Faculty of Information Systems and Applied Computer Sciences (WIAI)

Undergraduate and Graduate Studies in Information Systems and Computer Science

Module Handbook and Student Guide Academic Year 2021-2022

WIAI

www.uni-bamberg.de/wiai

Contact

Michael Mendler
International Affairs Representative
Faculty of Information Systems and Applied Computer Sciences
The Otto-Friedrich University of Bamberg
An der Weberei 5
D-96047 Bamberg
Germany

Phone: ++49 (0)951 863-2828
Email: michael.mendler@uni-bamberg.de

Legal notice

Contents

1 Introduction and General Information

This document lists all modules offered in the Faculty of Information Systems and Applied Computer Sciences during the academic year 2021-2022.

To prevent problems arising from late changes on these offerings, you are advised to confirm module specifications and schedules through the research groups' web pages or directly with the responsible teaching staff. There may also be new courses becoming available on short notice. Please see the faculty's web pages for relevant announcements. The official German module handbooks for the various degree courses can be accessed through the following links:

- BSc Angewandte Informatik
- BSc International Information Systems Management
- BSc Software Systems Sciences
- BSc Information Systems (Wirtschaftsinformatik)
- MSc Applied Computer Sciences
- MSc Computing in the Humanities
- MSc International Software Systems Science
- MSc Information Systems (Wirtschaftsinformatik)
- MSc International Information Systems Management
- MSc Wirtschaftspädagogik

1.1 Fees and Registration

All modules are currently open free of charge to foreign guest students who will study at Bamberg University within the frame of a partnership exchange programme, such as ERASMUS+. There are no tuition or bench fees. Enrolment with the University incurs a nominal registration fee covering administration charges, student union membership (*Studentenwerk*) and the City of Bamberg travel ticket.

Information on the registration and enrolment process may be obtained from the Academic Exchange Office (*Akademisches Auslandsamt*, see address below) who will also be able to advise you on any exchange scheme that may exist between Bamberg University and your home institution.

Once admitted to and enrolled with Bamberg University you do not need to register for attending a teaching module. Feel free to sit in and participate in any course offering that fits your educational needs and time table. Be aware, though, that some courses may have entry requirements and/or class size restrictions.

1.2 Teaching Times

The academic year 2021-2022 consists of two teaching periods:

Winter Semester: 18th October 2021 – 11th February 2022,
Summer Semester: 25th April 2022 – 29th July 2022.

1.3 Assessment

The course assessment is done mostly by written exams and optionally also by way of homework assignments and/or lab practicals. In a number of cases, typically for graduate level modules, the final exam is oral.

Final written exams are usually held immediately after the end of the semester, i.e. February/March for the Winter Semester and end of July/August for the Summer Semester. Make sure you plan your travelling so you are in Bamberg during the exams.

Be aware that there are firm deadlines for exam registration some time at the end of the first half of the semester. Watch out for the emails announcing the registration period and check up with your class mates if you are not sure. There is a short period of time during which you can deregister from an exam. Once this period has passed and you are registered you **must take** the exam at the specified day and time. Also, if you miss the online registration deadline, then you **cannot participate** in the exam.

There is one more thing to know: for written exams the registration in FlexNow! is sufficient. For oral exams, however, you also need to arrange an exam time with the lecturer in addition to the FlexNow! registration.

If for some reason you cannot attend the regular written exam, say because you are required to return home early, talk to the course lecturer before the FlexNow! registration period has passed. There are two options:

> **Option 1:** The course lecturer may be able to arrange an oral exam for you at an earlier date instead. In this case, because you are not writing the official exam, you **must not register** with the FlexNow! system.
>
> **Option 2:** It is sometimes possible that we schedule the regular exam at your Home University on the same day and at the same time when it is written in Bamberg. For this option you **must register** via FlexNow!

Whether such options are available is entirely at the discretion of the course lecturer.

The official exam language is German, but many courses may offer written or oral exams in English if required. Some modules are fully taught in English, some only at the discretion of the lecturer. If you need to be set an English exam for a module delivered in German you should contact the module lecturer early to find out if this is possible. The description of each module listed below in this booklet indicates if all or some part of the module is delivered in English.

1.4 Workload

The module descriptions below specify the total module workload in terms of ECTS (European Credit Transfer System) credit points according to the following approximate accounting scheme:

- 1 ECTS = 25-30 hrs total student workload (all inclusive)
- 30 ECTS = total module load per semester
- 6 ECTS = single standard course module of 4 contact hrs/week, combining lectures+tutorials

1.5 Course Levels and Teaching Format

In line with our traditional Diploma degree structure, modules are taught at 2 levels:

- Basic Studies

  These are foundational and introductory courses in the general disciplines of Information Systems, Applied Computer Science and Software Systems Science corresponding to the 1st and 2nd year of the undergraduate BSc programmes.

- Advanced Studies

  These are introductory courses to specialised fields within Information Systems, Applied Computer Science and Software Systems Science corresponding to the 3rd and 4th year of the BSc degree and advanced modules in particular research areas which correspond to the 1st and 2nd year of the graduate MSc programmes.

As our guest students you may attend modules at any of these levels. It is your responsibility to judge if your background will be sufficient to participate successfully in the course. Also, whether or not the credits you earn are valid towards your home degree, is not decided by us, but by your home institution.

Keep in mind, however, that graduate level modules normally assume a significant amount of background in the relevant subject area.

Most modules are based on combined lectures and tutorials. Some courses may also involve lab classes, excursions, blended learning and other teaching arrangements. Research groups regularly offer advanced level seminars and project modules on varying research topics. These may have special entry prerequisites.

1.6 Other Information

The Academic Exchange Office provides information on accommodation, living expenses, language courses and many other aspects of student life at Bamberg.

Academic Exchange Office (*Akademisches Auslandsamt*)

Mrs. Stephanie Hofmann

Secretary - Foreign Student Affairs

Akademisches Auslandsamt

Otto-Friedrich-Universität Bamberg

D-96047 Bamberg, Germany

Kapuzinerstraße 25

Tel: ++49 (0)951 863-1051

Fax: ++49 (0)951 863-1054

Email: auslandsamt@uni-bamberg.de

URL: http://www.uni-bamberg.de/auslandsamt/

You are also welcome to contact the International Affairs Representative of the WIAI Faculty (see address page 2).

2 Introducing the Faculty's Teaching and Research Groups

2.1 MI - Media Informatics

Prof. Dr. Andreas Henrich
Chair of Media Informatics
Office 02.031
An der Weberei 5
96047 Bamberg
Tel.: +49-951 / 863-2850
Fax: +49-951 / 863-2852
E-Mail: andreas.henrich@uni-bamberg.de
Internet: http://www.uni-bamberg.de/minf/

Media Informatics aims at a purposeful application of single media types and multimedia systems in various application areas. It considers aspects like media technique, media design, media storage and retrieval as well as the use of media with a focus on the development of multimedia systems. The main focus of this chair is on media storage and retrieval and the development of domain specific multimedia applications.

2.2 KTR - Communication Systems and Computer Networks

Prof. Dr. Udo R. Krieger
Head of Computer Networks Group
Office 05.037
An der Weberei 5
96047 Bamberg
Tel.: +49-951 / 863-2820
Fax: +49-951 / 863- 5528
E-Mail: udo.krieger@ieee.org
Internet: http://www.uni-bamberg.de/ktr

Research and development of the group is devoted to traffic and network management of current telecommunication networks and future IP based fixed and mobile computer networks. Current research topics include the evaluation of resource management processes in wireless local IP networks, the development of QoS management architectures for IP communication networks, teletraffic theory and performance evaluation of such distributed systems, the statistical analysis and characterization of Internet traffic, and the estimation of corresponding generic model parameters.

2.3 SWT - Software Technologies and Programming Languages

Prof. Dr. Gerald Lüttgen

Head of Software Technologies Research Group

Office 03.014

An der Weberei 5

96047 Bamberg

E-Mail: info@swt-bamberg.de

Internet: www.uni-bamberg.de/swt/

The group's teaching is heavily influenced by research and currently encompasses the modules Software Engineering, Project Management, Compiler Construction, Imperative Programming and Parallel Programming, with accompanying seminars and student projects. The group's research comprises the foundations and practice of software specification, verification and analysis. Foci of application are concurrent, reactive and embedded software. A further competence of the research group is requirements engineering.

2.4 GdI - Foundations of Computer Science

Prof. Michael Mendler, PhD (Edinburgh)

Informatics Theory Group

Office 05.041

An der Weberei 5

96047 Bamberg

Tel.: +49 (0) 951 / 863-2828

Fax: +49 (0) 951 / 863-1200/5861

E-Mail: michael.mendler@uni-bamberg.de

Internet: http://www.gdi.uni-bamberg.de

The group teaches the foundational aspects of computer science in all degree programmes, such as logic, automata and formal language theory, cryptography, functional programming and the theory of distributed systems. In our research we are mainly concerned with constructive modal logic and type theory and their applications as well as the semantics of synchronous programming languages.

2.5 KInf - Computing in the Cultural Sciences

Prof. Dr. Christoph Schlieder
Chair of Computing in the Cultural Sciences
Office 02.033
An der Weberei 5
96047 Bamberg
Tel.: +49 (0) 951 / 863-2840
Fax: +49 (0) 951 / 863-5841
E-Mail: christoph.schlieder@uni-bamberg.de
Internet: http://www.uni-bamberg.de/kinf/

In research and teaching, we focus on computational issues relevant to the cultural sciences. At our laboratory we develop software solutions that assist, for instance, preservation scientists working with built heritage or sociologists studying web-based communication processes. Technologically speaking, our software relies on methods from semantic information processing that we continue to improve. Our key areas of interest are Geoinformation systems and services, Digital libraries and archives, Mobile assistance systems, and Computer-mediated communication.

2.6 KogSys - Cognitive Systems

Prof. Dr. Ute Schmid
Head of Cognitive Systems Group
Applied Computer Science
Office 05.043
An der Weberei 5
96047 Bamberg
Tel.: +49-951 / 863-2860
Fax: +49-951 / 863-2862
E-Mail: ute.schmid@uni-bamberg.de
Internet: http://www.uni-bamberg.de/kogsys

In the research domain Cognitive Systems (CogSys) we are concerned with the development of approaches, concepts, and methods for design, description, construction and analysis of intelligent systems based on cognitive principles. Our research strategy is to combine empirical studies of cognitive phenomena, development of algorithms, and their testing in different areas of application. Main topics of our group are induction and learning as well as planning and problem solving in single- and multi-agent settings. Especially, we are interested in the inductive synthesis of recursive functional programs from incomplete specifications (e.g., input/output examples) which can be seen as a general approach to learning productive rules from experience. Furthermore, we investigate analogical reasoning as a powerful approach to problem solving as a special mechanism of knowledge acquisition. Application areas are, for example, support of human problem solvers in the domains of software development, classifier learning for medical diagnostics, quality control, decision support or incident mining and assistant systems for activities of daily life.

2.7 ISDL - Information Systems in the Service Industry

Prof. Dr. Tim Weitzel

Chair of Information Systems, esp. Information systems in the service industry

Office 04.040

An der Weberei 5

96047 Bamberg

Tel.: +49 /0951 / 863-2870

E-Mail: tim.weitzel@uni-bamberg.de

Internet: http://www.uni-bamberg.de/isdl

The four main research areas of the ISS team are (1) Business-IT Alignment and Business Value of IT, (2) (Social) Networks and IT, (3) IT Adoption and Usage and (4) Outsourcing Management. You can find a detailed description of our research projects in the section "for researchers". Our research results have been published in scientific journals and conferences.

2.8 DSG - Practical Computer Science (Distributed Systems)

Prof. Dr. Guido Wirtz

Chair of Practical Computer Science

Distributed Systems Group

Office 03.016

An der Weberei 5

96047 Bamberg

Tel.: +49 /0951 / 863-2527

Fax: +49 /0951 / 863- 2529

E-Mail: guido.wirtz@uni-bamberg.de

Internet: http://www.uni-bamberg.de/en/pi

Besides introductionary courses for 1th and 2nd year students, our teaching activities put an emphasis on combining the theoretical background of distributed systems with knowledge about middleware and architecture for complex systems. This is done by advanced courses and seminars as well as practical labs to get hands-on experience with real-life systems.

The DSG's research directions are centered around issues regarding the software development for complex, esp. distributed, systems on all levels. Our current research activities are focussed on service-oriented architectures, service eco systems, enterprise application integration and B2Bi, seamless transition from business processes to their implementation in SOA-like settings, visual design and programming languages as well as visualization of complex software systems.

2.9 HCI - Human-Computer Interaction

Prof. Dr. Tom Gross

Chair of Human-Computer Interaction

An der Weberei 3, Room WE5/01.032

D-96047 Bamberg

Tel.:	+49-951 / 863-3940
Fax:	+49-951 / 863-3945
E-Mail:	tom.gross@uni-bamberg.de
Internet:	http://www.uni-bamberg.de/hci

Human-Computer Interaction (HCI) aims at optimally supporting users through technology (mainly computer technology) by amplifying their strengths and compensating their weaknesses. The Special Interest Group HCI of the German Informatics Society provides the following definition: “The field of Human-Computer Interaction comprises the analysis, design, and evaluation of human- and task-centred computer applications" (in German). In the context of HCI interactive systems are often mentioned—an interactive (computer-) system thereby is described as a unity consisting of software and hardware that receives input from users and gives immediate feedback. The usability of interactive systems can be evaluated along three factors: effectiveness (accuracy and completeness with which users achieve their goals), efficiency (resources expended by users to achieve these goals), and satisfaction (the users' positive attitudes towards the use of the system).

2.10 SNA - Social Networks

Prof. Dr. Oliver Posegga

Chair in Information Systems, esp. Social Networks

Office 05.128

An der Weberei 5

96047 Bamberg

Tel.:	+49 (0) 951 / 863-2890
Fax:	+49 (0) 951 / 863-2872
E-Mail:	oliver.posegga@uni-bamberg.de
Internet:	http://www.uni-bamberg.de/sna

We dedicate our work to research and education on the role of information systems in social networks and the use of social media (e.g. blogs, wikis) for managing knowledge within and between organizations as well as on digital transformation.
We conduct theoretical and empirical research and collaborate with industry partners in these fields.

2.11 EESYS - Energy Efficient Systems

Prof. Dr. Thorsten Staake

Chair of Information Systems, esp. Energy Efficient Systems

Office 02.057

An der Weberei 5

96047 Bamberg

Tel.: +49 /0951 / 863-2076

E-Mail: thorsten.staake@uni-bamberg.de

Internet: http://www.uni-bamberg.de/eesys

The Energy Efficient Systems Group focuses on the development of Information Systems (IS) for supporting and motivating sustainable consumer behavior. Particular consideration is given to smart metering and smart grid infrastructure and the design of related systems that enable energy efficiency services for private households. With a clear focus on the development and assessment of applications that are applicable to the mass market, the group closely works together with both, industry partners and startup companies. The research results build an important cornerstone of the teaching activities at bachelor and master level.

2.12 SME - Smart Environments

Prof. Dr. Diedrich Wolter

Chair of Smart Environments

Office 03.040

An der Weberei 5

96047 Bamberg

Tel.: +49-951 / 863 2897

E-Mail: diedrich.wolter@uni-bamberg.de

Internet: http://www.uni-bamberg.de/

Smart environments is a young area of research in applied artificial intelligence (AI). The field draws its motivations from recent advancements in AI as well as in technology (disappearing computers, sensors) and human-centered computing. At the university of Bamberg, we contribute by tackling the following research questions:

- How can sensor data be interpreted to obtain useful knowledge?
- How can knowledge about space, time, events, and context be represented?
- And how can we reason with this knowledge in order to obtain smart decisions?

2.13 IIS - Industrial Information Systems

Prof. Dr. Sven Overhage

Chair of Information Systems, esp. Industrial Information Systems

Office 04.042

An der Weberei 5

96047 Bamberg

Tel.:	+49 /0951 / 863-2910
E-Mail:	sven.overhage@uni-bamberg.de
Internet:	http://www.uni-bamberg.de/iis

The Chair of Industrial Systems focuses on the design and the operation of industrial information systems, which are the backbone of production and commerce businesses. We offer courses for bachelor, master, and PHD students alike. Amongst others, our courses focus on the development and design of application systems, enterprise architecture management, electronic business, intra-organizational systems, and modular and on-demand systems.

2.14 MOBI - Mobile Software Systems / Mobility

Prof. Dr. Daniela Nicklas

Chair of Information Systems, esp. Mobile Software Systems / Mobility

Office 05.128

An der Weberei 5

96047 Bamberg

Tel.:	+49 /0951 / 863-3670
E-Mail:	dnicklas@acm.org
Internet:	http://www.uni-bamberg.de

The MOBI group is led by Prof. Dr. Daniela Nicklas and focuses on data management for mobile systems, data stream management/complex event processing and development support for sensor-based applications, in the area of smart cities.

2.15 PSI - Privacy and Security in Information Systems Group

Prof. Dr. Dominik Herrmann

Chair of Information Systems and Applied Computer Sciences,

esp. Privacy and Security Information Systems

Office 05.030

An der Weberei 5

96047 Bamberg

Tel.: +49 /951 / 2661

E-Mail: Dominik.Herrmann@uni-bamberg.de

Internet: https://www.uni-bamberg.de/en/psi/

The focus of the PSI Group is the protection of information systems and the protection of privacy with technical mechanisms. To this end, the PSI Group analyzes and evaluates existing systems and develops protective mechanisms. The PSI Group cooperates with working groups in the fields of machine learning (inference attacks, online tracking), law (data protection, law enforcement) and ethics (value-oriented system design).

2.16 ISM – Information Systems Management

Prof. Dr. Daniel Beimborn

Chair of Information Systems, esp. Information Systems Management

Office 01.029

An der Weberei 5

96047 Bamberg

Tel.: +49 /951 / 2512

E-Mail: Daniel.Beimborn@uni-bamberg.de

Internet: https://www.uni-bamberg.de/en/wi/chair-of-information-systems-especially-information-systems-management/home/

In research and teaching the Chair of ISM deals with questions concerning the management of information systems and technologies as well as with the challenges of digital innovation and transformation. The management of the information systems (IS) of an organization - consisting of those technical and personnel components, which are involved in the production, processing and use of information - is in particular in the 'age of digitization' an elementary component of successful organization management.

3 Module Descriptions

The following appendix titled “Module Handbook – International Studies” describes in detail all modules scheduled to run during 2021-2022.
The module handbook starts with an index of all modules listed by the area of studies. In Section 1 of the list (“International Studies taught in English (on demand)”) you find all modules that are offered in English either regularly or on demand. In the latter case, since the lecture may be in German if all students are German-speaking, you need to tell the lecturer before the semester if you require English tuition. Modules listed in Section 2 of the list (“Exams in English on demand, though course material often in German, may sometimes be available in English”) are delivered in German but (again ``on demand’’) are examined in English if requested. To find out more information please consult the detailed module descriptions or contact the module lecturer. Section 3 (“General Studies in German only, exams in English on request”) lists those modules which are available only in German. Within each Section the modules are organised by subject group (Applied Computer Science, Computer Science, Information Systems) and further by the name of the Teaching and Research Group who are responsible for the offering.

The index of a single module lists its acronym, its title, how many ECTS credit points it comprises, in which semester it is offered and on which page you can find its full description. For example, here is a description of the information related to the module “KogSys-ML-M”:

1	2	3	4	5
KogSys-ML-M	Machine Learning	6,00 ECTS	every winter semester	page number

1. “KogSys-ML-M”:
   a. “KogSys” stands for the research group that provides the module; in this case, this is “Cognitive Systems” (*Kognitive Systeme*)
   b. “ML” is the short form of the module title; here, this is Machine Learning
   “M” stands for “Master” which means the module is suggested for graduate students. The ending “B” indicates the recommendation that the module should be attended by undergraduate students.
   NOTE: International Exchange Students may attend any module offered, at undergraduate or graduate level.

2. “Machine Learning”: This is the title of the module
3. “6,00 ECTS”: ECTS indicate the work load for the module (see “Sec. 1.4 Workload”)
4. The module is offered every winter semester
5. The page on which you find a detailed module description.

Modules

Index by areas of study

1) International studies taught in English (on demand) (Bereich)

a) Applied Computer Science (Subject Group)

aa) Smart Environments (Subject)

bb) Cognitive Systems (Subject)

cc) Cultural Computing (Subject)

dd) Human-Computer Interaction (Subject)

b) Information Systems (Subject Group)

aa) Information Systems Management (Subject)

bb) Energy Efficient Systems (Subject)

cc) Information Systems and Services (Subject)

dd) Social Networks (Subject)

c) Computer Science (Subject Group)

aa) Privacy and Security in Information Systems Group (Subject)

bb) Communication Services, Telecommunication Systems, and Computer Networks (Subject)

cc) Distributed Systems Group (Subject)

dd) Foundations of Computer Science (Subject)

ee) Mobile Software Systems /Mobility (Subject)

ff) Software Technologies Research Group (Subject)

2) Course language German, exams in English on demand, course material may be available in English (Bereich)

a) Information Systems (Subject Group)

aa) Social Networks (Subject)

bb) Information Systems Management (Subject)

cc) Industrial Information Systems (Subject)

dd) Information Systems and Services (Subject)

ee) Energy Efficient Systems (Subject)

b) Computer Science (Subject Group)

aa) Distributed Systems Group (Subject)

bb) Foundations of Computer Science (Subject)

c) Applied Computer Science (Subject Group)

aa) Media Informatics (Subject)

bb) Smart Environments (Subject)

Module AI-KI-B Introduction to Artificial Intelligence *Einführung in die Künstliche Intelligenz*	6 ECTS / 180 h 42 h Präsenzzeit 138 h Selbststudium

(since SS21)
Person responsible for module: Prof. Dr. Diedrich Wolter
further responsible : Schmid, Ute, Prof. Dr.

Contents:

Dieses Modul bietet Studierenden einen Überblick über das Fachgebiet der Künstlichen Intelligenz (KI) und bietet eine Einführung in elementare Konzepte, Methoden und Algorithmen wie etwa Wissensrepräsentation, Suche, Wahrnehmung und Handlungsplanung. Die vermittelten Inhalte bilden eine Grundlage für kognitive und smarte Systeme sowie für wichtige Teile der Informatik und Wirtschaftsinformatik.

