press, 2009.</li> <li>• Clarke, E., Grumberg, O. and Peled, D. Model Checking. MIT Press, 1999.</li> <li>• Huth, M. and Ryan, M. Logic in Computer Science. 2nd ed. Cambridge University Press, 2004.</li> <li>• Kroening, D. and Strichman, O. Decision Procedures: An Algorithmic Point of View. Springer, 2008.</li> </ul>	
<p><b>2. Applied Software Verification</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> 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>Students will practice the various theoretical and practical concepts taught in the lectures (Vorlesungen) by applying them to solve verification problems using several modern model-checking tools, and also by engaging in pen-and-paper exercises. Emphasis will be put on presenting and discussing the solutions to the exercises by and among the students, within the timetabled practicals (Übungen).</p> <hr/> <p><b>Literatur:</b></p> <p>- see the corresponding lectures -</p>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b></p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten</p> <p>Bearbeitungsfrist: 3 Wochen</p> <p><b>Beschreibung:</b></p> <p>Assignment (Hausarbeit) consisting of questions that practice, review and deepen the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen). The assignment is set in English language, while answers may be provided in either English or German.</p> <p>Colloquium (Kolloquium) consisting 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 colloquium can be held electively in English or German language.</p>	

<b>Modul SWT-PCC-M Principles of Compiler Construction</b>		6 ECTS / 180 h
<i>Principles of Compiler Construction</i>		
(seit WS19/20)		
Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b>		
The module teaches the theoretical and practical principles of compiler construction, from lexical analysis and parsing, to semantic analysis, to code generation and optimisation.		
<b>Lernziele/Kompetenzen:</b>		
On completion of this module, students will be familiar with all phases of a modern compiler – from lexical analysis and parsing, to semantic analysis and finally code generation and code optimisation – and will have a deep understanding of the workings of compilers. As a result, students will be able to use compilers more effectively and learn better debugging practices. Students will also be able to start building compilers on their own.		
<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. reviewing the lectures, including researching and studying material from additional sources</li> <li>• 30 hrs. attending practicals (Übungen)</li> <li>• 30 hrs. preparing and reviewing the practicals, including researching and 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>		<b>Besondere Bestehensvoraussetzungen:</b>
Basic knowledge in programming languages, in the theoretical foundations of Computer Science (especially in language theory and automata theory) and in algorithms and data structures.		keine
<b>Angebotshäufigkeit:</b> WS, jährlich	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester

<b>Lehrveranstaltungen</b>	
<b>1. Principles of Compiler Construction</b> <b>Lehrformen:</b> Vorlesung <b>Dozenten:</b> Prof. Dr. Gerald Lüttgen <b>Sprache:</b> Englisch/Deutsch <b>Angebotshäufigkeit:</b> WS, jährlich	<b>2,00 SWS</b>
<b>Inhalte:</b>	
Students will be familiarised with a variety of theoretical and practical concepts, techniques and algorithms employed in compiler construction, which reach from language theory, to automata theory, and to data flow analysis. The lectures will	

<p>focus on the following aspects of compiler construction: lexical analysis, parsing, abstract syntax, semantic analysis, code generation and code optimisation.</p> <hr/> <p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Louden, K. C. Compiler Construction: Principles and Practice. Course Technology, 1997.</li> <li>• Aho, A. V., Lam, M. S., Sethi, R. and Ullman, J. D. Compilers: Principles, Techniques, and Tools, 2nd ed. Pearson, 2007.</li> <li>• Fischer, C. N., Cytron, R. K. and LeBlanc Jr., R. J. Crafting a Compiler. Pearson, 2010.</li> <li>• Muchnick, S. S. Advanced Compiler Design and Implementation, Morgan Kaufmann, 1997.</li> </ul>	
<p><b>2. Principles of Compiler Construction</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> WS, jährlich</p> <hr/> <p><b>Inhalte:</b></p> <p>Students will practice the theoretical concepts taught in the lectures by applying them to a variety of exercises, so that they can appreciate the diverse range of foundations that make modern programming languages possible. The exercises will largely be pen-and-paper exercises but may also involve some work using computers. Emphasis will be put on presenting and discussing the solutions to the exercises by and among the students, within the timetabled practicals (Übungen). Students can gain further practical experience in compiler construction by attending one of the modules "Masterprojekt Softwaretechnik und Programmiersprachen" (SWT-PR1-M) or "Masters Project in Software Systems Science" (SWT-PR2-M).</p> <hr/> <p><b>Literatur:</b></p> <p>- see the corresponding lectures -</p>	<p><b>2,00 SWS</b></p>
<p><b>Prüfung</b></p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 20 Minuten</p> <p>Bearbeitungsfrist: 3 Wochen</p> <p><b>Beschreibung:</b></p> <p>Assignment (Hausarbeit) consisting of questions practicing, reviewing and deepening the knowledge transferred in the lectures (Vorlesungen) and practicals (Übungen). The assignment is set in English language, while answers may be provided in either English or German.</p> <p>Colloquium (Kolloquium) consisting of questions testing the knowledge transferred in the lectures (Vorlesungen) and practicals (Übungen), on the basis of the submitted solutions to the assignment (Hausarbeit). The colloquium can be held electively in English or German language.</p>	