Themen:

- Problemstellungen im Fachgebiet KI
- KI-Programmierung
- intelligente Agenten
- Wissensrepräsentation und Logik
- Suche im Problemraum
- maschinelles Lernen
- Wahrnehmung
- Unsicherheit
- Handlungsplanung

Learning outcomes:

- Grundlegende Konzepte und Problemstellungen der KI definieren und erklären können
- Einfache Ki-Algorithmen auf konkrete – auch neue – Problemstellungen anwenden können
- Problemstellungen formal, insbesondere mit Mitteln der Logik modellieren können
- Grundzüge von KI-Programmiertechniken (insbesondere funktionale und logische Programmierung) beherrschen

Remark:

Die Vorlesung wird auf deutsch gehalten, die Folien sowie weitere Materialien sind überwiegend in englischer Sprache.

prerequisites for the module:

none

Recommended prior knowledge:	**Admission requirements:**
Fortgeschrittene Programmierkenntnisse (etwa durch Module DSG-EiAPS-B, DSG-JaP-B, GdI-IFP erworben) sowie Kenntnisse von Basisalgorithmen (etwa durch das Modul AI-AuD-B, vormals MI-AuD-B) werden vorausgesetzt, ebenso die Bereitschaft, sich in neue Programmiersprachen und -paradigmen einzuarbeiten. Grundlegende Kenntnisse in Mathematik (insbesondere formale Notation und Beweisführung, z.B. erworben in GdI-MfI-1) sowie theoretischer Informatik (z.B. erworben in GdI-GTI-B) werden empfohlen.	none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Einführung in Künstliche Intelligenz **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Ute Schmid, Prof. Dr. Diedrich Wolter **Language:** German/English **Frequency:** every summer semester **Learning outcome:** siehe Modulbeschreibung **Contents:** Präsentation und Diskussion der Inhalte (siehe Modulbeschreibung), insbesondere theoretische und konzeptionelle Aspekte. **Literature:** Stuart Russel und Peter Norvig (2010, 3. Auflage). Artificial Intelligence, a modern approach. Prentice Hall	**2,00 Weekly Contact Hours**
2. Einführung Künstliche Intelligenz **Mode of Delivery:** Practicals **Language:** German/English **Frequency:** every summer semester **Learning outcome:** siehe Modulbeschreibung **Contents:** praktische Vertiefungen zu den Inhalten der Vorlesung (siehe Modulbeschreibung)	**2,00 Weekly Contact Hours**

Examination / Duration of Examination: 105 minutes **Description:** Schriftliche Prüfung zu Inhalten der Vorlesung und Übung im Umfang von 90 Minuten. Zugelassene Hilfsmittel werden in der ersten Lehrveranstaltung bekanntgegeben. Die Prüfungsdauer beinhaltet eine Lesezeit von 15 Minuten, um die zu bearbeitenden Aufgaben im Rahmen der Wahlmöglichkeiten auswählen zu können. Im Semester werden studienbegleitend Teilleistungen in der Übung ausgegeben und besprochen, deren Abgabe freiwillig ist. Ist die Klausur bestanden, so werden die bei der Bearbeitung der Teilleistungen erreichten Punkte als Bonuspunkte angerechnet. Eine 1,0 ist dabei auch ohne Punkte aus der Bearbeitung der Teilleistungen erreichbar. Die Anzahl der erreichbaren Bonuspunkte wird in der ersten Lehrveranstaltung bekanntgegeben.	

Module DSG-DSAM-M Distributed Systems Architectures and Middleware *Distributed Systems Architecture and Middleware*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium

(since WS19/20)
Person responsible for module: Prof. Dr. Guido Wirtz

Contents:

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:

- Characteristics and Foundations of Distributed Systems
- Classical Middleware and Services
- Concurrency and Synchronization
- Component Technologies
- Cloud Computing, in particular platform as a service
- Business-to-Business Technologies

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.

Learning outcomes:

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.

Remark:

The main language of instruction in this course is English.

prerequisites for the module:

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

Recommended prior knowledge: Basic knowledge in software engineering and in distributed systems as introduced, e.g., in the module DSG-IDistrSys.		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Distributed Systems Architecture and Middleware **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Guido Wirtz **Language:** English **Frequency:** every winter semester	**2,00 Weekly Contact Hours**

Learning outcome: c.f. overall module description **Contents:** c.f. overall module description **Literature:** 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.	
2. Practicals Distributed Systems Architecture and Middleware **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Praktische Informatik **Language:** English/German **Frequency:** every winter semester **Learning outcome:** c.f. overall module description **Contents:** 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. **Literature:** c.f. overall module description	**2,00 Weekly Contact Hours**
Examination Coursework Assignment and Colloquium / Duration of Examination: 15 minutes Duration of Coursework: 3 months **Description:** Oral examination concerning the topics discussed in the lecture, exercises and assignment. **Students may choose English or German as the language for the oral examination.** 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). 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. **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.	

Module DSG-DistrSys-M Distributed Systems

Distributed Systems

6 ECTS / 180 h
45 h Präsenzzeit
135 h Selbststudium

(since SS20)
Person responsible for module: Prof. Dr. Guido Wirtz

Contents:

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 learn at least about the characteristics and basic issues of such systems. The course introduces to the different flavors of and issues with distributed systems, discusses the most basic problems arising with this kind of systems and presents solutions and techniques that are essential to make distributed systems work. Additionally, the course also teaches how to build simple distributed systems using Java-based technologies like process interaction, synchronization, remote message invocation and web service infrastructure. Students are required to work (in groups) on assignments in order to combine the theoretical concepts with practical experience and ... Yes, we program!

Learning outcomes:

Students know about the characteristics and different flavors of distributed systems and understand the essential differences compared to monolithic, centralized systems as well as their consequences when designing and building distributed systems. Students are able to apply the basic algorithmic techniques and programming paradigms in order to build simple distributed systems themselves. Students have gained basic experience with practically building and running distributed systems.

Remark:

The 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 assignment: 75h
- Literature study 30h
- preparation for and time of the final exam: 30h

This course is intended for 2nd/3rd year bachelor students as well as 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.

prerequisites for the module:

none

Recommended prior knowledge:

Knowledge of the basics of computer science in general, esp. operating systems, as well as practical experience in Java programming, as the subjects taught in DSG-EiAPS-B and DSG-EiRBS-B. Preferable also knowledge about multithreading and synchronization like, e.g., the subject-matters of DSG-PKS-B.

Module Introduction to Parallel and Distributed Programming (DSG-PKS-B) - recommended

Admission requirements:

none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lecture Distributed Systems **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Guido Wirtz **Language:** English/German **Frequency:** every summer semester **Learning outcome:** c.f. module description **Contents:** c.f. module description **Literature:** • George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair: Distributed Systems - Concepts and Design. Pearson Education UK, 2011 (5. edition); • Andrew Tanenbaum, Marten van Steen: Distributed Systems - Principles and Paradigms, 2017 (3rd edition) • Additional research literature will be provided during the term for selected readings and discussions	**2,00 Weekly Contact Hours**
2. Tutorial Distributed Systems **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Praktische Informatik **Language:** German **Frequency:** every summer semester **Learning outcome:** c.f. module description **Contents:** 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.	**2,00 Weekly Contact Hours**

Examination Coursework Assignment and Colloquium / Duration of Examination: 15 minutes Duration of Coursework: 3 months **Description:** Oral examination concerning the topics discussed in the lecture, exercises and assignment. **Students may choose English or German as the language for the oral examination.** 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.	

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.	

Module DSG-IDistrSys-B Introduction to Distributed Systems

Introduction to Distributed Systems

6 ECTS / 180 h
45 h Präsenzzeit
135 h Selbststudium

(since SS20)
Person responsible for module: Prof. Dr. Guido Wirtz

Contents:

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 learn at least about the characteristics and basic issues of such systems. The course introduces to the different flavors of and issues with distributed systems, discusses the most basic problems arising with this kind of systems and presents solutions and techniques that are essential to make distributed systems work. Additionally, the course also teaches how to build simple distributed systems using Java-based technologies like process interaction, synchronization, remote message invocation and web service infrastructure. Students are required to work (in groups) on assignments in order to combine the theoretical concepts with practical experience and ... Yes, we program!

Learning outcomes:

Students know about the characteristics and different flavors of distributed systems and understand the essential differences compared to monolithic, centralized systems as well as their consequences when designing and building distributed systems. Students are able to apply the basic algorithmic techniques and programming paradigms in order to build simple distributed systems themselves. Students have gained basic experience with practically building and running distributed systems.

Remark:

The 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 assignment: 75h
- Literature study 30h
- preparation for and time of the final exam: 30h

This course is intended for 2nd/3rd year bachelor students as well as 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.

prerequisites for the module:

none

Recommended prior knowledge:

Knowledge of the basics of computer science in general, esp. operating systems, as well as practical experience in Java programming, as the subjects taught in DSG-EiAPS-B and DSG-EiRBS-B. Preferable also knowledge about multithreading and synchronization like, e.g., the subject-matters of DSG-PKS-B.

Module Introduction to Parallel and Distributed Programming (DSG-PKS-B) - recommended

Admission requirements:

none

Frequency: every summer semester	**Recommended semester:** from 4.	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Introduction to Distributed Systems **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Guido Wirtz **Language:** English/German **Frequency:** every summer semester **Learning outcome:** c.f. overall module description **Contents:** c.f. overall module description **Literature:** • George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair: Distributed Systems. Pearson Education UK, 2011 (5. Auflage); ISBN: 9780273760597 • Kenneth P. Birman: Guide to Reliable Distributed Systems. Springer Texts in CS, Springer Verlag, 2012, ISBN 978-1-4471-2415-3	**2,00 Weekly Contact Hours**
2. Tutorials Introduction to Distributed Systems **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Praktische Informatik **Language:** English/German **Frequency:** every summer semester **Learning outcome:** c.f. overall module description **Contents:** 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. **Literature:** c.f. overall module description	**2,00 Weekly Contact Hours**

Examination	
Coursework Assignment and Colloquium / Duration of Examination: 10 minutes Duration of Coursework: 3 months **Description:** Oral examination concerning the topics discussed in the lecture, exercises and assignment. **Students may choose English or German as the language for the oral examination.** 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. **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.	

Module DSG-Project-2-SoSySc-B DSG Bachelorproject Software Systems Science

DSG Bachelorprojekt Software Systems Science

12 ECTS / 360 h
220 h Präsenzzeit
140 h Selbststudium

(since WS18/19)
Person responsible for module: Prof. Dr. Guido Wirtz

Contents:

Überschaubare Themen aus der aktuellen Forschungsarbeit der Arbeitsgruppe Verteilte Systeme (DSG), die aber eine umfangreiche Einarbeitung erfordern können, werden in einer zum Teil gemeinsam, zum Teil arbeitsteilig, arbeitenden Gruppe von Studierenden von der Konzeption bis zur praktischen Umsetzung im Rahmen des zweisemestrigen Projekts durchgeführt. Dabei geht es nicht nur um die programmiertechnische Umsetzung, sondern insbesondere auch um die Entwicklung tragfähiger und mit den vorgegebenen Rahmenbedingungen kompatibler Konzepte zur Lösung der gestellten Aufgabe, sowie um die Sicherstellung der robusten und verlässlichen Funktion der entwickelten Systeme. In der Regel wird dazu das Studium aktueller Literatur und die Auswahl, Umsetzung und/oder Adaption zum Thema vorgeschlagener Ansätze notwendig sein. Typische Themen - die sich jeweils den aktuellen Arbeiten der DSG anpassen - sind z.B. die Untersuchung von BPMN- oder BPEL basierten Standards und Ansätzen im Bereich von dienst-orientierten Systemen oder aber die Erstellung eines Prototyps zum Monitoring oder der Visualisierung verteilter Software-Systeme.

Learning outcomes:

Studierende sollen ein vertieftes Verständnis der bei der Durchführung von praktischen, arbeitsteilig organisierten, Softwareprojekten auftretenden Probleme wie auch von erfolgversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbreich der Verteilten Systeme geschieht, gewinnen die TeilnehmerInnen wichtige Erfahrungen mit der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in einem wissenschaftlich ausgerichteten Arbeitsbericht sowie einer Posterpräsentation.

Remark:

Dieses Modul erstreckt sich über 2 Semester (Start im Wintersemester):
2x6=12 ECTS, 2x4=8 SWS.

Der Arbeitsaufwand beträgt insgesamt 360 Std., welche sich grob wie folgt gliedern:

- 60 Std. Recherche, Planung und Teilnahme am Planungsworkshop
- 40 Std. Teilnahme an Projekttreffen, einschließlich Tutorien
- 180 Std. Durchführung des Projekts (Projektarbeit)
- 20 Std. Erstellung des Zwischenberichts (Hausarbeit)
- 60 Std. Erstellung des Abschlussberichts, sowie Erstellung und Präsentation des Projektposters (Hausarbeit und Kolloquium)

prerequisites for the module:

none

Recommended prior knowledge:

Grundlegende methodische Kenntnisse zur Planung und Durchführung von
Softwareprojekten und zum wissenschaftlichen Arbeiten, sowie Grundkenntnisse in der Programming paralleler und verteilter Systeme

Admission requirements:

none

wie sie z.B. durch DSG-PKS-B und/oder DSG-IDistrSys vermittelt werden.		
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 2 Semester

Module Units	
DSG Bachelorprojekt Software Systems Science **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Guido Wirtz, Scientific Staff Praktische Informatik **Language:** German/English **Frequency:** every semester **Learning outcome:** siehe Modulbeschreibung **Contents:** vgl. Modulbeschreibung **Literature:** Je nach Projektthematik; wird zu Beginn des Projekts bekannt gegeben.	**8,00 Weekly Contact Hours**

Examination Coursework Assignment / Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Anfertigen eines schriftlichen Zwischenberichts zum Projekt in Deutsch oder Englisch nach etwa 80 Std. geleisteter Projektarbeit, spätestens am Ende des Semesters, in dem das Projekt begonnen wurde. Die Gewichtung der Prüfungsleistungen wird zu Beginn des Semesters bekannt gegeben.	

Examination Coursework Assignment and Colloquium / Duration of Examination: 10 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** **Kolloquium:** Fachliche Diskussion auf der Grundlage des im Projekt bearbeiteten Themas im Rahmen einer Abschlussveranstaltung, auf der zunächst das zum Projekt angefertigte Poster erläutert wird; hier können auch praktische Projektergebnisse (z. B. lauffähige Software) demonstriert werden. **Hausarbeit:** Anfertigen eines schriftlichen Abschlussberichts in Deutsch oder Englisch nach abgeschlossener Projektarbeit. Die Gewichtung der Prüfungsleistungen wird zu Beginn des Semesters bekannt gegeben.	

Module DSG-Project-B Bachelor Project in Distributed Systems *Bachelorprojekt zur Praktischen Informatik*	6 ECTS / 180 h 130 h Präsenzzeit 50 h Selbststudium

(since WS18/19)
Person responsible for module: Prof. Dr. Guido Wirtz

Contents:

Überschaubare Themen aus der aktuellen Forschungsarbeit der Arbeitsgruppe Verteilte Systeme (DSG), die ohne umfangreiche Einarbeitung zu bearbeiten sind, werden in einer zum Teil gemeinsam, zum Teil arbeitsteilig, arbeitenden Gruppe von Studierenden von der Konzeption bis zur praktischen Umsetzung im Rahmen eines 6-wöchigen Projekts durchgeführt. Dabei geht es nicht nur um die programmiertechnische Umsetzung, sondern insbesondere auch um die Entwicklung tragfähiger und mit den vorgegebenen Rahmenbedingungen kompatibler Konzepte zur Lösung der gestellten Aufgabe. In der Regel wird dazu das Studium aktueller Literatur und die Auswahl, Umsetzung und/oder Adaption zum Thema vorgeschlagener Ansätze notwendig sein. Typische Themen - die sich jeweils den aktuellen Arbeiten der DSG anpassen - sind z.B. Transformationen zwischen verschiedenen Prozesssprachen oder XML-Darstellungen, die Erstellung einfacher, neuer Werkzeuge im Kontext der Beschreibung und Analyse verteilter Systeme oder aber die Erweiterung von Werkzeugen um neue Funktionalitäten inklusive Einarbeitung in schon vorhandene Programmsysteme usw. Dabei wird sowohl durch die konkrete Themenstellung als auch die enge betreuung und Unterstützung des Projekts darauf geachtet, dass die gestellten Aufgaben auch im (fortgeschrittenen) Bachelorstudium sinnvoll zu bearbeiten sind.

Learning outcomes:

Studierende sollen ein vertieftes Verständnis der bei der Durchführung von praktischen, arbeitsteilig organisierten, Softwareprojekten auftretenden Probleme wie auch von erfolgversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbreich der praktischen Informatik geschieht, gewinnen die TeilnehmerInnen wichtige Erfahrungen mit der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in einem wissenschaftlich ausgerichteten Arbeitsbericht.

Remark:

Der Arbeitsaufwand von insgesamt 180 Std. gliedert sich in etwa in:

- 50 Std. Einführung, Vorstellen von Werkzeugen, Vorträge zum Projektstand
- 30 Std. Recherchen zu und Einarbeitung in Thematik des Praktikums inkl. Vorbereitung von Kurzvorträgen
- 80 Std. praktische Projektarbeit (Softwareentwicklung)
- 10 Std. Abfassen des Projektberichts
- 10 Std. Vorbereitung auf und Zeit für das Kolloquium (unter o.g. schon erbrachten Aufwänden)

prerequisites for the module:

none

Recommended prior knowledge: Programmierkenntnisse sowie Kenntnisse in den Grundlagen des im Projekt behandelten Themengebiets. Modul Einführung in Algorithmen, Programmierung und Software (DSG-EiAPS-B) - empfohlen	**Admission requirements:** none

Modul Einführung in Rechner- und Betriebssysteme (PSI-EiRBS-B, vormals DSG-EiRBS-B) - empfohlen		
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Projektübung zur Praktischen Informatik **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Guido Wirtz, Scientific Staff Praktische Informatik **Language:** German **Frequency:** every summer semester **Contents:** vgl. Modulbeschreibung **Literature:** - je nach Projektthematik -	**4,00 Weekly Contact Hours**

Examination Coursework Assignment and Colloquium / Duration of Examination: 10 minutes Duration of Coursework: 2 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Anfertigen eines schriftlichen Berichts über das im Projekt durchgeführte Softwareprojekt. Diskussion des vorliegenden Projektberichts sowie der erstellten Artefakte vor dem Hintergrund des allgemeinen Themas der Projektarbeit.	

Module DSG-Project-M Master Project Distributed Systems
Masterproject Distributed Systems

9 ECTS / 270 h

(since SS19)
Person responsible for module: Prof. Dr. Guido Wirtz

Contents:

Within the project, a comprehensive topic in the field of system development in group work. Students work on problems that also arise in practice, with the most independent solution possible of a larger one, only conditionally solvable task in partly concretely given basic conditions. The Master-Project differs from the project work in the Bachelor's programme in the complexity of the task and the direct relation to current scientific work of the chair.

Learning outcomes:

Ability to independently develop solutions to problems on the basis of the acquired knowledge and the acquired skills from their studies as well as current scientific literature;
ability to solve complex problems within the framework of a systematic engineering development process into software and to document it professionally; ability to teamwork; scientific curiosity and the formation of a self-confident and researching Attitude towards technology.

Remark:

Compilation of a written project report, a poster on the project result and the existence of the verbal attestation of the project result.
The workload of 270 hours in total (as a block according to the respective SoSe) is roughly divided into
- 35 hrs. introduction, presentation of tools, short lectures
- 30 hours of research on and familiarization with the topic of the project including preparation of Short presentations
- 180 hours practical project work (software development)
- 15 hours of writing the project report and creating the joint poster
- 10 hours of preparation for the certificate (under the above-mentioned expenses already incurred)

prerequisites for the module:

none

Recommended prior knowledge:

The module builds on the DSG-DistrSys-M or DSG-IDistrSys-B (Introduction to) Distributed Systems module. Depending on the topic, also the previous visit to one of the events DSG-SOA-M or DSG-DSAM-M (to be announced at Theme announcement).
Students of the subject are expected to master a higher (object-oriented) programming language and the Readiness for practical work on the computer expected.
Modules Introduction to Distributed Systems (DSG-DistrSys-B)

Module Einführung in Verteilte Systeme (DSG-EiDistrSys) -

Admission requirements:

none

Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Distributed Systems Project **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Guido Wirtz, Scientific Staff Praktische Informatik **Language:** English **Frequency:** every semester **Learning outcome:** see module description **Contents:** Building on the knowledge gained in the lectures and exercises in the Distributed systems is presented in this event and the knowledge and skills acquired implemented a smaller project with scientific reference in a group After a short introduction to the technologies used and tools will be developed in a project with different working groups/packages organized form a related problem from the field of distributed and mobile systems. Usually this involves a prototype of a complex distributed software system or tool consisting of in this area. **Literature:** depends on the project topic	**6,00 Weekly Contact Hours**
Examination Coursework Assignment and Colloquium / Duration of Examination: 10 minutes Duration of Coursework: 3 months **prerequisites for module examination:** regular participation in the course and working in groups on the assignments **Description:** Report on the own contribution made to the project as clearly identified Part of the general report of the project group; collaboration in the preparation of a Poster demonstration of the project results. Oral examination discussion about the contents of the project, in particular the conceptual and practical work done by the respective student Services.	

Module DSG-SOA-M Service-Oriented Architecture and Web Services *Service-Oriented Architecture and Web Services*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(since SS20) Person responsible for module: Prof. Dr. Guido Wirtz	

Contents:

Building enterprise-scale IT systems requires sound concepts for integrating software. **Service-oriented architectures (SOAs)** 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 *independently deployable by fully automated deployment machinery*. 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.

When moving beyond company boundaries and opening up the solution space is necessary, **software ecosystems (SECOs)**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.

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.

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.

- Conceptual Foundations of SOA
- SOA Characterisitics
- Microservices
- WSDL and Basic Web Services
- REST-ful Services
- Software Ecosystems
- Container technology

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.

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

Learning outcomes:
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

Remark:
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 assignment: 75h
- Literature study 30h
- preparation for and time of final exam: 30h

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

Recommended prior knowledge: Basic knowledge in software engineering and distributed systems. Module Introduction to Distributed Systems (DSG-IDistrSys) - recommended	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Service-Oriented Architecture and Web Services **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Guido Wirtz, Scientific Staff Praktische Informatik **Language:** English **Frequency:** every summer semester **Learning outcome:** c.f. overall module description **Contents:** c.f. overall module description **Literature:**	**2,00 Weekly Contact Hours**

SOA is still a fast emerging field - most recent version of standards and up-to-date literature will be provided at the beginning of each course.	
2. Practicals Service-Oriented Architecture and Web Services **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Praktische Informatik **Language:** English/German **Frequency:** every summer semester **Learning outcome:** c.f. overall module description **Contents:** 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. **Literature:** c.f. overall module description	**2,00 Weekly Contact Hours**
Examination Coursework Assignment and Colloquium / Duration of Examination: 15 minutes Duration of Coursework: 3 months **Description:** Oral examination concerning the topics discussed in the lecture, exercises and assignment. S**tudents may choose English or German as the language for the oral examination.** 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. **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.	

Module DSG-Sem-B Bachelor Seminar in Practical Computer Science *Bachelorseminar zur Praktischen Informatik*		3 ECTS / 90 h
(since SS20) Person responsible for module: Prof. Dr. Guido Wirtz		
Contents: Verschiedene Themen aus dem Bereich der praktischen Informatik, die einen der fachlichen oder methodischen Aspekte aus den grundlegenden Informatik-Modulen DSG-EiAPS-B oder PSI-EiRBS-B anhand aktueller Literatur vertiefen und/oder ergänzen.		
Learning outcomes: Studierende sollen überschaubare aktuelle Themen der praktischen Informatik anhand eigener Literaturrecherchen unter Anleitung erarbeiten und in einer dem Thema angemessenen und für alle SeminarteilnehmerInnen verständlichen Form aufbereiten und präsentieren können.		
prerequisites for the module: none		
Recommended prior knowledge: Grundlegende Kenntnisse im jeweils im Seminar behandelten Gebiet der Praktischen Informatik, also mindestens eines der beiden Module DSG-EiAPS-B oder PSI-EiRBS-B. Module Introduction to Algorithms, Programming and Software (DSG-EiAPS-B) - recommended Module Introduction to Computer Architecture and Operating Systems (PSI-EiRBS-B) - recommended		**Admission requirements:** none
Frequency: every semester	**Recommended semester:** 2.	**Minimal Duration of the Module:** 1 Semester

Module Units	
Bachelorseminar zur Praktischen Informatik **Mode of Delivery:** Introductory seminar **Lecturers:** Prof. Dr. Guido Wirtz, Scientific Staff Praktische Informatik **Language:** German **Frequency:** every semester **Contents:** vgl. Modulbeschreibung **Literature:** - wird jeweils nach Seminarthemen vergeben -	**2,00 Weekly Contact Hours**
Examination Coursework Assignment with presentation / Duration of Examination: 20 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:**	

Begutachtung einer schriftlichen Ausarbeitung zu den wichtigsten Aspekten des erarbeiteten Themas mit formgerechter Liste der verwendeten Literatur. Teilnahme am Peer-Review der Ausarbeitungen anderer Teilnehmer*innen; Freies Halten eines Referats auf der Grundlage der von dem/der Vortragenden erstellten Folien oder elektronischen Präsentationsunterlagen inklusive Diskussion der Inhalte mit den Seminarteilnehmerinnen und Seminarteilnehmern.	