<b>Modul SWT-PR2-M SWT Masters Project in Software Systems Science</b>		9 ECTS / 270 h
<i>SWT Masters Project in Software Systems Science</i>		
(seit WS19/20)		
Modulverantwortliche/r: Prof. Dr. Gerald Lüttgen		
<b>Inhalte:</b>		
<b>Attention: The module SWT-PR2-M cannot take place in winter semester 2019/20!</b>		
<p>Topics in Software Systems Science are carried out individually or in 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 academic literature and relevant technologies and 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 tools for software verification, which requires the prior attendance of the module "Applied Software Verification" (SWT-ASV-M), or equivalent knowledge. Another example would be designing and implementing a compiler of a small programming language in either an imperative, object-oriented or functional language, which requires the prior attendance of the module "Principles of Compiler Construction" (SWT-PCC-M), or equivalent knowledge.</p>		
<b>Lernziele/Kompetenzen:</b>		
<p>Students will deepen their knowledge regarding the conceptual problems that arise when carrying out scientific projects related to Software Systems Science, and regarding approaches to possible solutions. Students will also gain important experience in carrying out such projects, from project planning, to the abstract and concrete design, to the realization, to the documentation of results in a scientific project report.</p>		
<b>Sonstige Informationen:</b>		
<p>The main language of instruction is English. The module may be delivered in German if all participating students are fluent in German. A regular participation in the project meetings is necessary.</p> <p>The total workload of 270 hrs. is split approximately as follows:</p> <ul style="list-style-type: none"> <li>• 10 hrs. participating in introductions to and tutorials on methods, software tools, and giving presentations on the project status</li> <li>• 20 hrs. completing the exercises for bonus points</li> <li>• 195 hrs. researching and familiarization with the project topic and conducting the project work</li> <li>• 45 hrs. compiling a project report (Assignment/Hausarbeit) and preparation of the Colloquium (Kolloquium).</li> </ul> <p>The project report can be written/composed in either English or German.</p>		
<b>Zulassungsvoraussetzung für die Belegung des Moduls:</b>		
keine		
<b>Empfohlene Vorkenntnisse:</b>		<b>Besondere Bestehensvoraussetzungen:</b>
Basic knowledge in software engineering and programming languages, knowledge in the subject matter of the project topic.		keine
<b>Angebotshäufigkeit:</b> WS, SS	<b>Empfohlenes Fachsemester:</b>	<b>Minimale Dauer des Moduls:</b>
		1 Semester

<b>Lehrveranstaltungen</b>	
<p><b>SWT Masters Project in Software Systems Science</b></p> <p><b>Lehrformen:</b> Übung</p> <p><b>Dozenten:</b> Prof. Dr. Gerald Lüttgen</p> <p><b>Sprache:</b> Englisch/Deutsch</p> <p><b>Angebotshäufigkeit:</b> WS, SS</p>	<b>6,00 SWS</b>
<p><b>Lernziele:</b></p> <p>To be announced at the beginning of the project.</p>	
<p><b>Inhalte:</b></p> <p>Conduct of the project, accompanied by tutorials and regular project meetings.</p>	
<p><b>Literatur:</b></p> <p>To be announced at the beginning of the project.</p>	

<p><b>Prüfung</b></p> <p>Hausarbeit mit Kolloquium / Prüfungsdauer: 30 Minuten</p> <p>Bearbeitungsfrist: 12 Wochen</p> <p><b>Zulassungsvoraussetzung zur Modulprüfung:</b></p> <p>Regular participation in the practicals.</p> <p><b>Beschreibung:</b></p> <p>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).</p>	
---	--