Module DSG-Sem-M Master Seminar in Distributed Systems *Masterseminar zu Verteilten Systemen*	3 ECTS / 90 h

(since SS20)
Person responsible for module: Prof. Dr. Guido Wirtz

Contents:
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-DistrSys-M, 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, 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.

Learning outcomes:
Students will learn how to read and work on research papers, how to present their essence as an outline talk to colleguages (students) and how to guide discussion sessions based on scientific talks. 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.

Remark:
The seminar will regularly be taught in English.

prerequisites for the module:
none

Recommended prior knowledge: Basic knowledge about distributed systems as offered, e.g., by the course *DSG-IDistrSys-B* oder *DSG-DistrSys-M* or similar knowledge. Dependend on the topic of the specific seminar, additional knowledge as discussed in DSG-SOA-M or DSG-DSAM-M may be helpful (ask if in doubt before enrolling in the course)	**Admission requirements:** none

Frequency: every semester	**Recommended semester:** 2.	**Minimal Duration of the Module:** 1 Semester

Module Units	
Master Seminar in Distributed Systems **Mode of Delivery:** Key competence **Lecturers:** Prof. Dr. Guido Wirtz **Language:** English/German **Frequency:** every semester **Learning outcome:** see module description **Contents:** see module description	**2,00 Weekly Contact Hours**

Literature: depends on specific topics of each seminar and will be given in the introductionary meeting	
Examination Coursework Assignment with presentation / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Review of a written elaboration on the most important aspects of the topic, including a correct list of references. Participation in peer reviewing the other participants; free holding of a a presentation based on presentation documents including discussion of the contents with the seminar participants.	

Module EESYS-ADAML-M Applied Data Analytics and Machine Learning in R

Applied Data Analytics and Machine Learning in R

6 ECTS / 180 h

(since SS21)

Person responsible for module: Prof. Dr. Thorsten Staake

Contents:

This course provides the theoretical foundation and conveys hands-on skills in the fields of data analytics and machine learning using the statistics software GNU R. It uses real-word datasets from the realm of energy efficiency and consumer behavior and conveys the subject matter through real-world examples and practical challenges.

Following a refresher in descriptive statistic, the course covers

- an introduction to the statistics software GNU R,
- the design of field experiments and the use of Information Systems to collect behavioral data,
- techniques to formulate, solve, and interpret linear and logistic regression analyses,
- techniques to formulate, solve, and interpret clustering analyses,
- setting up, training, and evaluating machine learning algorithms, including KNN, regression, and support vector machines, and
- ethical issues and data privacy regulations.

Learning outcomes:

After a successful participation in this course, participants can

- translate new business and research questions that can be answered using empirical methods into suitable experimental designs,
- plan and conduct corresponding experiments,
- choose suitable methods from the set of methods presented in class to analyze the data,
- explain their design choices, the choice of methods, and the steps of the analyses,
- apply the methods correctly and efficiently using the statics software R,
- adjust the methods if needed to solve new and specific problems based on an understanding of the necessary theories,
- interpret the outcome of such analyses and identify the strengths and limitations of the approaches, and
- reflect upon data protection, privacy and ethical issues related to powerful techniques for data acquisition and analytics.

Remark:

The lecture will be held as a self-paced, video-based online lecture.

The tutorials take place once per week as in-classroom events.

The online lecture includes instructional videos (scripted, i.e., with subtitles), reading material, exemplary data sets, and a multitude of online and offline tasks. It also includes an online discussion forum.

The online lecture is supported by three classroom lectures (in addition to the classroom tutorials):

1. Classroom lecture: The introductory event includes a course overview and motivation. Moreover, credentials to access the online resources will be announced. Date: First week of the semester.

2. Classroom lecture: This intermediate session includes a review of the concepts covered so far. It should help participants to self-assess their learning progress. Date: Announced in the first week of the semester.
3. Classroom lecture: Exam preparation and Q&A. Date: Last week of the semester.

An introduction to the statistics software GNU R will be given as in-classroom event during the tutorials at the beginning of the semester.

prerequisites for the module:
none

Recommended prior knowledge: This course requires a basic understanding of statistics (e.g., from a bachelor-level course). A statistics repetition and is part of the online material of the course and the of the first tutorials and should be complemented in self-study if necessary. Basic familiarity with a programming language.	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Data Analytics in Energy Informatics **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Thorsten Staake **Language:** German/English **Frequency:** every winter semester **Contents:** The video-based online lecture is divided into two parts. Part 1 conveys the statistical basics required for the module, including, for example, properties of random distributions and descriptive and injunctive statistics. This part serves as refresher of bachelor-level statistics and thereby enables students with no statistics-knowledge beyond a basic introductory course to participate. Part 2 covers the methods outlined in “Module EESYS-DAE-M” subsection “Contents”. It includes both, the theory behind the concepts and their application using R. Both, Part 1 and Part 2 use datasets and examples from industry and research and provides many hands-on examples. In order to deepen the understanding and to ease the transfer of the methods to new problems and settings, mini-tasks and small exercises are part of the online lecture. **Literature:** Reading material will be announced in class.	**2,00 Weekly Contact Hours**
2. Practicals Data Analytics in Energy Informatics **Mode of Delivery:** Practicals **Language:** German/English **Frequency:** every winter semester **Contents:** In the classroom tutorial, participants apply the methods, tools, and theories conveyed in the lecture to exemplary problems and to new challenges. This includes solving smaller tasks (e.g., acing case studies, working on concrete	**2,00 Weekly Contact Hours**

data problems) on paper and using the statistics software GNU R. Tasks are addressed individually or in small teams.

The tutorials can also cover new content, especially when its immediate application supports the learning process. Selected tutorials contain a self-assessment of the learning progress.

An introduction to GNU R is given in the first sessions.

Examination
Written examination / Duration of Examination: 90 minutes
Description:
The examination covers subject matter taught in the lectures and tutorials. The examination can also cover transfers of the subject matter to new problems and settings. Students can achieve up to 90 points.

Through the voluntary completion of coursework (“bonus exercises”) during the semester, participants can collect up to 12 additional points that are counted towards the exam, given that the exam is passed also without points from bonus exercises. Bonus exercises can take the form of written assignments, presentations, or smaller software projects. Points from bonus exercises are only valid in the semester they have been earned in and in the immediately following semester. In the first week of the course, the publishing dates of bonus exercise tasks, the submission deadlines, and the points per bonus exercise will be announced. It is possible to pass the exam with a grade of 1.0 also without points from bonus exercises.

Exam questions are stated in English, answers can be given in German or English.

Module EESYS-BIA-M Business Intelligence & Analytics

Business Intelligence & Analytics

6 ECTS / 180 h

(since SS21)
Person responsible for module: Dr. Konstantin Hopf

Contents:

This module covers topics from the field of Business Intelligence, Data Science, and Business Analytics and introduces data-driven decision support. The main topics include

- the role of Business Intelligence in organizations,
- the data analytics process using the CRISP-DM process model,
- data sources in organizations and publicly available data sources,
- an introduction to Data Science and the basics of data analytics including a repetition of the fundamentals of descriptive statistics and data visualization,
- fundamentals of spatio-temporal data analysis,
- advanced analytics methods including unsupervised and supervised machine learning, optimization and simulation, and
- legal and ethical aspects of data analytics (in particular privacy, data security and copyright).

Students approach the topics by means of concrete data analytics examples and case studies in the programming environment GNU R. The course covers the most important steps of the data analytics process (business understanding, data understanding, data preparation, modeling, evaluation and deployment).

Learning outcomes:

Students will be able to

- describe the role of business intelligence and data analyses in organizations,
- identify available internal and publicly available data sources, make them usable and describe the data,
- apply advanced analytics methods (especially: k nearest neighbor, Decision Trees, Support Vector Machines, Random Forest) in the software R on their own and create analyses for business-relevant questions, which can be used as a basis for decision-making,
- visualize the results of the analyses in a meaningful way, and
- describe selected ethical and legal aspects of data analytics.

prerequisites for the module:

none

Recommended prior knowledge:

This course requires a basic understanding of statistics (e.g., from a bachelor-level course). A statistics repetition is part of the course and should be complemented in self-study if necessary.

Basic familiarity with a programming language.

Admission requirements:

none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Lectures Business Intelligence & Analytics

2,00 Weekly Contact Hours

Mode of Delivery: Lectures
Lecturers: Prof. Dr. Thorsten Staake
Language: German/English
Frequency: every winter semester

Contents:
The lecture covers the topics mentioned in “Module EESYS-BIA-M”, subsection “Contents”. Traditional lecture elements, case studies, discussions, exercises, and group work are used to support participants in reaching the learning objectives. Methods, tools, and theories are introduced with references to practical challenges and applied to exemplary problems. For selected topics, the lectures rely on flipped classroom elements for which participants need to acquire knowledge in advance (e.g., through reading tasks), which is then critically reflected and extended in the classroom lecture.

The course materials are in English, the language of instruction is agreed upon in the first course together with the course participants.

Literature:
Reading material will be announced in class.

2. Practicals Business Intelligence & Analytics

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Language: German/English
Frequency: every winter semester

Contents:
In the classroom tutorial, participants apply the methods, tools, and theories conveyed in the lecture to exemplary problems and to new challenges. This includes solving smaller tasks (e.g., acing case studies, working on concrete data problems) on paper and using the statistics software GNU R. Tasks are addressed in small teams or individually.

The tutorials can also cover new content, especially when its immediate application supports the learning process. Selected tutorials contain a self-assessment of the learning progress.

An introduction to GNU R is given in the first sessions.

Examination
Written examination / Duration of Examination: 90 minutes
Description:
The examination covers subject matter taught in the lectures and tutorials. The examination can also cover transfers of the subject matter to new problems and settings. Students can achieve up to 90 points.

Through the voluntary completion of coursework (“bonus exercises”) during the semester, participants can collect up to 12 additional points that are counted towards the exam, given that the exam is passed also without points from bonus exercises. Bonus exercises can take the form of written assignments, presentations, or smaller software projects. Points from bonus exercises are

only valid in the semester they have been earned in and in the immediately following semester. In the first week of the course, the publishing dates of bonus exercise tasks, the submission deadlines, and the points per bonus exercise will be announced. It is possible to pass the exam with a grade of 1.0 also without points from bonus exercises. Exam questions are stated in English, answers can be given in German or English.	

Module EESYS-ES-M Energy Efficient Systems

Energieeffiziente Systeme

6 ECTS / 180 h

(since WS19/20)
Person responsible for module: Prof. Dr. Thorsten Staake

Contents:

The course covers the design and application of Information Systems that help increase energy efficiency and reduce greenhouse gas emissions. It is directed to computer science and Information Systems students that want to apply their skills to challenges in the fields of energy, mobility, production, and sustainable consumption/consumer behavior.

The course introduces methods and theories from behavioral economics, operations management, and simulation analysis that help to understand, analyze, and shape both, industry processes and consumer behavior in the field of sustainability. Also covered are cost/benefit considerations on a micro- and macro-level (including, for example, rebound effects) and a discussion on the economic and societal implications of the subject matter.

The course includes an introduction to physics and energy engineering to allow students with very limited knowledge in these fields to participate successfully.

Learning outcomes:

Successful participants of this course shall acquire the skills to

- explain the physical and technical principals covered in this course and apply them to new problems,
- explain the components, influencing factors, requirements and challenges related to electric mobility and describe the contribution that Information Systems can make to solve the challenges; moreover, successful participants shall be able to set up data-based simulations to derive important characteristic variables related to electric vehicles, such as electric reachability, peak loads to electric grids, etc.,
- outline, assess, and conceptually model the potential of Information Systems and the effects to heating and room climate applications,
- explain in detail the characteristics of and implications from environmental business Information Systems,
- explain the discussed behavioral theories (e.g., the prospect theory), make use of them when building Information Systems that support decision making and behavioral change, and be able to evaluate the effectiveness of such systems, and
- evaluate the effects of the tools and methods introduced, including their micro- and macro-economic effects, and critically assess the techniques used to perform such evaluations.

Moreover, successful participants shall be able to apply the acquired skills to new challenges and adjust and extend them as needed.

Finally, the participants shall realize the scope for design and the potential that results from their IT studies to favorably shape a sustainable and socially desirable development of our society.

prerequisites for the module:
none

Recommended prior knowledge: none	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Energy Efficient Systems **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Thorsten Staake **Language:** German/English **Frequency:** every summer semester **Contents:** The lecture covers the topics mentioned in “Module EESYS-ES-M”, subsection “Contents”. It uses traditional lecture elements, discussions, exercises, and group work to support participants in reaching the learning objectives. Special emphasis is placed on working on cases and on discussions of studies and scientific publications. Methods, tools, and theories are introduced with references to practical challenges and are applied to exemplary problems. For selected topics, the lecture relies on flipped classroom elements for which participants need to acquire knowledge in advance (e.g., through reading tasks), which is then critically reflected and extended in the classroom sessions. **Literature:** Weiterführende Unterlagen werden in der Veranstaltung bekanntgegeben.	**2,00 Weekly Contact Hours**
2. Practicals Energy Efficient Systems **Mode of Delivery:** Practicals **Language:** German/English **Frequency:** every summer semester **Contents:** The first tutorials convey basics in physics and electrical engineering in order to also allow students who did not take related modules to participate in this course. Subsequently, participants apply the methods, tools, and theories conveyed in the lecture to exemplary problems and to new challenges. Tutorials include small tasks, case studies, and reviews of scientific publications that are addressed individually or in small teams. The tutorials can also cover new content, especially when its immediate application supports the learning process. Selected tutorials contain a self-assessment of the learning progress. **Literature:** Reading material will be announced in class.	**2,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 90 minutes **Description:** The examination covers subject matter taught in the lectures and tutorials. The examination can also cover transfers of the subject matter to new problems and settings. Students can achieve up to 90 points. Through the voluntary completion of coursework (“bonus exercises”) during the semester, participants can collect up to 12 additional points that are counted	

towards the exam, given that the exam is passed also without points from bonus exercises. Bonus exercises can take the form of written assignments, presentations, or smaller software projects. Points from bonus exercises are only valid in the semester they have been earned in and in the immediately following semester. In the first week of the course, the publishing dates of bonus exercise tasks, the submission deadlines, and the points per bonus exercise will be announced. It is possible to pass the exam with a grade of 1.0 also without points from bonus exercises. Exam questions are stated in English, answers can be given in German or English.	

Module EESYS-IITP-B International IT Project Management *Internationales IT-Projektmanagement*		6 ECTS / 180 h
(since WS19/20) Person responsible for module: Prof. Dr. Thorsten Staake		
Contents: This course provides its participants with the knowledge they need to manage and support IT projects. It covers the entire project lifecycle from scoping to planning, executing, controlling, and closing projects and discusses both, traditional (e.g., waterfall- and V-models) and agile (e.g., Scrum) management techniques. The course addresses issues that are relevant for small, agile companies in unstable environments as well as for multinationals with well-established processes. Special attention is paid to the management of international projects and international teams. Throughout the course, care is taken to combine hands-on advice about tools, techniques, and management practices with insights about the concepts' theoretical foundations, strengths, and limitations.		
Learning outcomes: After successfully completing this course, students are well-equipped with sound methods to plan, execute, and manage IT projects in small and large organizations. They are also able to support local and international teams. Moreover, based on a solid theoretical background, successful participants are able to apply the methods and tools to new settings and problems and develop them further if it is required.		
Remark: The course is organized as classroom lecture with the exception that three to four lecture units are available exclusively online. The online lectures include scripted videos, references to literature and tasks that can be completed online. The tutorial will be held as classroom event.		
prerequisites for the module: none		
Recommended prior knowledge: none		**Admission requirements:** none
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Internationales IT-Projektmanagement **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Thorsten Staake **Language:** German/English **Frequency:** every summer semester **Contents:** The lecture covers the topics mentioned in "Module EESYS-IITP-B", subsection "Contents". It uses traditional lecture elements, case studies, discussions, exercises and group work to support participants in reaching the learning	**2,00 Weekly Contact Hours**

objectives. Methods, tools, and theories are introduced with references to practical challenges and applied to exemplary problems. For selected topics, the lectures rely on flipped classroom elements for which participants need to acquire knowledge in advance (e.g., through reading tasks), which is then critically reflected and extended in the classroom lecture.

Literature:
Reading material will be announced in class.

2. Practicals Internationales IT-Projektmanagement

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Language: German/English
Frequency: every summer semester

Contents:
In the tutorial, participants apply the methods, tools, and theories conveyed in the lecture to exemplary problems and to new challenges. Tutorials include small tasks, case studies, and reviews of management guidelines and scientific publications that are addressed individually or in small teams.

The tutorials can also cover new content, especially when its immediate application supports the learning process. Selected tutorials contain a self-assessment of the learning progress.

Examination
Written examination / Duration of Examination: 90 minutes
Description:
The examination covers subject matter taught in the lectures and tutorials. The examination can also cover transfers of the subject matter to new problems and settings. Students can achieve up to 90 points.

Through the voluntary completion of coursework (“bonus exercises”) during the semester, participants can collect up to 12 additional points that are counted towards the exam, given that the exam is passed also without points from bonus exercises. Bonus exercises can take the form of written assignments, presentations, or smaller software projects. Points from bonus exercises are only valid in the semester they have been earned in and in the immediately following semester. In the first week of the course, the publishing dates of bonus exercise tasks, the submission deadlines, and the points per bonus exercise will be announced. It is possible to pass the exam with a grade of 1.0 also without points from bonus exercises.

Exam questions are stated in English, answers can be given in German or English.

Module GdI-CSNL-M Computational Semantics of Natural Language

Computational Semantics of Natural Language

6 ECTS / 180 h

(since SS21)
Person responsible for module: Prof. Ph.D. Michael Mendler
further responsible : Luke Burke

Contents:

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.

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.

This course may also be of interest to students in philosophy and linguistics.

Learning outcomes:

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.

Remark:

The workload for this module consists of:

- participation in lectures and tutorial sessions: 45hrs
- individual preparation and reading: 105hrs
- exam preparation and oral exam: 30hrs

prerequisites for the module:

none

Recommended prior knowledge:

Willingness to read relevant literature, critically discuss and analyse it and write about it. Basic logic (GdI-MfI-1: Mathematik fur Informatik or an equivalent level of understanding). Some knowledge of modal logic more basic than that required for (GdI-MTL: Modal and

Admission requirements:

English language skills at Level B2 (UniCert II) or above.

Temporal Logic). Knowledge of the typed lambda calculus (abstraction and application) and elementary Haskell (GdI-IFP: Introduction to Functional Programming) would be very useful, though not essential.		
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** Semester

Module Units	
Computational Semantics of Natural Language **Language:** English **Frequency:** every summer semester **Contents:** 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. **Literature:** • 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.	**4,00 Weekly Contact Hours**

Examination Portfolio / Duration of Examination: 45 minutes **Description:** The portfolio assessment consists of • extended abstract (1200-1600 words) • final oral exam with presentation (45 min) Each portfolio part is graded individually. The grades are weighted as follows: 20% for the extended abstract, 80% for final oral exam including presentation.	

Module GdI-FPRS-M Functional Programming of Reactive Systems *Functional Programming of Reactive Systems*	6 ECTS / 180 h

(since SS21)
Person responsible for module: Prof. Ph.D. Michael Mendler

Contents:

Based on an existing basic knowledge of functional programming (FP), the aim of this module is to develop advanced skills in the use of FP languages to structure and solve algorithmic problems in designing interactive and concurrent systems. We will study advanced programming abstractions specifically developed for the functional modelling of synchronous reactive systems. Following the methodological structure of the introductory course GDI-IFP, this advanced course, too, combines both practical programming with a focused discussion of pertinent underlying mathematical concepts. Though we use Haskell as our main language we may also look at other FP languages such as F#, ML or OCAML where appropriate.

Learning outcomes:

At the end of this course students should

- be familiar with advanced FP programming concepts and their application (e.g., class mechanism, type families, higher-rank polymorphism, monad and arrow abstractions, lenses, continuation-style programming, stream programming, concurrency abstractions)
- be able to use these advanced language concepts to solve complex algorithmic problems efficiently, in particular involving the use of memory, concurrency and interaction
- be able use the Haskell stack build tool and understand the mechanisms of package management
- appreciate the importance of functional abstraction for conciseness and efficiency of programming complex applications
- be familiar with the second-order polymorphic lambda calculus (Hindley-Milner predicative let-polymorphism, impredicative System F) as an operational semantics behind (eager, lazy) functional programming
- be able to explain the encoding of recursive data structures in type theory
- have an elementary understanding of the execution model of functional languages and transformation to operational code through defunctionalisation and abstract machines.
- by able to use FP (specifically Haskell) as a development tool for the design of new programming languages

Remark:

The workload for this module splits up roughly like this:

- participation in lectures and tutorials: 45 hrs
- preparation of classes and tutorials as well literature research: 60 hrs
- solving (ungraded) programming exercises and participation in lab sessions: 45 hrs
- exam preparation: 30 hrs

prerequisites for the module:

none

Recommended prior knowledge: Elementary programming skills in a functional programming language, such as from module GdI-IFP-B; Basic knowledge in the use of	**Admission requirements:** none

temporal and modal logic specification formalisms such as from GdI-MTL-B. English language skills at Level B2 (UniCert II) or above. Module Introduction to Functional Programming (GdI-IFP) - recommended		
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Advanced Functional Programming
Mode of Delivery: Lectures
Lecturers: Prof. Ph.D. Michael Mendler
Language: English/German
Frequency: every summer semester

2,00 Weekly Contact Hours

Contents:
Through class presentations and direct interactions with the students the lecturer introduces the topics of the course in detail, poses exercises and suggests literature for self-study.

Literature:

- S. Marlow: The Haskell 2010 Language Report. https://www.haskell.org/onlinereport/haskell2010/
- V. Zsók, Z. Horváth, R. Plasmeijer: Central European Functional Programming School. Springer 2012.
- S. Marlow: Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming, O'Reilly 2013.
- B. O'Sullivan, J. Goerzen, D. Stewart: Real World Haskell. O'Reilly 2009.
- Ch. Okasaki: Purely Functional Data Structures, CUP 1998
- F. Rabhi, G. Lapalme: Algorithms - A Functional Approach.
- D. Syme, A. Granicz, A. Cisternino: Expert F#4.0, Apress 2015.
- B. Pierce: Types and Programming Languages. MIT Press 2002. (esp. Chapters 23+25)
- H. Barendregt, W. Dekkers, R. Statman: Lambda Calculus with Types. CUP 2013.

2. Functional Programming of Reactive Systems
Mode of Delivery: Practicals
Lecturers: Prof. Ph.D. Michael Mendler
Language: English/German
Frequency: every summer semester

2,00 Weekly Contact Hours

Contents:
The tutorials deepen the students' understanding of the theoretical concepts and constructions covered in the lectures through practical exercises. Participants are given the opportunity to discuss their solutions to homework question sheets and sample solutions are presented by the tutors or lecturer for selected exercises. The tutorials also provide exam preparation.

Literature:

The literature will be announced in class. Here are some general pointers on FP languages and synchronous programming.

- S. Marlow: The Haskell 2010 Language Report. https://www.haskell.org/onlinereport/haskell2010/
- V. Zsók, Z. Horváth, R. Plasmeijer: Central European Functional Programming School. Springer 2012.
- S. Marlow: Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming, O'Reilly 2013.
- D. Syme, A. Granicz, A. Cisternino: Expert F#4.0, Apress 2015.
- H. Barendregt, W. Dekkers, R. Statman: Lambda Calculus with Types. CUP 2013.

- Benveniste, A. et al: The Synchronous Languages 12 years later. Proc. IEEE, Vol 91(1), January 2003.
- Berry, G.: SCADE: Synchronous design and validation of embedded control software. In: Next Generation Design and Verification Methodologies for Distributed Embedded Control Systems. Proc. GM R&D Workshop, Bangalore, January 2007. pp. 19-33.
- Potop-Butucaru et. al: The Synchronous Hypothesis and Synchronous Languages. In Richard Zurawski. *Embedded Systems Design and Verification*, CRC Press, pp.6-1-6-27, 2009.

Examination
Written examination / Duration of Examination: 90 minutes
Description:
The examination language is English.

The form of examination is either oral (30 minutes) or written (90 minutes) depending on the number of participants. The form of examination will be determined at the beginning of the semester and announced in class.

Examination
Oral examination / Duration of Examination: 30 minutes
Description:
The examination language is English.

The form of examination is either oral (30 minutes) or written (90 minutes) depending on the number of participants. The form of examination will be determined at the beginning of the semester and announced in class.

Module GdI-GTI-B Machines and Languages *Grundlagen der Theoretischen Informatik*	6 ECTS / 180 h

(since WS18/19)
Person responsible for module: Prof. Ph.D. Michael Mendler

Contents:
This course addresses the questions "what is a computation?" and "what is an algorithm?" and explores the capabilities and limitations of computers and programming languages as well as the implication of these for a practical computer scientist. It introduces the basic concepts and methods that underlie the mathematical study of computing machines and formal languages.

Learning outcomes:
At the end of this course the students should be able to distinguish finite automata, pushdown automata, Turing machines, and know the difference between the deterministic and non-deterministic versions in each case; be able to distinguish regular, context-free, context-sensitive and general phrase structure grammars in the Chomsky Hierarchy; understand the relations between language classes and machine classes; have developed elementary automata and Turing machine programming skills; know the basic concepts of algorithmic complexity theory such as the big-O notation and key complexity classes such as N and NP as well as their relationship.

Remark:
The language of instruction in this course is German. However, all course materials (lecture slides and tutorial notes) as well as the exam are available in English.

prerequisites for the module:
None.

Recommended prior knowledge: Elementary concepts in logic and discrete mathematics for computer scientists; Basic programming skills; English language skills at Level B2 (UniCert II) or above. Module Introduction to Algorithms, Programming and Software (DSG-EiAPS-B) - Module Propositional and Predicate Logic (GdI-MfI-1) - recommended	**Admission requirements:** None.

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Machines and Languages **Mode of Delivery:** Lectures **Lecturers:** Prof. Ph.D. Michael Mendler **Language:** German/English **Frequency:** every summer semester	**2,00 Weekly Contact Hours**
Contents: Through prepared class presentations and direct interactions with the students the lecturer introduces the topics of the course in detail, poses exercises and suggests literature for self-study.	

Literature:

- Hopcroft, J. E., Motwani, R., Ullman, J. D.: Introduction to Automata Theory, Languages, and Computation. Addison Wesley, 2001.
- Martin, J. C.: Introduction to Languages and the Theory of Computation, McGraw Hill, (2nd ed.), 1997.
- Sudkamp, Th. A.: Languages and Machines. An Introduction to the Theory of Computer Science. Addison Wesley, (2nd ed.) 1997.

2. Machines and Languages

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Lecturers: Prof. Ph.D. Michael Mendler, N.N.
Language: English/German
Frequency: every summer semester

Contents:

The tutorials deepen the students' understanding of the theoretical concepts and constructions covered in the lectures through practical exercises. Participants are given the opportunity to present their solutions to homework question sheets and sample solutions are given by the lecturer for selected exercises. The tutorials also provide exam preparation.

Examination

Written examination / Duration of Examination: 90 minutes

Description:

90 min written examination. The exam takes place during the regular exam period after the end of the semester. An alternative oral exam may be negotiable for guest students only.

Module GdI-IFP-B Introduction to Functional Programming *Introduction to Functional Programming*	6 ECTS / 180 h
(since WS19/20) Person responsible for module: Prof. Ph.D. Michael Mendler	

Contents:

The aim of this module is to provide an introduction to functional programming using Haskell. This course develops both elementary practical programming skills and discusses the typed lambda calculus and its role as an operational semantics for functional programming, stressing the importance of types and type checking for static program analysis.

Learning outcomes:

At the end of this course students should be familiar with important language constructs of Haskell and their semantics (e.g., expressions, local declarations, higher-order function abstraction, recursion, lazy and eager evaluation, referential transparency, algebraic data types, monads); be able to use these language concepts to solve algorithmic problems; be familiar with the lambda calculus as an operational semantics behind functional programming; understand the difference between imperative and declarative programming styles; have an appreciation of the close relationship between programming language types and specification and the role of type checking as a static program analysis method; be familiar with polymorphic Hindley-Milner style type systems.

Remark:

The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German.

prerequisites for the module:

none

Recommended prior knowledge: Elementary concepts in logic and discrete mathematics for computer scientists; Basic programming skills; English language skills at Level B2 (UniCert II) or above. Module Introduction to Algorithms, Programming and Software (DSG-EiAPS-B) - recommended Module Propositional and Predicate Logic (GdI-MfI-1) - recommended		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Introduction to Functional Programming **Mode of Delivery:** Lectures **Lecturers:** Prof. Ph.D. Michael Mendler **Language:** English/German **Frequency:** every winter semester **Contents:**	**2,00 Weekly Contact Hours**

Through prepared class presentations and direct interactions with the students the lecturer introduces the topics of the course in detail, poses exercises and suggests literature for self-study. **Literature:** • Pierce, B. C.: Types and Programming Languages, MIT Press, 2002 • Thompson, S.: Haskell – The Craft of Functional Programming, Addison-Wesley 1999.	
2. Introduction to Functional Programming **Mode of Delivery:** Practicals **Lecturers:** Prof. Ph.D. Michael Mendler **Language:** English/German **Frequency:** every winter semester **Contents:** The tutorials deepen the students' understanding of the theoretical concepts and constructions covered in the lectures through practical exercises. Participants are given the opportunity to discuss their solutions to homework question sheets and sample solutions are presented by the tutors or lecturer for selected exercises. The tutorials also provide exam preparation.	**2,00 Weekly Contact Hours**
Examination Written examination / Duration of Examination: 90 minutes **Description:** 90 min written examination. The exam takes place during the regular exam period after the end of the semester.	

Module GdI-MTL Modal and Temporal Logic *Modal and Temporal Logic*	6 ECTS / 180 h

(since WS21/22)
Person responsible for module: Prof. Ph.D. Michael Mendler

Contents:
This advanced module aims to give a thorough introduction to a selection of modal logics with strong applications in Computer Science. Basic knowledge of classical propositional logic and predicate logic and associated calculi is assumed as a prerequisite. Among the logics covered are modal and temporal logics for the analysis of distributed systems or semantic information processing. Depending on the time available, the module also covers belief logics and other specialised logics for security protocols and distributed algorithms. The course addresses theoretical foundations (models and proof systems) but also discusses applications and offers practical experience through hands-on experimentation with automatic and interactive verification tools.

Learning outcomes:
At the end of the course students should understand the commonalities and differences between propositional and predicate logics on the one hand and modal logics on the other for system specification and modelling; be aware of the important role played by modal logics for the trade-off between expressiveness and automation; know the semantical foundations of modal logics based on Kripke structures; understand the difference between epistemic, temporal, deontic modalities; be familiar with basic results from modal correpondence theory with modal theories such as K, S4, S5; know the Hennessy-Milner Theorem, model filtration and minimzation techniques; apply standard reasoning procedures based on Hilbert, Gentzen Sequent and Tableau calculi; be familiar with the syntax and semantics of important temporal logics such as PLTL, CTL and description logics such as ALC; be able to apply deduction and model-checking techniques for the specification and verification of distributed and dynamic systems as well as semantic information processing.

Remark:
The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German.

prerequisites for the module:
none

Recommended prior knowledge: Elemantary logic and discrete mathematics for computer scientists; Basic programming skills. Module Introduction to Algorithms, Programming and Software (DSG-EiAPS-B) - recommended Module Propositional and Predicate Logic (GdI-MfI-1) - recommended	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Modal and Temporal Logic **Mode of Delivery:** Lectures and Practicals **Lecturers:** Prof. Ph.D. Michael Mendler	**4,00 Weekly Contact Hours**

Language: English/German
Frequency: every winter semester

Contents:

Through prepared class presentations and direct interactions with the students the lecturer introduces the topics of the course in detail, poses exercises and suggests literature for self-study.

Literature:

- Fagin, R., Halpern, J. Y., Moses, Y., Vardi, M. Y.: Reasoning about Knowledge. MIT Press, (2nd printing) 1996.
- Hughes, G. E., Cresswell, M. J.: A New Introduction to Modal Logic. Routledge, (3rd reprint) 2003.
- Popkorn, S.: First Steps in Modal Logic. Cambridge University Press, 1994.
- Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds): The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, (2nd ed.) 2007.

Examination
Written examination / Duration of Examination: 90 minutes
Description:
The examination language is English.

The form of examination is either oral (30 minutes) or written (90 minutes) depending on the number of participants. The form of examination will be determined at the beginning of the semester and announced in class.

Examination
Oral examination
Description:
The examination language is English.

The form of examination is either oral (30 minutes) or written (90 minutes) depending on the number of participants. The form of examination will be determined at the beginning of the semester and announced in class.

Module GdI-MfI-1 Propositional and Predicate Logic *Mathematik für Informatik 1 (Aussagen- u. Prädikatenlogik)*	6 ECTS / 180 h

(since SS17)
Person responsible for module: Prof. Ph.D. Michael Mendler

Contents:

This module gives an introduction to basic constructions in the proof theory and model theory of propositional and typed predicate logic. It provides the necessary logical foundations for many computer science courses which use mathematical formalisations and deductive methods and/or directly use symbolic reasoning in applications.

Learning outcomes:

At the end of this course students should be able to perform elementary calculations in algebraic structures such as Boolean, functional and relational algebras; be familiar with the concept of a formal system and formal calculus and have understood the fundamental difference between syntax and semantics, soundness and completeness; be able to formalize real-world concepts in propositional and predicate logic and have developed skills in reasoning using formal calculi for these logics; be able to apply elementary proof principles (proof by contraposition, proof by cases, natural and structural induction); be familar with the concept of types and logical signatures for static specification and language formalisation.

Remark:

The main language of instruction in this course is German. However, all course materials (lecture slides and tutorial notes) as well as the exam are available in English.

prerequisites for the module:

none

Recommended prior knowledge: English language skills at Level B2 (UniCert II) or above.		**Admission requirements:** none.
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Mathematik für Informatik 1
Mode of Delivery: Lectures
Lecturers: Prof. Ph.D. Michael Mendler
Language: German
Frequency: every winter semester

2,00 Weekly Contact Hours

Contents:

In der Vorlesung wird das Themengebiet der Veranstaltung durch Dozentenvortrag eingeführt und Anregungen zum weiterführenden Literaturstudium gegeben.

Literature:

- Ehrig, H., Mahr, B., Cornelius, F., Große-Rhode, Zeitz, M. P.: Mathematisch strukturelle Grundlagen der Informatik. Springer Verlag, 2. Aufl., 2001.
- Grassmann, W. K., Tremblay, J.-P.: Logic and Discrete Mathematics - A Computer Science Perspective. Prentice Hall, 1996.

- Scheinerman, E. R.: Mathematics – A Discrete Introduction. Brooks/Cole, 2000. - Barwise, J., Etchemendy, J: Language, Proof, and Logic. Seven Bridges Press, 2000.	
2. Mathematik für Informatik 1 **Mode of Delivery:** Practicals **Lecturers:** Prof. Ph.D. Michael Mendler, N.N. **Language:** German **Frequency:** every winter semester **Contents:** Die Übung vertieft die Konzepte und Konstruktionen aus der Vorlesung an konkreten Beispielen. Sie dient damit auch der Klausurvorbereitung.	**2,00 Weekly Contact Hours**
Examination Written examination / Duration of Examination: 90 minutes	

Module HCI-DISTP-B Design of Interactive Systems: Theory and Practice *Design Interaktiver Systeme: Theorie und Praxis*	3 ECTS / 90 h

(since WS17/18)
Person responsible for module: Prof. Dr. Tom Gross

Contents:

Theoretical, methodical, practical foundation of design and practical design with focus on a research challenge.

Learning outcomes:

The aim of this module is a general introduction to basic practical skills, processes, and methods of design with a special application-oriented focus on the user-centred design of complex interactive systems.

Remark:

http://www.uni-bamberg.de/hci/leistungen/studium

The workload for this module is roughly structured as following:

- Attendance of the lecture units
- Participation in the group meetings
- Work on the tasks alone and with the team
- Preparation of discussions and presentation
- Exam preparation

The workload for each participant may vary over the different tasks based on the task definitions and the joint coordination of tasks in the team

The default language of instruction is German and can be changed to English based on students' needs. All course materials (incl. exams) are available in English

prerequisites for the module:

none

Recommended prior knowledge: none		**Admission requirements:**
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Design of Interactive Systems: Theory and Practice **Mode of Delivery:** Lectures and Practicals **Lecturers:** Jochen Denzinger **Language:** German/English **Frequency:** every summer semester	**1,00 Weekly Contact Hours**

Contents:

In this lecture the following topics are covered:

- Design theory and history
- Design of multimodal user interfaces
- User-Centred Design, User-Experience Design
- Practical design, incl. practical application of methods for the iterative design

The assignements cover diverse topics based on the contents of the course. The practical part includes an iterative design as an assignment. The task is significantly more comprehensive than the normal assignments accompanying the lectures and therefore is solved in a small group. The results are documented and demonstrated in a final presentation. **Literature:** The course is based on a compilation of different sources; as additional sources and as a reference are recommended: • Krippendorff, K. The Semantic Turn. A New Foundation for Design. Taylor & Francis Group, Boca Raton, FL, 2006. • Moggridge, B. Designing Interactions. MIT Press, Cambridge, MA, 2007.	
Examination Colloquium / Duration of Examination: 30 minutes **Description:** Colloquium on the assignment process and results	

Module HCI-KS-B Cooperative Systems *Kooperative Systeme*		6 ECTS / 180 h
(since WS21/22) Person responsible for module: Prof. Dr. Tom Gross		
Contents: Theoretical, methodological, and practical foundation of Computer-Supported Cooperative Work.		
Learning outcomes: The aim of this module is to teach advanced paradigms and concepts of computer-supported cooperative work (CSCW) and the resulting design principles and prototypes. Hereby a broad perspective on the topic is applied; accordingly a central concern is the general technological support of social interaction, spanning cooperative work and learning as well as leisure activities.		
Remark: http://www.uni-bamberg.de/hci/leistungen/studium The workload for this module is roughly structured as following: • Attendance of the lectures and assignments: 45 hours • Credits of the lecture (incl. research and study of additional sources): ca. 30 hours • Credits of the assignments (incl. research and study of additional sources, but without optional homework assignment): ca. 30 hours • Solving the optional homework assignments: overall ca. 45 hours • Exam preparation: ca. 30 hours (based on the above mentioned preparation and revision of the subject material) The default language of instruction is German and can be changed to English based on students' needs. All course materials (incl. exams) are available in English		
prerequisites for the module: none		
Recommended prior knowledge: Basic knowledge in computer science to the extent of an introduction to algorithms, programming and software, as well as programming skills in Java.		**Admission requirements:** Passing the written exam
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Cooperative Systems **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Tom Gross **Language:** German/English **Frequency:** every summer semester **Contents:** After an introduction into the subject the following topics are covered in this lecture: • Basic concepts	**2,00 Weekly Contact Hours**

- Technological support for mutual awareness, communication, coordination, collaboration, and online communities
- Analysis of cooperative environments
- Design of CSCW and groupware systems
- Implementation of CSCW and groupware systems
- CSCW in a broader context and related topics

Literature:

The course is based on a compilation of different sources; as additional sources and as a reference are recommended:

- Gross, T. and Koch, M. Computer-Supported Cooperative Work (Computer-Supported Cooperative Work; in German). Oldenbourg, Munich, 2007.
- Borghoff, U.M. and Schlichter, J.H. Computer-Supported Cooperative Work: Introduction to Distributed Applications. Springer-Verlag, Heidelberg, 2000.

Examination

Oral examination

Description:

The oral exam takes 30 minutes and is worth a total of 90 points. Depending on the number of attendees the form of the exam can be changed to a written exam with 90 minutes and a total of 90 points. The final form of the exam is announced in the first lecture at the beginning of the term.

During the semester students can do assignments, which are optional. They are 12 points in total. The type of optional homework assignments as well as the deadlines are announced in detail at the beginning of the term. If the oral exam is passed (as a rule 50% of the points have to be reached) the points from the assignments are a bonus and added to the points from the oral exam. In any case, a top grade of 1,0 is also reachable without solving the assignments.

Module Units

Cooperative Systems

2,00 Weekly Contact Hours

Mode of Delivery: Practicals

Lecturers: Scientific Staff Mensch-Computer-Interaktion

Language: German/English

Frequency: every summer semester

Contents:

Practical assignments based on the subjects of the lecture including the programming of small prototypes

Literature:

Cf. lecture

Examination

Written examination / Duration of Examination: 90 minutes

Description:

In Abhängigkeit 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 sowie deren Bearbeitungsfrist werden zu Beginn der Lehrveranstaltung verbindlich bekannt gegeben. Ist die 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 Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.	

Module HCI-MCI-M Human-Computer Interaction *Mensch-Computer-Interaktion*	6 ECTS / 180 h

(since WS21/22)
Person responsible for module: Prof. Dr. Tom Gross

Contents:
Advanced theoretical, methodological, and practical foundation of Human-Computer Interaction

Learning outcomes:
The aim of this module is to teach advanced knowledge and skills in the area of human-computer interaction as well as a broad theoretical and practical methodological expertise concerned with the design, conception, and evaluation of ubiquitous systems. Students of this course learn the relevant literature and systems in breadth and depth and are later able to critical review new literature and systems.

Remark:
http://www.uni-bamberg.de/hci/leistungen/studium

The workload for this module is roughly structured as following:

- Attendance of the lectures and assignments: 45 hours
- Credits of the lecture (incl. research and study of additional sources): ca. 30 hours
- Credits of the assignments (incl. research and study of additional sources, but without optional homework assignment): ca. 30 hours
- Solving the optional homework assignments: overall ca. 45 hours
- Exam preparation: ca. 30 hours (based on the above mentioned preparation and revision of the subject material)

The default language of instruction in this course is German, but can be changed to English on demand. All course materials (incl. exams) are available in English.

prerequisites for the module:
none

Recommended prior knowledge: Module Algorithms and data structures (MI-AuD-B) Module Introduction to Algorithms, Programming and Software (DSG-EiAPS-B)	**Admission requirements:** Passing the written exam

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Human - Computer Interaction **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Tom Gross **Language:** German/English **Frequency:** every summer semester	**2,00 Weekly Contact Hours**

Contents:
After an introduction into the subject the following topics are covered in this lecture:

- Mobile human-computer interaction

- Adaptivity and adaptibility
- Information visualisation
- Tangible user interaction
- Usability engineering
- Usability and economics

Literature:
The course is based on a compilation of different sources; as additional sources and as a reference are recommended:

- Jacko, J.A. and Sears, A., (Eds.). Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications. Lawrence Erlbaum, Hillsdale, NJ, 2002.
- Hammond, J., Gross, T. and Wesson, J., (Eds.). Usability: Gaining a Competitive Edge. Kluwer Academic Publishers, Dordrecht, 2002.

Examination
Oral examination
Description:
The oral exam takes 30 minutes and is worth a total of 90 points. Depending on the number of attendees the form of the exam can be changed to a written exam with 90 minutes and a total of 90 points. The final form of the exam is announced in the first lecture at the beginning of the term.

During the semester students can do assignments, which are optional. They are 12 points in total. The type of optional homework assignments as well as the deadlines are announced in detail at the beginning of the term. If the oral exam is passed (as a rule 50% of the points have to be reached) the points from the assignments are a bonus and added to the points from the oral exam. In any case, a top grade of 1,0 is also reachable without solving the assignments.

Module Units

Human-Computer Interaction **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Mensch-Computer-Interaktion **Language:** German/English **Frequency:** every summer semester	**2,00 Weekly Contact Hours**

Contents:
Practical assignments based on the subjects of the lecture.

Literature:
Cf. lecture

Examination
Written examination / Duration of Examination: 90 minutes
Description:
In Abhängigkeit 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 mündlichen Prüfung können 90 Punkte erzielt werden. Die Prüfungsdauer wird im ersten Veranstaltungstermin mitgeteilt. 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 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 Prüfung auf jeden Fall auch ohne Punkte aus der Bearbeitung optionaler Studienleistungen erreichbar.	

Module HCI-Proj-B Project Human-Computer Interaction *Projekt Mensch-Computer-Interaktion*	6 ECTS / 180 h

(since SS20)
Person responsible for module: Prof. Dr. Tom Gross

Contents:
Practical work on a research topic of Human-Computer Interaction.

Learning outcomes:
Based on the knowledge and skills obtained in the human-computer interaction lectures and assignments a group of students will develop a small prototype based on current research topics. Central to this project is the development of skills regarding the implementation of systems as well as competencies regarding project management and teamwork.

Remark:
http://www.uni-bamberg.de/hci/leistungen/studium

The workload for this module is roughly structured as following:

- Participation in the kick-off meeting
- Participation in the group meetings
- Work on the project tasks alone and with the team
- Preparation of project discussions and presentation
- Exam preparation

The workload for each participant may vary over the different tasks based on the project definition and the joint coordination of tasks in the team

The default language of instruction in this course is German, but can be changed to English on demand. All course materials (incl. exams) are available in English

prerequisites for the module:
none

Recommended prior knowledge: Module Algorithms and Data Structures (MI-AuD-B) Module Interactive Systems (HCI-IS-B)	**Admission requirements:** Passing the exam

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Project Human-Computer Interaction **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Tom Gross, Scientific Staff Mensch-Computer-Interaktion **Language:** German/English **Frequency:** every winter semester **Contents:** The project covers diverse topics based on the contents of the courses. The project task is significantly more comprehensive than the normal assignments	**4,00 Weekly Contact Hours**

accompanying the lectures and therefore is solved in a small group. The results of the project are documented and demonstrated in a final presentation.

Literature:
To be announced in the course

Examination
Coursework Assignment and Colloquium / Duration of Examination: 30 minutes
Duration of Coursework: 4 months
prerequisites for module examination:
Regelmäßige Teilnahme an der Lehrveranstaltung
Description:
Documentation on the development process and project results as well as colloquium on the development process and project results.

Module HCI-Proj-M Project Human-Computer Interaction *Projektpraktikum Mensch-Computer-Interaktion*		6 ECTS / 180 h
(since SS20) Person responsible for module: Prof. Dr. Tom Gross		
Contents: Advanced practical work on a research topic of Human-Computer Interaction.		
Learning outcomes: Based on the knowledge and skills obtained in the human-computer interaction lectures and assignments a group of students develops a small prototype based on current research topics. Central to this project is the development of skills regarding the implementation of systems as well as competencies regarding project management and teamwork. Through the complexity of the task and the direct relation to on-going research at the human-computer interaction group this project is significantly different from the projects at Bachelor's level.		
Remark: http://www.uni-bamberg.de/hci/leistungen/studium The workload for this module is roughly structured as following: • Participation in the kick-off meeting • Participation in the group meetings • Work on the project tasks alone and with the team • Preparation of project discussions and presentation • Exam preparation The workload for each participant may vary over the different tasks based on the project definition and the joint coordination of tasks in the team The default language of instruction is German and can be changed to English based on students' needs. All course materials (incl. exams) are available in English.		
prerequisites for the module: none		
Recommended prior knowledge: Module Human-Computer Interaction (HCI-MCI-M) Module Algorithms and Data Structures (AI-AuD-B) - recommended		**Admission requirements:** Passing the exam
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Human-Computer Interaction **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Tom Gross, Scientific Staff Mensch-Computer-Interaktion **Language:** German/English **Frequency:** every summer semester **Contents:**	**4,00 Weekly Contact Hours**

The project will cover varying topics based on the contents of the courses. As normally the aspects of several courses are relevant, teams of students that have visited different courses will supplement each other. The project task is significantly more comprehensive than the normal assignments accompanying the lectures and therefore is solved in a small group. The results of the project are documented and demonstrated in a final presentation. **Literature:** To be announced in the course	
Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Documentation on the development process and project results as well as colloquium on the development process and project results.	