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A1 Software Systems Science</b>			<b>30 - 48</b>		
In den Modulgruppen A1 und A2 sind Module im Gesamtumfang von 48 ECTS-Punkten unter Einhaltung der für die Modulgruppen geltenden Mindest- und Höchstgrenzen zu absolvieren.					
<b>Teilmodulgruppe: Pflichtbereich</b>			<b>30</b>		
DSG-DSAM-M	Distributed Systems Architecture and Middleware	WS, jährlich	6	2 Vorlesung 2 Übung	Hausarbeit mit Kolloquium 3 Monate 15 Minuten
KTR-GIK-M	Grundbausteine der Internet-Kommunikation	SS, jährlich(on demand also WS)	6	4 Vorlesung und Übung	Hausarbeit mit Kolloquium 4 Monate 30 Minuten
MOBI-DSC-M	Data Streams and Complex Event Processing	WS, jährlich(1)	6	2 Vorlesung 2 Übung	mündliche Prüfung 15 Minuten schriftliche Prüfung (Klausur) 60 Minuten
PSI-AdvaSP-M	Advanced Security and Privacy	SS, jährlich(1)	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
SWT-PCC-M	Principles of Compiler Construction	WS, jährlich	6	2 Vorlesung 2 Übung	Hausarbeit mit Kolloquium 3 Wochen 20 Minuten
<b>Teilmodulgruppe: Wahlpflichtbereich</b>			<b>0 - 18</b>		
DSG-DistrSys-M	Distributed Systems	SS, jährlich(2020)	6	2 Vorlesung 2 Übung	Hausarbeit mit Kolloquium 3 Monate 15 Minuten
DSG-SOA-M	Service-Oriented Architecture and Web Services	SS, jährlich	6	2 Vorlesung 2 Übung	Hausarbeit mit Kolloquium 3 Monate 15 Minuten
DSG-SRDS-M	Selected Readings in Distributed Systems	WS, SS	3	2 Vorlesung/Seminar	Hausarbeit mit Kolloquium 4 Monate

## Modultabelle

Gdl-AFP-M	Advanced Functional Programming	SS, jährlich	6	2 Vorlesung 2 Übung	20 Minuten schriftliche Prüfung (Klausur) 90 Minuten
Gdl-FP-M	Functional Programming	WS, jährlich	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
KTR-MAKV-M	Modellierung und Analyse von Kommunikationsnetzen und Verteilten Systemen	SS, jährlich	6	4 Vorlesung und Übung	mündliche Prüfung 30 Minuten
KTR-MMK-M	Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen	SS, jährlich	6	4 Vorlesung und Übung	mündliche Prüfung 30 Minuten
KTR-Mobi-M	Mobilkommunikation	WS, jährlich	6	4 Vorlesung und Übung	mündliche Prüfung 30 Minuten
MOBI-ADM-M	Advanced Data Management	SS, jährlich(1)	6	2 Vorlesung 2 Übung	mündliche Prüfung 15 Minuten schriftliche Prüfung (Klausur) 60 Minuten
SWT-ASV-M	Applied Software Verification	SS, jährlich	6	2 Vorlesung 2 Übung	Hausarbeit mit Kolloquium 3 Wochen 20 Minuten



## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A2 Domain-specific Software Systems Science</b>			<b>0 - 18</b>		
In den Modulgruppen A1 und A2 sind Module im Gesamtumfang von 48 ECTS-Punkten unter Einhaltung der für die Modulgruppen geltenden Mindest- und Höchstgrenzen zu absolvieren.					
EESYS-DAE-M	Data Analytics in der Energieinformatik	WS, jährlich	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
EESYS-ES-M	Energieeffiziente Systeme	SS, jährlich	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
Gdl-CSNL-M	Computational Semantics of Natural Language	SS, jährlich(1)	6	4	Portfolio
HCI-MCI-M	Mensch-Computer-Interaktion	SS, jährlich	6	2 Vorlesung 2 Übung	mündliche Prüfung 30 Minuten schriftliche Prüfung (Klausur) 90 Minuten
HCI-US-B	Ubiquitäre Systeme	WS, jährlich	6	2 Vorlesung 2 Übung	mündliche Prüfung 30 Minuten schriftliche Modulprüfung (Klausur) 90 Minuten
KogSys-ML-M	Lernende Systeme (Machine Learning)	WS, jährlich	6	2 Vorlesung 2 Übung	schriftliche Prüfung (Klausur) 90 Minuten
SME-STE-M	Introduction to Knowledge Representation: Space, Time, Events	WS, jährlich	6	2 Vorlesung 2 Übung	mündliche Prüfung 20 Minuten
SNA-OSN-M	Projekt zu Online Social Networks	WS, jährlich	6	4 Übung	Hausarbeit mit Kolloquium 4 Monate 30 Minuten