Module HCI-Proj1-M Research-Project Human-Computer Interaction

Forschungsprojektpraktikum Mensch-Computer-Interaktion

15 ECTS / 450 h

(since WS17/18)
Person responsible for module: Prof. Dr. Tom Gross

Contents:
Advanced practical work on a research topic of Human-Computer Interaction with scientific methods.

Learning outcomes:
Based on the knowledge and skills obtained in the human-computer interaction lectures and assignments a group of students work on a project on current research topics. Central to this project is the development of skills regarding the implementation of systems as well as competencies regarding project management and teamwork.

This research project addresses recent trends and is research oriented. The primary focus of this interaction project is on learning methods for the analysis, design, and technical realisation of interaction concepts.

Remark:
The workload for this module is roughly structured as following:

- Participation in the kick-off meeting
- Participation in the group meetings
- Work on the project tasks alone and with the team
- Preparation of project discussions and presentation
- Exam preparation

The workload for each participant may vary over the different tasks based on the project definition and the joint coordination of tasks in the team

The default language of instruction in this course is German, but can be changed to English on demand. All course materials (incl. exams) are available in English.

prerequisites for the module:
none

Recommended prior knowledge: Module Human-Computer Interaction (HCI-MCI-M)	**Admission requirements:** Passing the exam

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Human-Computer Interaction
Mode of Delivery: Practicals
Lecturers: Prof. Dr. Tom Gross, Scientific Staff Mensch-Computer-Interaktion
Language: German/English
Frequency: every summer semester

6,00 Weekly Contact Hours

Contents:
The project covers varying topics based on the contents of the courses. As normally the aspects of several courses are relevant, teams of students that have visited different courses will supplement each other. The project task is,

according to the 15 ECTS, complex and challenging. The results of the project are documented and demonstrated in a final presentation.

Literature:
To be announced in the course

Examination
Coursework Assignment and Colloquium / Duration of Examination: 30 minutes
Duration of Coursework: 4 months
prerequisites for module examination:
Regelmäßige Teilnahme an der Lehrveranstaltung
Description:
Documentation on the development process and project results as well as colloquium on the development process and project results.

The exact schedule of the project’s homework and colloquium are announced at the beginning of the term.

Module HCI-Proj2-M Research-Project Human-Computer Interaction *Forschungsprojektpraktikum Mensch-Computer-Interaktion*	15 ECTS / 450 h

(since WS17/18)
Person responsible for module: Prof. Dr. Tom Gross

Contents:
Advanced practical work on a research topic of Human-Computer Interaction with scientific methods.

Learning outcomes:
Based on the knowledge and skills obtained in the human-computer interaction lectures and assignments a group of students work on a project on current research topics. Central to this project is the development of skills regarding the implementation of systems as well as competencies regarding project management and teamwork.

This research project addresses recent trends and is research oriented. The primary focus of this interaction project is on learning methods for the analysis, design, and technical realisation of interaction concepts.

Remark:
The workload for this module is roughly structured as following:

- Participation in the kick-off meeting
- Participation in the group meetings
- Work on the project tasks alone and with the team
- Preparation of project discussions and presentation
- Exam preparation

The workload for each participant may vary over the different tasks based on the project definition and the joint coordination of tasks in the team

The default language of instruction in this course is German, but can be changed to English on demand. All course materials (incl. exams) are available in English.

prerequisites for the module:
none

Recommended prior knowledge: Module Human-Computer Interaction (HCI-MCI-M)		**Admission requirements:** Passing the exam
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Human-Computer Interaction **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Tom Gross, Scientific Staff Mensch-Computer-Interaktion **Language:** German/English **Frequency:** every winter semester **Contents:** The project covers varying topics based on the contents of the courses. As normally the aspects of several courses are relevant, teams of students that have visited different courses will supplement each other. The project task is,	**6,00 Weekly Contact Hours**

according to the 15 ECTS, complex and challenging. The results of the project are documented and demonstrated in a final presentation. **Literature:** To be announced in the course	

Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Documentation on the development process and project results as well as colloquium on the development process and project results. The exact schedule of the project’s homework and colloquium are announced at the beginning of the term.	

Module HCI-Prop-M Propaedeutic: Human-Computer-Interaction *Propädeutikum Mensch-Computer-Interaktion*		3 ECTS / 90 h
(since WS17/18) Person responsible for module: Prof. Dr. Tom Gross		
Contents: Scientific foundation of the research field of Human-Computer Interaction		
Learning outcomes: The aim of this module is a general introduction to and teaching of fundamental paradigms and scientific methods of the organisation, the written documentation, oral presentation of research activities in Human-Computer Interaction. The primary focus is on domain-specific documentation and presentation of designs, prototypes, and user studies.		
Remark: http://www.uni-bamberg.de/hci/leistungen/studium The workload for this module is roughly structured as following: • Participation in the course meetings (theoretical foundation; practical case studies): ca. 30 hours • Working on the case studies: ca. 30 hours • Preparation of presentation: ca. 15 hours • Writing of term paper: ca. 15 hours The default language of instruction in this course is German, but can be changed to English on demand. All course materials (incl. exams) are available in English.		
prerequisites for the module: none		
Recommended prior knowledge: none		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Propaedeutic: Human-Computer-Interaction **Mode of Delivery:** Seminar **Lecturers:** Prof. Dr. Tom Gross, Scientific Staff Mensch-Computer-Interaktion **Language:** German/English **Frequency:** every winter semester **Contents:** This seminar is concerned with the documentation and presentation of current concepts, technologies, and tools and user studies of human-computer interaction. **Literature:** The course is based on a compilation of different sources; as additional sources and as a reference are recommended:	**2,00 Weekly Contact Hours**

Jacko, Julie A., ed. Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications. (3rd ed.). Lawrence Erlbaum, Hillsdale, NJ, 2012.	
Examination Coursework Assignment with presentation / Duration of Examination: 30 minutes Duration of Coursework: 4 months **Description:** Written term paper and presentation on the chosen topic by the participant, incl. discussion	

Module HCI-Sem-B Bachelor-Seminar Human-Computer Interaction *Bachelorseminar Mensch-Computer-Interaktion*	3 ECTS / 90 h

(since WS17/18)
Person responsible for module: Prof. Dr. Tom Gross

Contents:
Active scientific work on current concepts, technologies and tools of Human-Computer Interaction

Learning outcomes:
The aim of this course is the acquisition of abilities to do research and presentation of topics in the field of human-computer interaction on basis of the existing literature. The focus lies on the development of skills that allow to critically and systematically review literature and to give presentations.

Remark:
http://www.uni-bamberg.de/hci/leistungen/studium

The workload for this module is roughly structured as following:

- Participation in the seminars (introduction to the topics, discussions, presentations): ca. 20 hours
- Literature review and getting familiar with the topic: ca. 25 hours
- Preparation of presentation: ca. 15 hours

Writing of term paper: ca. 30 hours

The default language of instruction is German and can be changed to English based on students' needs. All course materials (incl. exams) are available in English

prerequisites for the module:
none

Recommended prior knowledge: Module Interactive Systems (HCI-IS-B)		**Admission requirements:** Passing the exam
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Human-Computer Interaction **Mode of Delivery:** Seminar **Lecturers:** Prof. Dr. Tom Gross, Scientific Staff Mensch-Computer-Interaktion **Language:** German/English **Frequency:** every summer semester **Contents:** Based on the knowledge and skills obtained in the human-computer interaction lectures and assignments varying, current research topics are discussed in this seminar. Thereby, aspects of several courses are of relevance. **Literature:** To be announced at the beginning of the course	**2,00 Weekly Contact Hours**
Examination Coursework Assignment with presentation / Duration of Examination: 30 minutes Duration of Coursework: 4 months	

Description: Written term paper and presentation on the chosen topic by the participant, incl. discussion	

Module HCI-Usab-M Usability in Practice *Usability in der Praxis*	6 ECTS / 180 h
(since WS17/18) Person responsible for module: Prof. Dr. Tom Gross	

Contents:
Practical work on a real-world topic of Human-Computer Interaction.

Learning outcomes:
In this course the knowledge and skills obtained in the human-computer interaction lectures and assignments are applied in practice. Based on real use cases from industry contexts students will analyse the usability of existing concepts and systems and gather requirements for innovative concepts. Central to this course is the development of skills regarding the practical application of methods as well as competencies regarding project management and teamwork.

Remark:
http://www.uni-bamberg.de/hci/leistungen/studium

The workload for this module is roughly structured as following:

- Participation in the kick-off meeting
- Participation in the group meetings
- Work on the tasks alone and with the team
- Preparation of discussions and presentation
- Exam preparation

The workload for each participant may vary over the different tasks based on the task definitions and the joint coordination of tasks in the team

The default language of instruction is German and can be changed to English based on students' needs. All course materials (incl. exams) are available in English

prerequisites for the module:
none

Recommended prior knowledge: Module Human-Computer Interaction (HCI-MCI-M)		**Admission requirements:** Passing the exam
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Usability in der Praxis **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Tom Gross, Scientific Staff Mensch-Computer-Interaktion **Language:** German/English **Frequency:** every summer semester **Contents:** The course covers diverse topics from human-computer interaction that are cooperatively solved with companies. They typically range from specifying challenges to selecting and applying methods as well as analysing the captured data to deriving conclusions. The task is significantly more comprehensive than	**4,00 Weekly Contact Hours**

the normal assignments accompanying the lectures and therefore is solved in a small group. The results are documented and demonstrated in a final presentation. **Literature:** To be announced in the course	
Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:** regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Documentation on the project process and results as well as colloquium on the project process and results.	

Module IIS-Sem-B Bachelor Seminar Industrial Information Systems *Bachelorseminar Industrielle Informationssysteme*		3 ECTS / 90 h
(since WS17/18) Person responsible for module: Prof. Dr. Sven Overhage		
Contents: Seminar with changing topics in industrial information systems. The specific seminar topic will be announced by the examiner at the beginning of the winter semester.		
Learning outcomes: none		
Remark: The main language of instruction in this course is German. The exam may be delivered in English on demand.		
prerequisites for the module: none		
Recommended prior knowledge: none		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Bachelor Seminar Industrial Information Systems **Mode of Delivery:** Introductory seminar **Lecturers:** Prof. Dr. Sven Overhage **Language:** German **Frequency:** every winter semester **Contents:** The specific seminar topic will be announced by the examiner at the beginning of the winter semester.	**2,00 Weekly Contact Hours**

Examination	
Coursework Assignment with presentation / Duration of Examination: 30 minutes Duration of Coursework: 3 months	

Module IIS-Sem-M Master Seminar Industrial Information Systems *Masterseminar Industrielle Informationssysteme*		3 ECTS / 90 h
(since WS17/18) Person responsible for module: Prof. Dr. Sven Overhage		
Contents: Seminar with changing topics in industrial information systems. The specific seminar topic will be announced by the examiner at the beginning of the winter semester.		
Learning outcomes: none		
Remark: The main language of instruction in this course is German. The exam may be delivered in English on demand.		
prerequisites for the module: none		
Recommended prior knowledge: none		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Master Seminar Industrial Information Systems **Mode of Delivery:** Introductory seminar **Lecturers:** Prof. Dr. Sven Overhage **Language:** German **Frequency:** every winter semester **Contents:** The specific seminar topic will be announced by the examiner at the beginning of the winter semester.	**2,00 Weekly Contact Hours**

Examination Coursework Assignment with presentation / Duration of Examination: 30 minutes Duration of Coursework: 3 months	

Module ISDL-DEXP-B Digital Experimentation

Digital Experimentation

6 ECTS / 180 h

Person responsible for module: Dr. Christoph Weinert

Contents:

Durch das Internet kamen sogenannte Online-Experiment auf, die gerade von großen Tech-Konzernen wie Google, Facebook oder Alibaba genutzt werden, um Produkte und Dienstleistungen zu evaluieren. Darüber hinaus können Experimente dabei helfen

sozialen und wirtschaftlichen Aktivitäten, an denen sich Menschen online beteiligen besser zu verstehen. Das liegt daran, dass Experimente sowohl in der Forschung als auch in der Praxis eine exzellente Möglichkeit sind, um Reiz-Reaktions-Beziehungen abzubilden und untersuchen zu können. In einem Experiment wird ein Reiz bewusst manipuliert, um die darauffolgenden Reaktionen messen zu können während die Kontextvariablen stabil gehalten oder kontrolliert werden. Die Durchführung von Experimenten hat eine lange Historie in den Naturwissenschaften, allerdings wird diese Methode immer häufiger in die Praxis und Forschung der Wirtschaftsinformatik eingesetzt.

Die Vorlesung gliedert sich ausgehend von generellen Einsatz von Experimenten in Forschung und Praxis bis hin zur konkreten Planung, Aufbau und Durchführung von verschiedenen Arten von Experimenten (z.B. Online-Experimente, Laborexperimente, Feldexperimente).

Learning outcomes:

Das Modul vermittelt ein grundlegendes Verständnis sowie Kenntnisse zu Planung, Aufbau, Durchführung, und Auswertung für verschiedene Arten von Experimenten (z.B. Online-Experimente, Laborexperimente, Feldexperimente). Das Modul befähigt die Teilnehmer zur eigenständigen Durchführung von Experimenten in wissenschaftlichen wie auch praktischen Kontexten.

Remark:

Der Arbeitsaufwand für dieses Modul gliedert sich ungefähr wie folgt:

- Teilnahme an Vorlesung und Übung: insgesamt 42 Stunden
- Vor- und Nachbereitung der Vorlesung und Übung (inkl. Recherche und Studium zusätzlicher Quellen): 56 Stunden
- Bearbeiten der Übungsaufgaben: insgesamt 40 Stunden
- Prüfungsvorbereitung ink. Prüfung: 42 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)

prerequisites for the module:

keine

Recommended prior knowledge: keine	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Experimentelle Forschung in der Wirtschaftsinformatik

Mode of Delivery: Lectures and Practicals

Lecturers: Dr. Christoph Weinert

Language: German

Frequency: every winter semester

2,00 Weekly Contact Hours

Contents:
Die Inhalte der Vorlesung werden anhand von praktischen Beispielen vertieft. Die Studierenden bekommen die Möglichkeit ein eigenes Experiment zu planen, durchzuführen und auszuwerten. Hierbei werden unter anderem psychologische Tests und objektive Messmethoden (z.B. Eye-tracking, Skin conductance) genutzt.

Literature:

Jarvenpaa, S. L., Dickson, G. W., and DeSanctis, G. 1985. “Methodological Issues in Experimental IS Research: Experiences and Recommendations,” MIS Quarterly (9:2), pp. 141–156.

Karahanna, E., Benbasat, I., Bapna, R., and Rai, A. 2018. “Opportunities and Challenges for Different Types of Online Experiments,” MIS Quarterly (42:4), pp. iii–x.

Weitere Literatur wird in der Vorlesung bekannt gegeben.

Examination
Written examination / Duration of Examination: 90 minutes
prerequisites for module examination:
keine
Description:
In der Klausur werden die in der Vorlesung und Übung behandelten Inhalte geprüft. Es können 90 Punkte erzielt werden. Durch die freiwillige Abgabe von semesterbegleitenden Studienleistungen (Planung und Durchführung eines Experiments) 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 bekannt gegeben. Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.

<table>
<tr><td colspan="2"><b>Module ISM-FIISM-B Fundamentals of International IS Management</b>
<i>Fundamentals of International IS Management</i></td><td>6 ECTS / 180 h</td></tr>
<tr><td colspan="3">(since SS20)
Person responsible for module: Prof. Dr. Daniel Beimborn</td></tr>
<tr><td colspan="3"><b>Contents:</b>
This module equips IISM students with the basics of their IISM curriculum and serves as introductory course.
Building on the basics of information systems (such as learned about in ISM-EidWI-B, SNA-WIM-B or in a similar course), we will develop a deeper understanding about information systems (IS) management, about international management, and about particularities of IS management in an international context. Accordingly, the course consists of three parts, covering those areas.</td></tr>
<tr><td colspan="3"><b>Learning outcomes:</b>
After having accomplished this course, students will have an understanding about particularities of general management and IS management in an international context. They will be able to handle basic IS management tasks in an international environment and they will be sensitive to challenges caused by international and intercultural settings as well as by virtual collaboration.</td></tr>
<tr><td colspan="3"><b>Remark:</b>
The workload of 180 academic hours is allocated as follows:
• 56h for participating in class
• 80h for preparing classes (i.e., retrieving and studying literature and case studies, accomplishing small assignments) and reviewing course material after class
• 44h for self-managed studies and preparing for the final exam including the final exam</td></tr>
<tr><td colspan="3"><b>prerequisites for the module:</b>
none</td></tr>
<tr><td colspan="2"><b>Recommended prior knowledge:</b>
ISM-EidWI-B (or any equivalent „Introduction into IS“ course) is required. SNA-WIM-B is recommended, but not necessary (students can catch up the relevant parts by reading some extra literature).</td><td><b>Admission requirements:</b>
none</td></tr>
<tr><td><b>Frequency:</b> every summer semester</td><td><b>Recommended semester:</b>
from 4.</td><td><b>Minimal Duration of the Module:</b>
1 Semester</td></tr>
</table>

<table>
<tr><td colspan="2"><b>Module Units</b></td></tr>
<tr><td><b>Fundamentals of International IS Management</b>
<b>Mode of Delivery:</b> Lectures
<b>Lecturers:</b> Prof. Dr. Daniel Beimborn
<b>Language:</b> English
<b>Frequency:</b> every summer semester
<b>Contents:</b>
In this course, students will gain a deeper understanding of IS management tasks and issues with a particular focus on international contexts and environments. Therefore, the course will first give an overview about important IS management</td><td><b>2,00 Weekly Contact Hours</b></td></tr>
</table>

fields, then give an introduction to (general) international management, and finally combine both foundational parts by discussing particularities of managing information systems in an international context (i.e., the core of IISM). Accordingly, the course will consist of three parts:

Part 1: Information Systems Management (IS Management):

- 1.1 Strategic IS Management: IT Strategy and Strategic Alignment, IT Organization and IT

Governance

- 1.2 Tactical IS Management: IS Development & Project Management, IS Procurement &

Outsourcing Management

- 1.3 Operational IS Management; IT Service Management, IT Operations

Part 2: International Management

- 2.1 Theoretical and Conceptual Foundations of International Management
- 2.2 Organization of International Firms
- 2.3 Foreign Market Entry Strategies
- 2.4 Intercultural Management and Virtual Teams

Part 3: International IS Management

- 3.1 Managing Global IT Organizations and People
- 3.2 Managing Global IT/Software Development Projects and System Roll-outs
- 3.3 Managing Offshore IT Outsourcing
- 3.4 Global Issues of IS Management – Ethics and Sustainability

While parts 1 and 2 will be primarily taught using teaching books, lectures, and exercises, part 3 will draw on the knowledge and skills gained in the first two parts by applying a predominantly case-based teaching approach.

Literature:
Will be announced in class.

Module Units	
Fundamentals of International IS Management **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Daniel Beimborn **Language:** English **Frequency:** every summer semester **Contents:** The content of the course will be reviewed by assignment tasks and discussion of case studies **Literature:** see lecture	**2,00 Weekly Contact Hours**
Examination Written examination / Duration of Examination: 90 minutes	

Description: In the exam, the contents covered in the module (lecture, exercise, readings) are examined. 90 points can be scored. It is possible to earn bonus points for the exam during the lecture term. Earned bonus points will be credited to the results if the exam has been passed successfully. Bonus points can be earned by completing a voluntary, written coursework in which students independently have to work on transfer tasks related to the lecture course. At the beginning of the course it will be announced whether bonus points are offered. If bonus points are offered, the number, type, scope, and duration of the assignments as well as the number of attainable bonus points will be announced at this time. A final grade of 1.0 can be achieved without bonus points from the coursework.	

Module ISM-IOM-M International Outsourcing Management *International Outsourcing Management*	6 ECTS / 180 h

(since WS20/21)

Person responsible for module: Prof. Dr. Daniel Beimborn

Contents:

- Grundlagen des Outsourcings: Definitionen, grundlegende Konzepte und Arten von Outsourcing; Geschichte, Trends; Märkte und Wachstum; Überblick über die wissenschaftliche Forschung im Outsourcing-Kontext
- Outsourcing-Gründe und grundlegende Theorien: Ökonomische und strategische Gründe für Outsourcing; Theorien zu Kosten- und strategischen Vorteilen, Kernkompetenzen usw.
- Outsourcing-Risiken: Ökonomische und strategische Risiken durch Outsourcing
- Outsourcing-Entscheidungen: Analyse der Nutzenpotenziale und Risiken durch Outsourcing; Modelle zur Bewertung der Vorteilhaftigkeit von Outsourcing; Prozess und Bewertungskriterien zur Auswahl von Dienstleistern
- Outsourcing-Verträge: Gestaltung und Verhandlung von Outsourcing-Verträgen und Service-Level-Agreements; Verhandlung mit einem Dienstleister; ausgewählte regulatorische Rahmenbedingungen
- Organisatorische Vorbereitungen im eigenen Unternehmen („Outsourcing Readiness"); Vorbereitung und Durchführung des Transitionsvorgangs
- Outsourcing-Governance: Aufbau einer Outsourcing-Governance zur Steuerung der Dienstleisterbeziehung; Kontrolle, Change-Management und Beziehungsmanagement; Management des Wissensaustausches und Fördern von Innovationen
- Besonderheiten beim Cloud Computing: Grundlegende Konzepte und Arten von Cloud Computing als besonderer Form von Outsourcing; Spezifische Vorteile, Herausforderungen und Risiken von Cloud Computing; spezifische Aspekte bei Cloud-basierten Sourcing-Entscheidungen sowie bei einer Cloud-Computing-Governance
- Offshore- und Nearshore-Outsourcing: Besonderheiten hinsichtlich Risiken, Kosten und Chancen; Bedeutung von und Umgang mit kulturellen Unterschieden; Globale IT-Delivery-Modelle
- Ökonomische und gesellschaftliche Auswirkungen von Outsourcing und Offshoring: Gesellschaftliche Reaktionen und Veränderungen; Implikationen für nationale Arbeitsmärkte und globale IT-Märkte

Learning outcomes:

Die Teilnehmer können Chancen und Risiken von IT-Outsourcing in Firmen identifizieren, Outsourcing-Projekte planen (Outsourcing-Strategie, Business Case, Auswahl unterschiedlicher Sourcing-Modi und Vendorenmodelle) und implementieren (Vertragsmanagement, Outsourcing-Governance, Beziehungsmanagement, Wissenstransfer). Damit sind Sie in der Lage,

- die grundlegenden Argumente für das Treffen von IT-Outsourcing-Entscheidungen zu identifizieren und zu evaluieren(Wann macht Outsourcing Sinn?),
- IT-Outsourcing-Optionen zu identifizieren und zu bewerten (Welche Form von Outsourcing ist sinnvoll?),
- IT-Outsourcing-Projekte zu planen und zu managen (Wie kann ein erfolgreicher Transfer zum Dienstleister gewährleistet werden?),
- eine Outsourcing-Governance zu implementieren (Wir wird gesteuert? Wer hat welche Verantwortlichkeiten inne?),
- IT-Outsourcing-Beziehungen zu gestalten und zu managen (Vertragsmanagement, Kontrolle, Beziehungsmanagement, Wissenstransfer) sowie

Nearshore- und Offshore-IT-Outsourcing-Optionen zu identifizieren und zu bewerten.

Remark:

Der Arbeitsaufwand für dieses Modul gliedert sich ungefähr wie folgt:

- Teilnahme an Vorlesung und Übung: insgesamt 45 Stunden
- Vor- und Nachbereitung der Vorlesung und Übung (inkl. Recherche und Studium zusätzlicher Quellen): 90 Stunden
- Prüfungsvorbereitung inkl. Prüfung: 45 Stunden (basierend auf dem bereits im obigen Sinne erarbeiteten Stoff)

Für das erfolgreiche Absolvieren des Moduls ist die regelmäßige Teilnahme an den Lehrveranstaltungen und die Vorbereitung von Fallstudien/Readings empfohlen.

prerequisites for the module:

none

Recommended prior knowledge: none		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

International Outsourcing Management **Mode of Delivery:** **Lecturers:** Prof. Dr. Daniel Beimborn **Language:** German **Frequency:** every winter semester	**4,00 Weekly Contact Hours**

Contents:

Outsourcing, der Fremdbezug von Leistungen von einem Dienstleister, ist eine wichtige Handlungsoption für IT-Manager. In diesem Modul werden Grundlagen, Vor- und Nachteile des Outsourcing sowie Entscheidungsmodelle, Vorgehensweisen, „Good Practices" („warum outsourcen, was outsourcen, wie outsourcen?"), aber auch Probleme und kulturelle Hürden im Bereich IT-Outsourcing und -Offshoring vermittelt, diskutiert und angewendet. Neben klassischem Outsourcing werden auch Cloud-basierte IT-Delivery-Modelle und die entsprechenden Management-besonderheiten betrachtet. Auf Basis von ausführlichen Vorlesungsunterlagen und mittels Diskussion von Fallstudien werden die Management-Anforderungen für die Gestaltung eines erfolgreichen Outsourcing-Arrangements umfassend vermittelt. Anhand von Übungsaufgaben werden die Inhalte zusätzlich ausführlich vertieft. Eine Vorbereitung der Lektüre für jede Einheit ist zwingend erforderlich. Die Universität Bamberg ist der einzige deutsche Academic Alliance Partner der International Association of Outsourcing Professionals (IAOP), die sich die globale Qualitätssteigerung und Standardisierung von Outsourcing-Management- Kompetenzen zum Ziel gesetzt hat. Entsprechend werden maßgeblich auch internationale (englischsprachige) Lehrmaterialien der IAOP verwendet.

Literature:

Beimborn, D. 2008. Cooperative Sourcing - Simulation Studies and Empirical Data on Outsourcing Coalitions in the Banking Industry. Wiesbaden: Gabler.

Carmel, E., and Tjia, P. 2005. Offshoring Information Technology - Sourcing and Outsourcing to a Global Workforce. Cambridge: Cambridge University Press. IAOP. 2014. Outsourcing Professional Body of Knowledge. Zaltbommel: VanHaren Publishing. Lacity, M.C., Khan, S.A., and Willcocks, L.P. 2009. "A Review of the IT Outsourcing Literature: Insights for Practice," Journal of Strategic Information Systems (18:3), pp 130-146. Oshri, I., Kotlarksy, J., and Willcocks, L. 2015. The Handbook of Global Outsourcing and Offshoring. London, New York: Palgrave. Weitere Literatur zu den einzelnen Themen wird in den jeweiligen Vorlesungen bekannt gegeben.	

Examination Written examination / Duration of Examination: 90 minutes **Description:** 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.	

Module KInf-MobAss-M Mobile Assistance Systems *Mobile Assistance Systems*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium

(since WS17/18)
Person responsible for module: Prof. Dr. Christoph Schlieder

Contents:

The module introduces students into the research literature on mobile assistance systems. It consists of two parts, a lecture and reading course (Vorlesung) which covers methods and lab sessions in which the methods are applied in a software development project (Übung).

For more detail refer to the content descrition of the lecture.

Learning outcomes:

After completion of this module, students will be able to

- explain and compare the fundamental concepts of mobile assistance systems
- describe and analyze methods for geo-positioning and place modeling
- critically discuss approaches to specific types of mobile applications such as:
- geographic recommender, tourist guides, location-based games, documentation systems

Remark:

The main language of instruction in this course is English. The lab may be delivered in German if all participating students are fluent in German.

The total workload of 180 hrs. is split approximately as follows:

- 45 hrs. attending lecture and lab sessions
- 30 hrs. preparing and reviewing the lectures
- 30 hrs. preparing and reviewing the lab sessions
- 45 hrs. working on the written assignment
- 30 hrs. preparation for the exam

prerequisites for the module:
none

Recommended prior knowledge: Students are expected to come with general programming and software engineering skills and to be familiar with formal methods in computer science	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Mobile Assistance Systems **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften **Language:** English **Frequency:** every summer semester **Contents:**	**2,00 Weekly Contact Hours**

Students solve a small number of programming problems related to mobile assistance systems. The software is developed in Android and typically tested on GPS smartphones. Students should come with basic Java programming skills and can familiarize themselves with Android during the course. Solutions to the programming problems are presented by the students in a colloquium (20 min) at the end of the semester. **Literature:** Literature and online ressouces are presented in the course.	
Examination Colloquium / Duration of Examination: 20 minutes **Description:** In the lab, students are working on a software development project. At the end of the semester, each student presents the results of her or his lab project (Kolloquium). The grade for the lab project contributes 50% to the final grade.	

Module Units	
Mobile Assistance Systems **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Christoph Schlieder **Language:** English/German **Frequency:** every summer semester **Contents:** A digital travel guide running on a smart phone and a CAD-based system for the documentation of built heritage with a TabletPC are two examples of software solutions designed to assist mobile users, that is, examples of mobile assistance systems. The course introduces students to the research literature on mobile assistance systems and enables them to put concepts and methods into practice. Introductions to positioning technologies, place models, and mobile applications such as geographic recommender or location-based games are presented in form of a lecture. Other parts of the material are organized in form of a reading course in which the students critically analyze and discuss the research literature. **Literature:** Taylor, George and Blewitt, Geoff (2006): Intelligent Positioning: GIS-GPS Unification, Wiley & Sons, ISBN 0470850035 Further literature is presented in the course.	**2,00 Weekly Contact Hours**
Examination Written examination / Duration of Examination: 60 minutes **Description:** The written exam covers the material of the reading couse. The grade of the written exam contributes 50% to the final grade.	

Module KInf-SemInf-M Semantic Information Processing *Semantische Infomationsverarbeitung*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(since WS19/20) Person responsible for module: Prof. Dr. Christoph Schlieder	

Contents:

The module introduces students into the research field of semantic information processing. It consists of two parts, a lecture (Vorlesung) which covers the basic methods and lab sessions in which the methods are applied to problems (Übung).

For more detail refer to the content descritiption of the lecture.

Learning outcomes:

After completion of this module, students will be able to

- explain and compare the fundamental concepts of semantic information processing
- describe and analyze methods for problem solving by heuristic search
- critically discuss different approaches to knowledge representation
- select algorithms that are appropriate for a given type of application problem

Remark:

The main language of instruction in this course is English. The lab sessions may be delivered in German if all participating students are fluent in German.

The total workload of 180 hrs. is split approximately as follows:

- 45 hrs. attending lecture and lab sessions
- 30 hrs. preparing and reviewing the lectures
- 30 hrs. preparing and reviewing the lab sessions
- 45 hrs. working on the written assignment
- 30 hrs. preparation for the exam

prerequisites for the module:

none

Recommended prior knowledge: Students are expected to come with general programming skills and to be familiar with formal methods in computer science		**Admission requirements:** keine
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Lectures on Semantic Information Processing **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Christoph Schlieder **Language:** German **Frequency:** every winter semester	**2,00 Weekly Contact Hours**
Contents: Semantic information processing addresses problems in which software systems need to represent knowledge, not just data. Facts from different knowledge	

sources are combined and integrated by machine reasoning processes. The services of the Semantic Web provide a prominent example for applications that make extensive use of knowledge representation and reasoning. The lecture introduces into the computational methods and tools for semantic information processing which have been developed by Artificial Intelligence research. Topics covered include problem solving by heuristic search, constraint solving, search strategies for games, representations for domain-specific knowledge, reasoning with formal ontologies, technologies of the Semantic Web, machine learning and knowledge discovery. The design of intelligent agents and agent systems is adopted as unifying perspective for presenting the material. Applications from different fields such as geographic information systems, digital libraries, and social computing illustrate how the methods from semantic information processing are used to build intelligent assistant systems.

Literature:

Russell, S., Norvig, P. & Davis, E. (2010): Artificial Intelligence. A Modern Approach. 3rd. Upper Saddle River: Prentice Hall.

Hitzler, P.; Krötzsch, M.; Rudolph, S. (2010): Foundations of Semantic Web technologies. CRC Press

2. Übung Semantische Informationsverarbeitung

2,00 Weekly Contact Hours

Mode of Delivery: Practicals

Lecturers: Scientific Staff Angewandte Informatik in den Kultur-, Geschichts- und Geowissenschaften

Language: German

Frequency: every winter semester

Contents:

The course applies the concepts and methods taught in the lecture by solving practical exercises. Most of the exercises can be completed with paper and pencil while some include programming in Java or working with software tools for semantic information processing. The solutions to the exercises are prepared as homework and presented by the students during the lab sessions.

Literature:

siehe Vorlesung

Examination

Written examination / Duration of Examination: 90 minutes

Description:

The written exam covers the material presented in the lecture and the lab sessions

Module KTR-GIK-M Foundations of Internet Communication *Grundbausteine der Internet-Kommunikation*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium

(since SS20)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:

The course provides an introduction to the theoretical foundations of important technical issues related to the fundamentals of Internet communication, the data link layer, routing and transport protocols in IP networks, as well as advanced topics such as real-time communication and security in IP networks. The implementation of the learnt concepts in terms of predetermined configuration tasks in the communication laboratory by small teams of students constitutes the tutorial part of the course. For this purpose, guidelines, technical instructions, and tools will be provided.

The implementation tasks include the configuration and testing of computer networks in the laboratory setting. Operating system and required software components like Wireshark, Atheris and Vyatta software router will be provided. The basic handling of the hardware and software itself will be perfomed by the students as part of their individual intellectual efforts within te couse.

Learning outcomes:

The important skill to provide a qualified assessment of current communicaton technologies and corresponding practical knowledge can only be acquired by team-oriented processes subject to time constraints and the clear specification of technical and administrative objectives. In the course Foundations of Internet Communication and its tutorials in the router laboratory students will learn to work independently with a high level of responsibility as self-confident member of a successful team.

It is the objective of the course that the students acquire practical knowledge on modern data communication in Internet and learn how communication concepts can be developed, implemented and judged with th highest level of expertise.

The course is open to bachelor students in their transition phase to the master prgram. It attempts to prepare for the job in communication industry related fields. Master students in the first semester and exchange students from abroad are invited to join the course.

Remark:

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

The workload is composed of the following items:

- participation in lectures, tutorials in the laboratory, laboratory meetings: 45 hours
- preparation, execution, post-processing of lectures and tutorials in the laboratory: 100 hours
- preparation of the examination: 35 hours

prerequisites for the module:

none

Recommended prior knowledge: • data communication similar to module KTR-Datkomm-B • fundamental knowledge on programming in JAVA (or C++) • working knowledge on LINUX is recommended, but not assumed Module Algorithms and Data Structures (AI-AuD-B) - recommended	**Admission requirements:** governed by examination regulations (StuFPO)

Module Introduction to Algorithms, Programming and Software (DSG-EiAPS-B) - recommended Module Data communication (KTR-Datkomm-B) - recommended		
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Foundations of Internet Communication
Mode of Delivery: Lectures and Practicals
Lecturers: Prof. Dr. Udo Krieger
Language: English/German
Frequency: every summer semester

4,00 Weekly Contact Hours

Learning outcome:

The important skill to provide a qualified assessment of current communicaton technologies and corresponding practical knowledge can only be acquired by team-oriented processes subject to time constraints and the clear specification of technical and administrative objectives. In the course Foundations of Internet Communication and its tutorials in the router laboratory students will learn to work independently with a high level of responsibility as self-confident member of a successful team.

It is the objective of the course that the students acquire practical knowledge on modern data communication in Internet and learn how communication concepts can be developed, implemented and judged with th highest level of expertise.

The course is open to bachelor students in their transition phase to the master prgram. It attempts to prepare for the job in communication industry related fields. Master students in the first semester and exchange students from abroad are invited to join the course.

Contents:

The course provides an introduction to the theoretical foundations of important technical issues related to the fundamentals of Internet communication, the data link layer, routing and transport protocols in IP networks, as well as advanced topics such as real-time communication and security in IP networks. The implementation of the learnt concepts in terms of predetermined configuration tasks in the communication laboratory by small teams of students constitutes the tutorial part of the course. For this purpose, guidelines, technical instructions, and tools will be provided.

The implementation tasks include the configuration and testing of computer networks in the laboratory setting. Operating system and required software components like Wireshark, Atheris and Vyatta software router will be provided. The basic handling of the hardware and software itself will be perfomed by the students as part of their individual intellectual efforts within the couse.

The organization of the laboratories is following the framework of industry. It comprises definition, preparation, implementation and presentation phases. An incremental processing is performed like in industrial projects. It means

- a segmentation into specific work packages,

- its division into tasks and subtasks including milestones
- the presentation of intermediate results
- a final report with presentation

Further laboratories related to current research issues in "Future Generation Internet" will be integrated into the course on demand. Details are discussed in the first lecture.

An actual list of studied topics and related references are presented in the first lecture.

The language of the course wil be announced during the first lecture.

Literature:
Foundations:

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

Further references related to specific workpackages:

- Kurose, J., Ross, K.W.: Computer Networking – a Top-Down Approach, Addison-Wesley, 2013 .
- Tanenbaum, A. S.: Computer Networks, Pearson Education, 2010.
- Leon-Garcia, A., Widjaja, I.: Communication Networks, McGraw-Hill, Boston, 2nd ed. 2004.
- Flaig, G., u.a.: Internet-Telefonie, Open source Press, München, 2006.

An up-to-date list is provided by the course.

Examination
Coursework Assignment and Colloquium / Duration of Examination: 30 minutes
Duration of Coursework: 4 months
Description:
The evaluation of the course will take place after completion of all lectures within the examination cycle. It is based on following items:

- assessment of the chapters composed by the candidate in the final course report about all workpackages written by a team of students
- presentation and explanation of specific tasks and outcomes of laboratories by an individual colloquium lasting 30 minutes

The evaluation rules of these components will be announced during the first lecture. The overall individual grading has to reach the level "satisfactory/ ausreichend (4.0)" to pass the examination of the module.

The language of the examination wil be announced during the first lecture.

Module KTR-MAKV-M Modeling and Analysis of Communication Networks and Distributed Systems *Modellierung und Analyse von Kommunikationsnetzen und Verteilten Systemen*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium

(since WS17/18)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:

The course deals with the analysis and performance evaluation of complex distributed systems such as telecommunication systems, computer networks and complex networks as well as cloud computing systems. The latter are transformed to abstract system-theoretical models and their associated parameters. The models are used to analyze the system behavior, and to predict relevant performance metrics such as utilization, throughput, waiting and response times of request, person or data flows in distributed systems or social networks. Such predictions have great importance regarding economic or technical design and decision processes in future generation networks and their distributed service architectures.

The course presents the modeling of distributed systems and discusses associated description methods such as relevant load and machine models. The system-theorectical analysis of these models and the included resource assignment and management strategies are sketched based on simple analytic methods like Markov chains, algebraic and numerical solution methods for queueing models.

Learning outcomes:

It is the objective of the course to teach students the fundamentals of measurement, analysis, and performance evaluation methods in modern computer and communication networks, and distributed systems. Students will learn how they can apply the underlying system-theoretical monitoring, modeling, and analysis techniques to a given technical context. The application of the sketched models and methods is illustrated by exercises covering views of distributed systems with a realistic characteristic. Students are encouraged to apply a given methodology to new technical contexts and scientific tasks.

Remark:

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

prerequisites for the module:

none

Recommended prior knowledge: • solid knowledge of calculus (like Mathematik I) and linear algebra (like Mathematik für Informatik 2) • basic knowledge of probability theory and statistics • programming experience in JAVA (or C++)	**Admission requirements:** governed by examination regulations (StuFPO)

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Modeling and Analysis of Communication Networks and Distributed Systems **Mode of Delivery:** Lectures and Practicals **Lecturers:** Prof. Dr. Udo Krieger	**4,00 Weekly Contact Hours**

Language: English/German
Frequency: every summer semester

Learning outcome:
It is the objective of the course to teach students the fundamentals of measurement, analysis, and performance evaluation methods in modern computer and communication networks, and distributed systems. Students will learn how they can apply the underlying system-theoretical monitoring, modeling, and analysis techniques to a given technical context. The application of the sketched models and methods is illustrated by exercises covering views of distributed systems with a realistic characteristic. Students are encouraged to apply a given methodology to new technical contexts and scientific tasks.

Contents:
The course deals with the analysis and performance evaluation of complex distributed systems such as telecommunication systems, computer networks and complex networks. The latter are transformed to abstract system-theoretical models and their associated parameters. The models are used to analyze the system behavior, and to predict relevant performance metrics such as utilization, throughput, waiting and response times of request, person or data flows in distributed systems or social networks. Such predictions have great importance regarding economic or technical design and decision processes in future generation networks and their distributed service architectures.

The course presents the modeling of distributed systems and discusses associated description methods such as relevant load and machine models. The system-theorectical analysis of these models and the included resource assignment and management strategies are sketched based on simple analytic methods like Markov chains, algebraic and numerical solution methods for queueing models, and simulative analysis schemes.

The content of the lectures is illustrated by exercises and laboratories covering important performance aspects in high-speed networks and distributed systems. Knowledge and skills to perform an efficient system analysis, system monitoring, and performance evaluation will be trained in this manner. The independent processing of tasks, the qualified presentation and critical discussion of the outcomes by teams of students is part of the course. It improves the technical understanding and provides means to work as project leader in industry on those topics.

The language of the course wil be announced during the first lecture.

Literature:

- G. Bolch, S. Greiner, H. de Meer, K. S. Trivedi: Queueing Networks and Markov Chains. Wiley, 2nd ed., 2006.
- R. Nelson: Probability, Stochastic Processes, and Queueing Theory. Springer, 1995.

A list of further references is presented in the first lecture.

Examination
Oral examination / Duration of Examination: 30 minutes
Description:

30 minutes oral examination related to the technical topics of all lectures and practicals. The language of the examination wil be announced during the first lecture.	

Module KTR-MMK-M Multimedia Communication in High Speed Networks *Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium

(since WS17/18)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:
Based on the foundations of data communication, this advanced course of the master program presents the design of high-speed networks (HSN) and the advanced protocol elements of the signaling and user plane that are required to implement new real-time and multimedia services. It includes the digital switching technologies and protocol stacks of HSNs, the quality-of-service architectures, as well as the traffic management protocols of these next generation IP networks. The extension of the TCP/IP protocol stack to realize communication relations among mobile or stationary end systems that are supported by quality-of-service guarantees and associated improved switching concepts are discussed in detail by lectures of the course.

These lectures focus on effective access technologies and new transport and QoS-architectures in the core network like Diffserv, MPLS and GMPLS. Further, enhancement of IPv4 by IPv6 switching and the extension of TCP by moern multipath concepts such as MPTCP and SCTP are presented. Advanced QoS-management concepts, effective resource and traffic management schemes like buffer management by RED, RIO or schedluing by WFQ, are discuessed, too. Furthermore, we present new architectures for next generation networks (NGNs) such as software-defined networks and information-centric networks.

Modern multimedia service architectures with interactive applications for third to fourth generation Internet like Web applications based on HTTP 2.0, WebRTC, peer-to-peer VoIP and media streaming applications are sketched.

The course can be supplemented by the module Foundations of Internet Communcation (KTR-GIk-M) with its instructive tasks executed in the router laboratory, by master seminars and projects or a master thesis on related topics in next generation networks.

Learning outcomes:
The students will be enabled to work independently according to the highest scientific standards on design and analysis tasks associated with high-speed network protocols. They will learn about the fundamentals of multimedia communication in high-speed networks and the systematic analysis of the applied communication algorithms by means of an interactive tutorial concept. They will assess the implementations of existing network protocols and to evaluate their performance by means of a measurement analysis with Wireshark and other tools. The processing of the design, assessment, measurement, and implementation tasks will be performed by teams of students. Thus, learning effective teamwork is part of the course.

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

prerequisites for the module:
none

Recommended prior knowledge:	**Admission requirements:** governed by examination regulations (StuFPO)

• successful examination in data communication similar to module KTR-Datkomm-B and substantial knowledge of related technical concepts • knowledge in progamming with JAVA (or C++) Module Advanced Java Programming (DSG-AJP-B) - recommended Module Data communication (KTR-Datkomm-B) - recommended	
Frequency: every summer semester	**Recommended semester:**

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Multimedia Communication in High Speed Networks **Mode of Delivery:** Lectures and Practicals **Lecturers:** Prof. Dr. Udo Krieger **Language:** English/German **Frequency:** every summer semester	**4,00 Weekly Contact Hours**
Learning outcome: The students will be enabled to work independently according to the highest scientific standards on design and analysis tasks associated with high-speed network protocols. They will learn about the fundamentals of multimedia communication in high-speed networks and the systematic analysis of the applied communication algorithms by means of an interactive tutorial concept. They will assess the implementations of existing network protocols and to evaluate their performance by means of a measurement analysis with Wireshark and other tools. The processing of the design, assessment, measurement, and implementation tasks will be performed by teams of students. Thus, learning effective teamwork is part of the course.	
Contents: Based on the foundations of data communication, this advanced course of the masters programme presents the design of high-speed networks (HSN) and the advanced protocol elements of the signaling and user plane that are required to implement new real-time and multimedia services. It includes the digital switching technologies and protocol stacks of HSNs, the quality-of-service architectures, as well as the traffic management protocols of these next generation IP networks. The extension of the TCP/IP protocol stack to realize communication relations among mobile or stationary end systems that are supported by quality-of-service guarantees and associated improved switching concepts are discussed in detail by lectures of the course. These lectures focus on effective access technologies and new transport and QoS-architectures in the core network like Diffserv, MPLS and GMPLS. Further, the enhancement of IPv4 by IPv6 switching and the extension of TCP by modern multipath concepts such as MPTCP and SCTP are presented. Advanced QoS-management concepts, effective resource and traffic management schemes like buffer management by RED, RIO or schedluing by weighted fair queueing (WFQ), are discussed, too. Furthermore, we present new architectures for next generation networks (NGNs) such as software-defined networks and information-centric networks.	

Modern multimedia service architectures with interactive applications for third to fourth generation Internet like Web applications based on HTTP 2.0, WebRTC, peer-to-peer VoIP and media streaming applications are sketched.

The content of the lectures is illustrated by exercises and laboratories covering important aspects of the protocol stacks in high-speed networks. The independent processing of tasks, the qualified presentation and critical discussion of the outcomes by teams of students is part of the course. It improves the technical understanding and provides means to work as project leader in industry on those topics.

The course can be supplemented by the module Foundations of Internet Communcation (KTR-GIK-M) with its instructive tasks executed in the router laboratory, by master seminars and projects or a master's thesis on related topics in next generation networks.

The language of the course wil be announced during the first lecture.

Literature:

- 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.

Examination
Oral examination / Duration of Examination: 30 minutes
Description:
30 minutes oral examination related to the technical topics of all lectures and practicals.

The language of the examination wil be announced during the first lecture.

Module KTR-Mobi-M Mobile Communication

Mobilkommunikation

6 ECTS / 180 h
45 h Präsenzzeit
135 h Selbststudium

(since SS20)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:

The course presents the fundamentals of mobile communication. We sketch the underlying standards, system architectures and their realizations as well as current research and development trends. Due to the complexity of the field the course can only present some basic important aspects of those mobile communication systems that exhibit the strongest growth in the markets and affect all busieness areas of the information societies at most. The course will focus on the technical system and design perspectives regarding the service architectures and local or wide area mobile communication networks.

The following topics are discussed in detail:
• technical foundation of wireless transmission
• media access control protocols
• resource management protocols in mobile communication networks (including resource assignment strategies at the radio layer, handoff management, error control protocols, scheduling etc.)