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
<b>A3 Seminar and Project</b>			<b>12</b>		
<b>Teilmodulgruppe: Seminar</b>			<b>3</b>		
SSS-SEM-M	Master-level Seminar in Software Systems Science	WS, SS(1)	3	2 Hauptseminar	Hausarbeit mit Referat
<b>Teilmodulgruppe: Projekt</b>			<b>9</b>		
<b>Attention: The module SWT-PR2-M cannot take place in winter semester 19/20!</b>					
MOBI-PRS-M	Master Project Mobile Software Systems (SoSySc)	SS, jährlich(1)	9	6 Übung	Hausarbeit mit Kolloquium
DSG-Project-M	Masterprojekt Verteilte Systeme	WS, SS	9	6 Übung	Hausarbeit mit Kolloquium 3 Monate 20 Minuten
SWT-PR2-M	SWT Masters Project in Software Systems Science	WS, SS	9	6 Übung	Hausarbeit mit Kolloquium 12 Wochen 30 Minuten
KTR-SSSProj-M	KTR Masterprojekt Software Systems Science	WS, SS(Beginn WS)	9	6 Projekt	Hausarbeit mit Kolloquium 4 Monate 30 Minuten
PSI-ProjectCAD-M	Project Complex Attacks and Defenses	WS, SS(1)	9	6 Übung	Hausarbeit mit Kolloquium 3 Monate 30 Minuten
PSI-ProjectSP-M	Project Security and Privacy	WS, SS(1)	9	6 Übung	Hausarbeit mit Kolloquium 3 Monate 30 Minuten

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
	<b>A4 Master Thesis</b>		<b>30</b>		
SSS-Thesis-M	Master Thesis in Software Systems Science	WS, SS	30		schriftliche Hausarbeit 6 Monate Kolloquium

## Modultabelle

ID	Modul	Semester	ECTS	SWS	Prüfung
	<b>A5 International Experience</b>		<b>30</b>		
	<p>In der Modulgruppe A5 sind Module im Umfang von 30 ECTS-Punkten zu absolvieren.</p> <p>Zur Auswahl stehen die im Folgenden genannten Module der Wahlpflichtbereiche 5a bis 5c. Soweit in diesen Wahlpflichtbereichen insgesamt weniger als 30 ECTS-Punkte erbracht werden, sind weitere, noch nicht absolvierte Module der Modulgruppen A1, A2 oder A3 zu absolvieren.</p>				
	<b>Teilmodulgruppe: Module aus dem gelenkten Auslandsstudium</b>		<b>0 - 30</b>		
	<p>Im Wahlpflichtbereich a können Module im Umfang von 0 bis 30 ECTS-Punkten eingebracht werden, die im Rahmen eines gelenkten Auslandsstudiums an einer ausländischen Hochschule absolviert werden, sofern sie sich wesentlich von den nach Vorgabe der vorliegenden Ordnung zu absolvierenden Modulen unterscheiden und fachsystematisch den Modulgruppen A1, A2 oder A3 zugeordnet werden können.</p>				
	<b>Teilmodulgruppe: Praktikum im internationalen Kontext</b>		<b>0 - 12</b>		
SSS-PraktIntKon- M	Praktikum im International Kontext	WS, SS(1)	12		Praktikumsbericht
	<b>Teilmodulgruppe: Fremdsprachen</b>		<b>0 - 18</b>		
	<p>Im Wahlpflichtbereich 5c können Module im Umfang von 0 bis 18 ECTS-Punkten aus dem Angebot des Sprachenzentrums Bamberg absolviert werden. Ausgenommen sind Module der englischen Sprache sowie Module der Sprache, in der die Hochschulzugangsberechtigung erworben wurde.</p> <p>Einzelheiten, insbesondere die zur Auswahl stehenden Module sowie die jeweils abzulegenden Modulprüfungen und Modulteilprüfungen sind in der Prüfungsordnung und dem Modulhandbuch für sprachpraktische Module der Otto-Friedrich-Universität Bamberg festgelegt.</p>				