• mobility support at the network layer by mobile IP
• transport protocols and their enhancements
• wirelss LANs and their development (IEEE802.11 standards, WiMAX etc.)
• wireless wide area networks based on TDMA technology (GSM basics and protocols, GPRS)
• data communication in wireless wide area networks (UMTS, HSPA, LTE, LTE-A etc.)
• service architectures for mobile networks (including Android programming and WebRTC architectures)

Learning outcomes:

The students are encouraged to independent scientific work. They learn the fundamentals of mobile communication and are trained to analyze the applied protocols and communication algorithms in a systematic manner. Students are instructed to investigate the sketched mobile communication protocols by measurements using Wireshark and other tools, to evaluate their performance, and to develop new protocol elements. The processing of design, programming, and performance assessment tasks by teams of students and the effective arrangement of workgroups is part of the training.

Remark:

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

prerequisites for the module:

none

Recommended prior knowledge:

- substantial knowledge of the foundations of data communication similar to module KTR-Datkomm-B
- good knowledge of programming in JAVA (or C++)
- knowledge of algorithms and data structures similar to module MI-AuD-B

Module Algorithms and Data Structures (AI-AuD-B) - recommended
Module Advanced Java Programming (DSG-AJP-B) - recommended

Admission requirements:

governed by examination regulations (StuFPO)

Module Data communication (KTR-Datkomm-B) - recommended		
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Mobile Communication Course
Mode of Delivery: Lectures and Practicals
Lecturers: Prof. Dr. Udo Krieger
Language: English/German
Frequency: every winter semester

4,00 Weekly Contact Hours

Learning outcome:

The students are encouraged to independent scientific work. They learn the fundamentals of mobile communication and are trained to analyze the applied protocols and communication algorithms in a systematic manner. Students are instructed to investigate the sketched mobile communication protocols by measurements using Wireshark and other tools, to evaluate their performance, and to develop new protocol elements. The processing of design, programming, and performance assessment tasks by teams of students and the effective arrangement of workgroups is part of the training.

Contents:

The course presents the fundamentals of mobile communication. We sketch the underlying standards, system architectures and their realizations as well as current research and development trends. Due to the complexity of the field the course can only present some basic important aspects of those mobile communication systems that exhibit the strongest growth in the markets and affect all busieness areas of the information societies at most. The course will focus on the technical system and design perspectives regarding the service architectures and local or wide area mobile communication networks.

The following topics are discussed in detail:

- technical foundation of wireless transmission
- media access control protocols
- resource management protocols in mobile communication networks (including resource assignment strategies at the radio layer, handoff management, error control protocols, scheduling etc.)

- mobility support at the network layer by mobile IP
- transport protocols and their enhancements
- wirelss LANs and their development (IEEE802.11 standards, WiMAX etc.)
- wireless wide area networks based on TDMA technology (GSM basics and protocols, GPRS)
- data communication in wireless wide area networks (UMTS, HSPA, LTE, LTE-A etc.)
- service architectures for mobile networks (including Android programming and WebRTC architectures)

The content of the lectures is illustrated by exercises and laboratories covering important aspects of the protocol stacks in mobile networks. The independent processing of tasks, the qualified presentation and critical discussion of the outcomes by teams of students is part of the course. It improves the technical understanding and provides means to work as project leader in industry on those topics.

The course can be supplemented by the module Foundations of Internet Communcation (KTR-GIK-M) with its instructive tasks executed in the router laboratory, by master seminars and projects or a master's thesis on related topics in next generation networks.

The language of the course wil be announced during the first lecture.

Literature:

- Schiller, J.: Mobile Communications. Pearson-Education, Munich, 2004.
- Walke, B.: Mobile Radio Networks, Wiley, 2002.
- Pahlavan, K., Krishnamurthy, P.: Principles of Wireless Networks, A Unified Approach. Prentice Hall, 2002.
- Pahlavan, K., Krishnamurthy, P.: Networking Fundamentals: Wide, Local and Personal Area Communications, Wiley, 2009.
- Holma, H., Toskala, A.: LTE for UMTS, Evolution to LTE-Advanced, 2. ed, Wiley, 2011.

Examination

Oral examination / Duration of Examination: 30 minutes

Description:

30 minutes oral examination covering all topics of the lectures and practicals.

The language of the examination wil be announced during the first lecture.

Module KTR-Proj Project Communication Networks and Services *Projekt Kommunikationsnetze und -dienste*	6 ECTS / 180 h 40 h Präsenzzeit 140 h Selbststudium

(since WS17/18)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:

Important skills regarding the planning, development and implementation of new communication technologies, their advanced services, and the related protocols in next generation networks can only be learnt by team oriented development projects subject to stringent time and resource constraints, and clear development objectives, similar to an industrial project environment.Following these lines, the course will provide fundamental insights on the functionality of modern service architectures and communication principles of next generation Internet and its development.

After a short training phase and based on an autonomous working mode, students will learn by a teamwork project to solve advanced communication tasks and to implement new communication services associated with current research issues of the professorship. The basic network equipment, network operating systems, software and development tools will be provided in the laboratory. Handling hardware and software will be learnt by the students independently as part of the project after a guidance phase. All development steps and results will be documented by a corresponding written report and oral presentations of the results.

Actual topics will be announced on the web page of the module. The related specification of the development tasks and their milestones will be done in cooperation with the supervisor.

Learning outcomes:

The students are encouraged to a scientific working mode aftr a short guidance phase. They learn how to plan, develop and implement multimedia services and communication protocols in existing and future generation networks. They are trained to efficiently implement the applied protocols and to analyze the performance of the communication algorithms in a systematic manner. Students are instructed to investigate their developed protocol code elements by measurements and other tools, to evaluate their performance, and to develop improved protocol units. The processing of design, programming, and performance assessment tasks by teams of students and the effective arrangement of the group work is part of the training. It is the objective to gain practical experience on QoS-based multimedia communication and to develop the skills to implement and evaluate network components of modern service architectures.

The project follows scientific standards and deals with research issues of the professorship. The overall objective is to develop skills and knowledge required for a successful career in industry or research in the field of communication engineering.

Remark:

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

prerequisites for the module:

none

Recommended prior knowledge: - good programming skills in JAVA (or C++)	**Admission requirements:**

- good knowledge in data communication, similar to module KTR-Datkomm-B Module Data communication (KTR-Datkomm-B) - recommended		governed by examination regulations (StUFPO)
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Projekt Kommunikationsnetze und-dienste **Mode of Delivery:** **Lecturers:** Prof. Dr. Udo Krieger **Language:** English/German **Frequency:** every winter semester **Learning outcome:** The details are sketched previously in the module description. **Contents:** Important skills regarding the planning, development and implementation of new commmunication technologies, their advanced services, and the related protocols of communication networks can only be learnt by team oriented development projects subject to stringent time and resource constraints, and clear development objectives, similar to an industrial project environment. The students will get insight on the service and network architectures of next generation Internet. The main objective is the realization of development tasks applying accumulated knowledge on communication networks. After a short training phase and based on an autonomous working mode, students will learn by a teamwork project to solve advanced communication development tasks and to implement new communication services associated with current research issues of the professorship. The organization of the project is following the framework of industry. It comprises definition, preparation, implementation and presentation phases. An incremental processing is performed like in industrial projects. It means • a segmentation into specific work packages, • its division into tasks and subtasks including milestones • the presentation of intermediate results • a final report with presentation and an individual colloquium to defend the outcome. Research and development tasks are related to current research issues in "Future Generation Networks" and will be integrated into the module. An actual list of studied topics and related references are presented in the first lecture. **Literature:** A reference list will be provided in the first meeting of the project.	**4,00 Weekly Contact Hours**
Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:**	

Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** The results of teamwork and individual work phases which are reflected by the written project report and the associated presentations of the project results will be evaluated. The outcome must be completed within 4 months. The final assessment of the examination includes the corresponding chapters of the project report composed by the candidate and the evaluation of an individual colloquium of the candidate lasting 30 minutes. All contributions must be achieved within the same semester. A regular participation in all units of the module is required to be admitted to the final examination.	

Module KTR-SSSProj-B KTR Bachelor Project Software Systems Science *KTR Bachelorprojekt Software Systems Science*	12 ECTS / 360 h 70 h Präsenzzeit 290 h Selbststudium

(since WS18/19)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:

Wichtige Fertigkeiten bei der Anwendung neuer Kommunikationstechnologien und zur Entwicklung neuer Kommunikationsdienste sind nur durch die Vermittlung praktischer Fähigkeiten und Erfahrungen in teamorientierten Prozessen unter Zeit- und Zielvorgaben industrienah erlernbar. Die Studierenden werden in der Lehrveranstaltung in einem angeleiteten, aber ansonsten eigenverantwortlich durchgeführten teamorientierten Arbeitsprozess aktuelle Entwicklungsaufgaben aus dem Forschungsbereich der Professur für Informatik bearbeiten.

Learning outcomes:

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 kleinerer, 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.

Remark:

Dieses Modul erstreckt sich über 2 Semester und umfasst 2x6=12 ECTS und 2x4=8 SWS.

Der Arbeitsaufwand beträgt insgesamt 360 Std., welche sich grob wie folgt gliedern:

- 30 Std. Recherche, Planung und Teilnahme am Planungsworkshop
- 40 Std. Teilnahme an Projekttreffen, einschließlich Tutorien
- 180 Std. Durchführung des Projekts (Projektarbeit)
- 50 Std. Erstellung des Zwischenberichts (Hausarbeit)
- 60 Std. Erstellung des Abschlussberichts, Erstellung und Präsentation der Projektergebnisse (Hausarbeit und Kolloquium)

prerequisites for the module:

none

Recommended prior knowledge:	**Admission requirements:**

• gute Kenntnisse in Mathematik für Informatiker 2 • mindestens gute JAVA (oder C/C++) Kenntnisse • Kenntnisse der Datenkommunikation im Umfang von KTR-Datkomm-B oder vergleichbare Kenntnisse werden empfohlen • grundlegende methodische Kenntnisse zur Planung und Durchführungvon Softwareprojekten, z.B. im Umfang des Moduls "Software EngineeringLab" (SWT-SWL-B), werden empfohlen. Module Introduction to Parallel and Distributed Programming (DSG-PKS-B) - recommended Module Data communication (KTR-Datkomm-B) - recommended Module Mathematics for Computer Science 2 (Linear Algebra) (KTR-MfI-2) - recommended	none

Frequency: every semester	**Recommended semester:** 4.	**Minimal Duration of the Module:** 2 Semester

Module Units	
Bachelorprojekt Software Systems Science **Mode of Delivery:** **Lecturers:** Prof. Dr. Udo Krieger **Language:** German/English **Frequency:** every semester **Contents:** 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 2 konsekutive 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: • Unterteilung der Arbeiten in Arbeitspakete (laboratories/work packages), • ihre Untergliederung in Aufgaben (tasks) und Teilaufgaben (subtasks) mit Meilensteinen • und der Darlegung von Zwischenergebnissen in einem Zwischenbericht nach dem 1. Semester sowie • einem Abschlussbericht mit Abschlusspräsentation der Arbeitsergebnisse in einem Kolloquium im 2. Semester.	**8,00 Weekly Contact Hours**

Es werden Entwicklungsaufgaben zu aktuellen Forschungsfragen im "Future Generation Internet" bearbeitet. Details werden auf der Webseite der Lehrveranstaltung angekündigt. Eine aktuelle Liste der bearbeiteten Themen der Lehrveranstaltung wird in der 1. Besprechung bereitgestellt.

Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstalltung.

Examination
Coursework Assignment / Duration of Coursework: 4 months
prerequisites for module examination:
Regelmäßige Teilnahme an der Lehrveranstaltung
Description:
Die Lehrveranstaltung erstreckt sich über 2 konsekutive Semester. Es werden die Leistungen der als Gruppen- oder Einzelarbeit ausgeführten individuellen schriftlichen Ausarbeitung der Projektaufgaben mit einer Bearbeitungsdauer von 6 Monaten im 1. Semester bewertet.

Die Bearbeitungsdauer der Hausarbeit beträgt 6 Monate.

Die Bekanntgabe der Prüfungssprache sowie der Gewichtung der Prüfungsleistungen erfolgt in der ersten Sitzung der Lehrveranstalltung.

Examination
Coursework Assignment and Colloquium / Duration of Examination: 30 minutes
Duration of Coursework: 4 months
prerequisites for module examination:
Regelmäßige Teilnahme an der Lehrveranstaltung
Description:
Die Lehrveranstaltung erstreckt sich über 2 konsekutive Semester. Es werden die Leistungen der als Gruppen- oder Einzelarbeit ausgeführten individuellen schriftlichen Ausarbeitung der Projektaufgaben und ihrer Präsentation im 2. Semester sowie die Ergebnisse einer abschließenden, individuellen Kolloquiumssprüfung bewertet.

Die Dauer des Kolloquiums beträgt 30 Minuten. Die Bearbeitungsdauer der 2. Hausarbeit beträgt 6 Monate.

Alle Teilleistungen müssen in jedem Semester erfolgreich absolviert werden. Die Gewichtung der Prüfungsleistungen wird zu Beginn des Semesters bekannt gegeben.

Module KTR-SSSProj-M KTR Master Project Software Systems Science

KTR Masterprojekt Software Systems Science

9 ECTS / 270 h
70 h Präsenzzeit
200 h Selbststudium

(since WS17/18)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:
Important skills regarding the planning, development and implementation of new commmunication technologies, their advanced services, and the related protocols in next generation networks can only be learnt by team oriented development projects subject to stringent time and resource contraints, and clear development objectives, similar to an industrial project environment. After a short training phase and based on an autonomous working mode, students will learn by a teamwork project to solve advanced communication tasks and to implement new communication services associated with current research issues of the professorship. Actual topics will be announced on the web page of the module.

Learning outcomes:
The students are encouraged to independent scientific work. They learn how to plan, develop and implement new advanced multimedia services and communication protocols in next generation networks. They are trained to efficiently implement the applied protocols and to analyze the performance of the communication algorithms in a systematic manner. Students are instructed to investigate their developed protocol code elements by measurements and other tools, to evaluate their performance, and to develop improved protocol units. The processing of design, programming, and performance assessment tasks by teams of students and the effective arrangement of the groupwork is part of the training.

The project follows scientific standards and deals with research issues of the professorship. The overall objective is to develop skills and knowledge required for a successful career in industry or research in the field of communication engineering.

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

prerequisites for the module:
A bachelor degree in computer science, computer engineering or mathematics is required. Students must be enroled in the masters degree programme "M.Sc. International Software Systems Science".

Recommended prior knowledge:

- good knowledge in mathematics and statistics, similar to module Mathematik für Informatiker 2
- good programming skills in JAVA (or C++)
- good knowledge in data communication, similar to module KTR-Datkomm-B
- solid methodological know-how in planning and execution of software projects, similar to the module "Software Engineering Lab" (SWT-SWL-B)

Module Introduction to Parallel and Distributed Programming (DSG-PKS-B) - recommended
Module Data communication (KTR-Datkomm-B) - recommended

Admission requirements:
governed by examination regulations (StuFPO)

Module Mathematics for Computer Science 2 (Linear Algebra) (KTR-MfI-2) - recommended Module Software Engineering Lab (SWT-SWL-B) - recommended		
Frequency: every semester	**Recommended semester:** 2.	**Minimal Duration of the Module:** 1 Semester

Module Units

KTR Master Project Software Systems Science
Mode of Delivery:
Lecturers: Prof. Dr. Udo Krieger
Language: English/German
Frequency: every semester

6,00 Weekly Contact Hours

Learning outcome:
The details are sketched previously.

Contents:
Important skills regarding the planning, development and implementation of new commmunication technologies, their advanced services, and the related protocols in next generation networks can only be learnt by team oriented development projects subject to stringent time and resource contraints, and clear development objectives, similar to an industrial project environment. After a short training phase and based on an autonomous working mode, students will learn by a teamwork project to solve advanced communication tasks and to implement new communication services associated with current research issues of the professorship.

The organization of the project is following the framework of industry. It comprises definition, preparation, implementation and presentation phases. An incremental processing is performed like in industrial projects. It means

- a segmentation into specific work packages,
- its division into tasks and subtasks including milestones
- the presentation of intermediate results
- a final report with presentation and an individual colloquium to defend the outcome.

Research and development tasks are related to current research issues in "Future Generation Internet" and will be integrated into the module. An actual list of studied topics and related references are presented in the first lecture.

The language of the course wil be announced during the first lecture.

Literature:
A reference list will be provided in the first meeting of the project.

Examination
Coursework Assignment and Colloquium / Duration of Examination: 30 minutes
Duration of Coursework: 4 months
prerequisites for module examination:
Regelmäßige Teilnahme an der Lehrveranstaltung

Description: The course duration is one semester. The assessment of the module covers the results of the project report, written either as groupwork or on an individual basis by the student, the project presentation, and the final colloquium arranged on an individual basis. The language of the course and its examination is announced during the first lecture.	

Module KTR-Sem-B Bachelor Seminar Communication Systems and Computer Networks *Bachelorseminar zu Kommunikationssystemen und Rechnernetzen*	3 ECTS / 90 h 20 h Präsenzzeit 70 h Selbststudium

(since SS20)
Person responsible for module: Prof. Dr. Udo Krieger

Contents:

Die Studierenden lernen, aktuelle technische Fragestellungen aus den Themenbereichen der Komunikationsnetze und -dienste der neuesten Generation anhand der Fachliteratur unter Anleitung wissenschaftlich zu bearbeiten und das erworbene Wissen in systematischer Form schriftlich und mündlich darzulegen.

Learning outcomes:

Die Studierenden lernen, aktuelle technische Fragestellungen aus dem Themenfeld der Komunikationsnetze und -dienste anhand der Fachliteratur unter Anleitung wissenschaftlich zu bearbeiten und das erworbene Wissen in systematischer Form schriftlich und mündlich darzulegen. Die Fähigkeit zur kritischen Bewertung komplexer technischer Inhalte nach wissenschaftlichen Grundsätzen der Informatik stelllt ein wichtiges Lernziel dar.

Remark:

Der Arbeitsaufwand gliedert sich grob wie folgt:

- Präsenzveranstaltungen inkl. Themenvergabe und Besprechungen mit dem Betreuer: 20 Stunden
- Bearbeitung des Fachthemas und schriftliche Darstellung: 54 Stunden
- Erarbeitung der Präsentation: 16 Stunden

prerequisites for the module:

none

Module Data communication (KTR-Datkomm-B) - Pflicht

Recommended prior knowledge: Module gemäß der Spezifikationen des Pflichtbereichs sowie solide Kenntnisse der Datenkommunikation		**Admission requirements:**
Frequency: winter and summer semester, on demand	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Bachelorseminar KTR-Bachelor — **2,00 Weekly Contact Hours**

Mode of Delivery: Seminar
Lecturers: Prof. Dr. Udo Krieger
Language: German/English
Frequency: winter and summer semester, on demand

Learning outcome:

Die Studierenden lernen, aktuelle Fragestellungen aus dem Themenfeld der Komunikationsnetze und -dienste anhand der Fachliteratur unter Anleitung wissenschaftlich zu bearbeiten und das erworbene Wissen in systematischer Form schriftlich und mündlich darzulegen.

Contents:

Es werden aktuelle Fragestellungen aus dem Bereich der Kommunikationstechnik und Rechnernetze unter Anleitung bearbeitet . Die aktuelle Themenliste wird auf der Webseite bereitgestellt.

Die Bekanntgabe der Lehrsprache erfolgt in der ersten Sitzung der Lehrveranstalltung.

Die schriftliche Ausarbeitung erfolgt in LATEX, die mündliche Darstellung im Rahmen einer Powerpoint-, LATEX-Beamer oder PDF-Präsentation auf Basis der schriftlichen Ausarbeitung in möglichst freier Rede und logisch korrekter, verständlicher Form.

Literature:
Die aktuelle Literaturliste wird bei der Vorbesprechung bereitgestellt.

Examination
Coursework Assignment with presentation / Duration of Examination: 30 minutes
Duration of Coursework: 4 months

prerequisites for module examination:
Regelmäßige Teilnahme an der Lehrveranstaltung

Description:
Die Gesamtnote ergibt sich zu gleichen Teilen aus der Bewertung der schriftlichen Ausarbeitung (mit Bearbeitungsdauer von maximal 4 Monaten) und des Referats und muss mit mindestens ausreichend bewertet sein.

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

Module KTR-Sem-M Master Seminar Communication Systems and Computer Networks *Hauptseminar zu Kommunikationssystemen und Rechnernetzen*		3 ECTS / 90 h
(since SS20) Person responsible for module: Prof. Dr. Udo Krieger		
Contents: The seminar will discuss hot topics in the fields of stationary and mobile communication networks, new Internet services as well as fog and cloud computing architectures. The development of powerful transport and edge computing platforms for future generation software-defined networks supporting quality-of-service and mobility requirements will constitute a technical focus of the seminar.		
Learning outcomes: A major competence objective is given by the ability to evaluate the scientific literature in a critical manner and to apply new scientific results while solving a technical problem at hand. We shall improve the ability to adopt effectively the new technical methodologies stemming from the fields of software-defined communication networks, the theory of distributed systems, and the foundations of computer science.		
Remark: The workload comprises the following components: • personal presence phases including topic dissemination and discussions witht the lecturers: 20 hours • preparation of the technical topic and writing of the report: 54 hours • preparation of the oral presentation: 16 hours		
prerequisites for the module: • knowledge on topics of the module Foundations of Intenet Communication (KTR-GIK-M) Module Foundations of Internet Communication (KTR-GIK-M) - Pflicht		
Recommended prior knowledge: • basic knowledge on the principles of data communication • additional knowledge according to the technical specification of the offered seminar Module Data communication (KTR-Datkomm-B) - recommended		**Admission requirements:** none
Frequency: winter or summer semester, on demand	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Seminar KTR-Master **Mode of Delivery:** Advanced seminar **Lecturers:** Prof. Dr. Udo Krieger **Language:** English/German **Frequency:** winter and summer semester, on demand	**2,00 Weekly Contact Hours**
Learning outcome: The students will prepare the writing of a master's thesis and their industrial or scientific employment. A major competence objective is given by the ability to evaluate the scientific literature in a critical manner and to apply new scientific results while solving a technical problem at hand.	

Contents:
The seminar will discuss hot topics in the fields of stationary and mobile communication networks, new Internet services as well as fog and cloud computing architectures. The development of powerful transport and edge computing platforms for future generation software-defined networks supporting quality-of-service and mobility requirements will constitute the technical focus of the seminar.

The seminar offers a student the perspectives on the system-theoretical foundations of actual technical topics arising in the rapidly evolving areas of modern communication and fog/cloud computing systems. It is the objective of study to independently adopt the new technical methodologies stemming from the fields of software-defined communication networks, the theory of distributed systems, and the foundations of computer science.

Passing the examination of the seminar is, in general, a prerequisite to successfully write a master's thesis at the Professorship of Computer Science or in cooperation with industrial peers.

The used language of the module will be announced during the first session of the seminar.

Literature:
The relevant reference list will be announced during the first session.

Examination
Coursework Assignment with presentation / Duration of Examination: 40 minutes
Duration of Coursework: 4 months
prerequisites for module examination:
Regelmäßige Teilnahme an der Lehrveranstaltung
Description:
The final grade evaluates the written report (- this phase lasts at most 4 months -) and the oral presentation as equally weighted components. Both the report and oral presentation have to achieved at least the grade 4.0 to pass the examination.

The language of the examination will be announced during the first session of the seminar.

Module KogSys-ML-M Machine Learning *Lernende Systeme (Machine Learning)*	6 ECTS / 180 h
(since SS21) Person responsible for module: Prof. Dr. Ute Schmid	

Contents:

The course introduces the field of machine learning and provides a broad overview of symbolic, neural and statistical approaches to machine learning, their mathematical foundations and their algorithmic realisation.

Learning outcomes:

The students will be able to:

- explain and apply basic concepts and standard approaches of machine learning
- apply central symbolic, neural and statistical algorithms of classification learning to given data
- assess the suitability of given data for classification learning algorithms
- evaluate the quality of learned models
- discuss similarities and differences between human and machine learning

Remark:

Teaching language German (if necessary English). The slides and other materials are mainly available in English language.

The workload for this module is roughly divided as follows:
22.5 h lecture + 30 h follow-up over 15 weeks
22.5 h exercise + 75 h working on exercises over 15 weeks
30 h exam preparation

prerequisites for the module:

none

Recommended prior knowledge: Module Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik) (GdI-MfI-1). Module Mathematik für Informatik 2 (lineare Algebra) (KTR-MfI-2-B). Module Algorithmen und Datenstrukturen (AI-AuD-B).		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Machine Learning **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Ute Schmid **Language:** German/English **Frequency:** every winter semester **Learning outcome:** see above **Contents:** The lecture introduces essential symbolic, statistical and neural approaches to machine learning, in particular decision tree algorithms, artificial neural networks,	**2,00 Weekly Contact Hours**

instance-based learning, inductive logical programming, genetic algorithms, bayesian learning, kernel methods, support vector machines, reinforcement learning. References to human learning and current issues such as transparency and explainability are established.

The teaching language will be announced in the first lecture.

Literature:
Mitchell, Machine Learning, McGraw-Hill, 1997.
Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data, 2012.
Goodfellow et al., Deep Learning, MIT Press, 2016.
Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

2. Practices Machine Learning

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Lecturers: Scientific Staff Angewandte Informatik, insb. Kognitive Systeme
Language: German/English
Frequency: every winter semester

Learning outcome:
see above

Contents:
Consolidation of methods and techniques introduced in the lecture, partly with programming tasks based on Python machine learning libraries.

The teaching language will be announced in the first tutorial.

Literature:
see lecture

Examination
Written examination / Duration of Examination: 105 minutes
Description:
The duration of the examination includes a reading time of 15 minutes to select the tasks within the given options.

In the exam 90 points can be scored. The exam is passed when at least 40 percent are reached.

During the semester, voluntary course work (exercise sheets) are issued. Through the voluntary work on these sheets, points can be awarded for getting better grades, which can be credited to the exam, provided that the exam is also passed without the points from these optional sheets.

At the beginning of the course, it will be announced:

- Type and number of the sheets
- Scope (number of achievable points) of the sheets
- Working time of the sheets

An evaluation of 1.0 can also be achieved without points from the sheets.

Permitted aids: Handwritten and printed materials, calculators without full alphanumeric keyboard and graphic display.

The examination language will be announced in the first lecture.	

Module MOBI-ADM-M Advanced Data Management *Advanced Data Management*		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(since SS21) Person responsible for module: Prof. Dr. Daniela Nicklas		
Contents: With the rapid growth of the internet and more and more observable processes, many data sets became so large that they cannot be processed with traditional database methods any more. This modul covers advanced data management and integration techniques (also known under the term “big data”) that are useful when dealing with very large data sets.		
Learning outcomes: The students will understand the challenges of big data, and will be able to apply some of the new techniques to deal with it.		
Remark: The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German. The written reports/seminar essay and the presentation may be delivered in English or in German.		
prerequisites for the module: none		
Recommended prior knowledge: Foundations of relational databases, relational algebra and SQL; e.g. from Modul SEDA-DMS-B: Data management systems		**Admission requirements:** none
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Advanced Data Management **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Daniela Nicklas **Language:** English **Frequency:** every summer semester **Contents:** The lecture will cover various algorithms for clustering, association rule mining, or page ranking and their scalable processing using map and reduce methods, data integration, data cleansing and entity recognition. The exercises will be built upon the Hadoop framework. The language of the course will be announced in the first lecture. **Literature:** L. Wiese, Advanced Data Management, For SQL, NoSQL, Cloud and Distributed Databases. Berlin, Boston: De Gruyter, 2015	**2,00 Weekly Contact Hours**
2. Practicals Advanced Data Management **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Daniela Nicklas	**2,00 Weekly Contact Hours**

Language: English

Frequency: every summer semester

Contents:

see Lectures

The language of the course will be announced in the first lecture.

Examination

Written examination / Duration of Examination: 75 minutes

Description:

Central written exam. The examination language is English.

The exam questions will be in English. The questions can be answered in English or German. The content that is relevant for the exam consists of the content presented in the lecture and in the practical assignments.

The exam consists of 7 tasks of which only 6 will be graded. The exam time includes a reading time of 15 minutes to select the tasks to be completed within the scope of the choices.

Participants who submit solutions for practical assignments can achieve bonus points. Details regarding the number of assignments, the number of bonus points per assignment, the conversion factor from bonus points to exam points (e.g., 10:1) and the type of assignments will be announced in the first practical assignment session.

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% of the points have been obtained), the converted bonus points will be added to the points achieved in the exam.

The grade 1.0 can be achieved without the bonus points.

<table>
<tr><td colspan="2"><b>Module MOBI-DSC-M Data Streams and Complex Event Processing</b>
<i>Data Streams and Complex Event Processing</i></td><td>6 ECTS / 180 h
45 h Präsenzzeit
135 h Selbststudium</td></tr>
<tr><td colspan="3">(since WS20/21)
Person responsible for module: Prof. Dr. Daniela Nicklas</td></tr>
<tr><td colspan="3"><b>Contents:</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</td></tr>
<tr><td colspan="3"><b>Learning outcomes:</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</td></tr>
<tr><td colspan="3"><b>prerequisites for the module:</b>
none</td></tr>
<tr><td colspan="2"><b>Recommended prior knowledge:</b>
Foundations of relational databases, relational algebra and SQL; e.g. from Modul MOBI-DBS-B: Database Systems</td><td><b>Admission requirements:</b>
none</td></tr>
<tr><td><b>Frequency:</b> every winter semester</td><td><b>Recommended semester:</b></td><td><b>Minimal Duration of the Module:</b>
1 Semester</td></tr>
</table>

<table>
<tr><td colspan="2"><b>Module Units</b></td></tr>
<tr><td><b>Data Streams and Complex Event Processing</b>
<b>Mode of Delivery:</b> Lectures
<b>Lecturers:</b> Prof. Dr. Daniela Nicklas
<b>Language:</b> English
<b>Frequency:</b> every winter semester
<b>Learning outcome:</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</td><td><b>2,00 Weekly Contact Hours</b></td></tr>
</table>

Understand the main security challenges and solutions in data stream management systems	
Contents: 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	
Examination Oral examination / Duration of Examination: 15 minutes **Description:** oral or written exam (will be announced in class at the beginning of the semester). The examination language is English.	

Module Units	
Data Streams and Complex Event Processing **Mode of Delivery:** Practicals **Language:** English **Frequency:** every winter semester	**2,00 Weekly Contact Hours**
Contents: see lecture	

Examination Written examination / Duration of Examination: 60 minutes **Description:** oral or written exam (will be announced in class at the beginning of the semester). The examination language is English.	

Module MOBI-MSS-B Mobility in Software Systems *Mobility in Software Systems*		6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium
(since SS21) Person responsible for module: Prof. Dr. Daniela Nicklas		
Contents: This modul covers architectures, implementation techniques and algorithms for mobile software systems and software systems that manage mobility. This includes client-side aspects (mobile applications like location-based services), server-side aspects (data management of moving objects), and aspects of distribution (data communication). In addition, since many mobile software systems deal with sensitive information like the location of users, aspects of location privacy are covered.		
Learning outcomes: The students will understand the challenges of mobility in software systems, and will be able to apply techniques and methods to realize such systems.		
Remark: The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German. The written reports/seminar essay and the presentation may be delivered in English or in German.		
prerequisites for the module: none		
Recommended prior knowledge: Basic knowledge about relational databases, relational algebras and SQL (e.g. from module SEDA-DMS-B: Datenmanagementsysteme)		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Mobility in Software Systems **Mode of Delivery:** Lectures and Practicals **Language:** English **Frequency:** every winter semester **Learning outcome:** The students will understand the challenges of mobility in software systems, and will be able to apply techniques and methods to realize such systems. **Contents:** The language of the course will be announced in the first lecture.	**4,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 105 minutes **Description:** The language of the exam will be announced in the first lecture.	

Module MOBI-PRAI-B Bachelor Project Mobile Software Systems (AI) *Bachelor Project Mobile Software Systems (AI)*	6 ECTS / 180 h

(since WS19/20)
Person responsible for module: Prof. Dr. Daniela Nicklas

Contents:

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 Data streams and event processing.

The tasks in the project will be tailored to Bachelor level.

Learning outcomes:

Studierende sollen ein vertieftes Verständnis der bei der Durchführung von praktischen, arbeitsteilig organisierten, Softwareprojekten auftretenden Probleme wie auch von erfolgversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich der praktischen Informatik geschieht, gewinnen die TeilnehmerInnen wichtige Erfahrungen mit der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in einem wissenschaftlich ausgerichteten Arbeitsbericht.

Remark:

The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German.

The written reports/seminar essay and the presentation may be delivered in English or in German.

prerequisites for the module:

none

Recommended prior knowledge: Programmierkenntnisse sowie grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z. B. erworben im Modul "Software Engineering Lab" (SWT-SWL-B), und zum wissenschaftlichen Arbeiten, z. B. erworben im Modul "Wissenschaftliches Arbeiten in der Informatik" (IAIWAI-B).	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Bachelor project Mobile Software Systems (AI) **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Daniela Nicklas	**4,00 Weekly Contact Hours**

Language: English/German **Frequency:** every summer semester **Contents:** Projektdurchführung	
Examination Coursework Assignment and Colloquium **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Prüfung Hausarbeit mit Kolloquium 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).	

Module MOBI-PRAI-M Master Project Mobile Software Systems (AI) *Master Project Mobile Software Systems (AI)*	6 ECTS / 180 h

(since WS19/20 to SS21)
Person responsible for module: Prof. Dr. Daniela Nicklas

Contents:

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 Data streams and event processing.

The tasks in the project will be tailored to Master level.

Learning outcomes:

Students will deepen their knowledge regarding the conceptual problems that arise when carrying out theoretical and/or practical research on 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.

Remark:

The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German.

The written reports/seminar essay and the presentation may be delivered in English or in German.

prerequisites for the module:

none

Recommended prior knowledge: Foundations of relational databases, relational algebra and SQL; e.g. from Modul SEDA-DMS-B: Data management systems	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Master project Mobile Software Systems (AI) **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Daniela Nicklas **Language:** English/German **Frequency:** every winter semester **Contents:** The language of the course will be announced in the first lecture.	**4,00 Weekly Contact Hours**

Examination Coursework Assignment and Colloquium **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** The language of the exam will be announced in the first lecture.	

Module MOBI-PRS-B Bachelor Project Mobile Software Systems (SoSySc) *Bachelor Project Mobile Software Systems (SoSySc)*	12 ECTS / 360 h

(since WS19/20)
Person responsible for module: Prof. Dr. Daniela Nicklas

Contents:
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 Data streams and event processing.

The tasks in the project will be tailored to Bachelor level.

Learning outcomes:
Studierende sollen ein vertieftes Verständnis der bei der Durchführung von Softwaresystem-Projekten auftretenden konzeptionellen und praktischen Probleme wie auch von erfolgsversprechenden Lösungsansätzen zu diesen Problemen erhalten. Da dies anhand der intensiven Bearbeitung eines Themas aus dem Forschungsbereich von mobilen Softwaresystemen in Kleingruppen – oder ggf. auch einzeln – geschieht, gewinnen die Studierenden wichtige Erfahrungen in der Durchführung kleinerer, forschungsorientierter Projekte von der Grobkonzeption über die Detailplanung bis hin zur Umsetzung und Dokumentation der Ergebnisse in wissenschaftlich ausgerichteten Arbeitsberichten und in der Präsentation dieser Ergebnisse.

Remark:
The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German.

The written reports/seminar essay and the presentation may be delivered in English or in German.

prerequisites for the module:
none

Recommended prior knowledge: Programmierkenntnisse sowie grundlegende methodische Kenntnisse zur Planung und Durchführung von Softwareprojekten, z. B. erworben im Modul "Software Engineering Lab" (SWT-SWL-B), und zum wissenschaftlichen Arbeiten, z. B. erworben im Modul "Wissenschaftliches Arbeiten in der Informatik" (IAIWAI-B).	**Admission requirements:** none

Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 2 Semester

Module Units	
Bachelor project Mobile Software Systems (SoSySc) **Mode of Delivery:** Practicals	**8,00 Weekly Contact Hours**

Lecturers: Prof. Dr. Daniela Nicklas **Language:** English/German **Frequency:** every semester	
Contents: The language of the course will be announced in the first lecture.	

Examination Coursework Assignment and Colloquium **prerequisites for module examination:** regelmäßige Teilnahme an der Lehrveranstaltung **Description:** The language of the exam will be announced in the first lecture.	

Module MOBI-PRS-M Master Project Mobile Software Systems (SoSySc) *Master Project Mobile Software Systems (SoSySc)*	9 ECTS / 270 h

(since WS20/21)
Person responsible for module: Prof. Dr. Daniela Nicklas

Contents:
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.

Learning outcomes:
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.

Remark:
The main language of instruction in this course is English. However, the lectures and/or tutorials may be delivered in German if all participating students are fluent in German.

The written reports/seminar essay and the presentation may be delivered in English or in German.

prerequisites for the module:
none

Recommended prior knowledge: Basic programming skills (Java or Python); scientific writing skills, e.g., obtained from the course SSS-SRW-M or from a scientific Bachelor thesis; basic knowledge in Mobile Computing as offered, e.g., by the course MOBI-MSS-B. Dependent on the topic of the specific project, additional knowledge as discussed in the courses MOBI-DSC-M or MOBI-ADM-M can be required.	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Master Project Mobile Software Systems (SoSySc) **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Daniela Nicklas	**6,00 Weekly Contact Hours**

Language: English/German
Frequency: every summer semester

Contents:

Conduct of the project, accompanied by regular meetings between students and lecturer.

The language of the course will be announced in the first lecture.

Examination

Coursework Assignment and Colloquium

prerequisites for module examination:

Regelmäßige Teilnahme an der Lehrveranstaltung

Description:

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.

Module PSI-AdvaSP-M Advanced Security and Privacy *Advanced Security and Privacy*	6 ECTS / 180 h 45 h Präsenzzeit 135 h Selbststudium

(since WS20/21)
Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:

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.

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).

Learning outcomes:

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.

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.

Remark:

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.

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.

Workload breakdown:

- Lecture: 22.5 hours (2 hours per week)
- Tutorials: 22.5 hours (2 hours per week)
- Preparation and studying during the semester: 30 hours
- Assignments: 67.5 hours
- Preparation for the exam (including the exam itself): 37.5 hours

prerequisites for the module:

none

Recommended prior knowledge: 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).	**Admission requirements:** none

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. Module Web Technologies (MI-WebT-B) - recommended Module Introduction to Security and Privacy (PSI-IntroSP-B) - recommended		
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Advanced Security and Privacy **Mode of Delivery:** Lectures **Language:** English/German **Frequency:** every summer semester	**2,00 Weekly Contact Hours**
Learning outcome: cf. module description	
Contents: Selected topics: • 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. • Attacks on privacy in datasets and communications (inference techniques, online tracking) • Privacy engineering and privacy enhancing technologies (e.g., Tor) • Usable security and privacy • Other current topics in privacy and security Some parts of the lecture are aligned with current events and recently published research. The selected topics are therefore subject to change.	
Literature: Selected books: • R. Anderson: Security Engineering • A. Shostack: Threat Modelling • J.-P. Aumasson: Serious Cryptography • W. Stallings: Computer Security: Principles and Practice • B. Schneier et al.: Cryptography Engineering • J. Erickson: Hacking: The Art of Exploitation • J. Katz & Y. Lindell: Introduction to Modern Cryptography • L. Cranor & S. Garfinkel: Security and Usability	
2. Tutorials for Advanced Security and Privacy **Mode of Delivery:** Practicals	**2,00 Weekly Contact Hours**

Language: English/German
Frequency: every summer semester

Examination
Written examination / Duration of Examination: 90 minutes
Description:
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.

Module PSI-EDS-B Ethics for the Digital Society *Ethics for the Digital Society*	3 ECTS / 90 h

(since SS20)
Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:
This module introduces students to fundamental concepts of ethics and their application to techniques that shape the digital society. It discusses the influence of current and upcoming technologies and their implications from an ethical perspective. The lecture is accompanied by a series of case studies, which focus on a concrete problem that is to be analyzed by the participants. Topics include decision making in autonomous systems and systems that employ so-called artificial intelligence, the reliability and dependability of computer systems, and privacy aspects of information systems.

Learning outcomes:
Participants will be able to reflect on their actions as a scientist as well as a computer professional. They learn how to evaluate the trade-offs that are inherent in new technologies and how to design information systems in ways that support the needs of a digital society. Successful participants will obtain the ability to apply ethical thinking to novel problems and potential solutions.

Remark:
The module is taught in English unless all participants are fluent in Germen. There may be a small number of guest lectures that is taught in German.

During the semester multiple case studies will be published. Participants will be asked to submit essays or solutions (small programs) discussing ethical aspects of those case studies. Essays will be peer-reviewed by other participants.

prerequisites for the module:
keine

Recommended prior knowledge: keine	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Ethics for the Digital Society **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Dominik Herrmann **Language:** English/German **Frequency:** every winter semester	**2,00 Weekly Contact Hours**

Learning outcome:
cf. module description

Contents:
cf. module description

Literature:

- Ibo van de Poel and Lamber Royakkers: Ethics, Technology, and Engineering – an Introduction
- Jay Quinn: Ethics for the Information Age

• Herman T. Tavani: Ethics and Technology: Controversies, Questions, and Strategies for Ethical Computing	

Examination Written examination / Duration of Examination: 60 minutes **Description:** The exam questions will be in English. The questions can be answered in English or German. The content that is relevant for the exam consists of the content presented in the lecture and in the case studies. The maximum number of points that can be achieved in the exam is 100. Participants that submit all case study essays can collect up to 10 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.	

Module PSI-IntroSP-B Introduction to Security and Privacy

Introduction to Security and Privacy

6 ECTS / 180 h

(since WS20/21)
Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:
This module introduces students to fundamental concepts in the fields of information security and the protection of privacy. It provides a broad overview over the most relevant topics from a technical perspective. The focus lies on practical issues that have to be considered when professional and personal information systems are built and operated.

Learning outcomes:
Successful students will know the mathematical background behind basic cryptographic primitives and be able to explain fundamental concepts of information security and privacy, including classical attacks and defenses. They will be able to apply their knowledge when implementing simple attack programs as well as building and operating defensive teczniques.

Remark:
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.

Workload breakdown:

- Lecture: 22.5 hours (2 hours per week)
- Tutorials: 22.5 hours (2 hours per week)
- Preparation and studying during the semester: 30 hours
- Assignments: 67.5 hours
- Preparation for the exam (including the exam itself): 37.5 hours

prerequisites for the module:
none

Recommended prior knowledge:
It is strongly recommended to take this module only after successful completion of PSI-EiRBS-B, which lays the foundation for PSI-IntroSP-B, i.e., prospective PSI-IntroSP-B participants should be familiar with fundamentals of computer architecture (binary representation of strings and numbers in computers, bitwise operators (such as XOR), operation of a CPU, basics of assembly language), operating systems (memory layout and process management), and computer networks (basic IP routing and addressing, TCP/IP connection establishment). Finally, basic familiarity with the Linux command line is recommended.

Moreover, basic familiarity with common web technologies (HTTP, HTML, JavaScript) as well as relational database systems and SQL is a recommended prerequisite. Finally, participants should have working knowledge in at least one programming language (e.g., Python, C, or

Admission requirements:
none

Java) so that they can write small tools for automation purposes on demand.		
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Introduction to Security and Privacy **Mode of Delivery:** Lectures **Language:** English **Frequency:** every winter semester **Learning outcome:** cf. module description **Contents:** Selected topics • Security Terminology (protection goals, attacker and attack types) • Authentication and Authorization Fundamentals • Software Security in C and Assembly (e.g., buffer overflows, selected defenses) • Cryptography (e.g., historic ciphers, symmetric and asymmetric cryptosystems, Diffie-Hellman key exchange, TLS protocol) • Network Security (spoofing, denial of service, authentication protocols, intrusion detection systems) • Web Security (attacks and defenses related to the OWASP Top 10 including SQL injections and Cross Site Scripting) • Privacy and Techniques for Data Protection (re-identification risks, anonymization networks, k-anonymity, the idea of differential privacy) **Literature:** Selected books: • A. Shostack: Threat Modelling • W. Stallings: Computer Security: Principles and Practice • J. Erickson: Hacking: The Art of Exploitation	**2,00 Weekly Contact Hours**
2. Introduction to Security and Privacy **Mode of Delivery:** Practicals **Language:** English **Frequency:** every winter semester	**2,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 90 minutes **Description:** The content that is relevant for the exam consists of the content presented in the lecture and tutorials. The exam questions are in English. The exam questions can be answered in English or German.	

Module PSI-ProjectCAD-M Project Complex Attacks and Defenses

Project Complex Attacks and Defenses

9 ECTS / 270 h

(since SS18)

Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:

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.

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.

This project is offered together with PSI-ProjectPAD, which focuses on conceptually simpler attacks and defenses.

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.

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.

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.

Learning outcomes:

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.

Remark:

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:

- 20 hrs: Getting familiar with the platform
- 50 hrs: Reading papers and researching security vulnerabilities
- 20 hrs: Preparing the talk (including time for attendance of other talks)
- 90 hrs: Implementing the vulnerable service and defenses
- 90 hrs: Writing training material and documentation

Note that there is another project (PSI-ProjectPAD) with a workload equivalent to 180 hours.

prerequisites for the module: none		
Recommended prior knowledge: This project is primarily intended for students in master programs. Students in bachelor programs can participate, if they are qualified. 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. 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.		**Admission requirements:** none
Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Project Complex Attacks and Defenses **Mode of Delivery:** Practicals **Language:** English/German **Frequency:** every semester **Learning outcome:** cf. module description **Contents:** Potential topics include: • web security (injection flaws and other issues mentioned in the OWASP Top 10) • network security (such as DNS cache poisoning and rebinding attacks) • security issues in C programs (buffer overflows, etc.) • cryptography (low-level attacks on ciphers, high-level attacks on protocols, e.g., TLS) • business logic failures • misconfigurations • attacks on availability (denial of service) • attacks on privacy (such as inference, tracking, re-identification, fingerprinting) • privacy defenses (such as k-anonymity, related concepts, differential privacy) **Literature:** Literature will be announced at the beginning of the project.	**6,00 Weekly Contact Hours**

Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 3 months **prerequisites for module examination:** Regular attendance at project meetings. **Description:** 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.	

Module PSI-ProjectPAD Project Practical Attacks and Defenses

Project Practical Attacks and Defenses

6 ECTS / 180 h

(since SS18)
Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:

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.

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.

This project is offered together with PSI-ProjectCAD-M, which focuses on conceptually more complex attacks and defenses.

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.

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.

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.

Learning outcomes:

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.

Remark:

This project is taught in English, unless all participants are fluent in German. The workload of this project is equivalent to 180 hours.

Workload breakdown:

- 10 hrs: Getting familiar with the platform
- 30 hrs: Reading papers and researching security vulnerabilities
- 15 hrs: Preparing the talk (including time for attendance of other talks)
- 70 hrs: Implementing the vulnerable service and defenses
- 55 hrs: Writing training material and documentation

Note that there is another project (PSI-ProjectCAD-M) with a workload equivalent to 270 hours.

prerequisites for the module: none		
Recommended prior knowledge: Students in bachelor and master programs can participate in this project. 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. 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.		**Admission requirements:** none
Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Project Practical Attacks and Defenses **Mode of Delivery:** Practicals **Language:** English/German **Frequency:** every semester	**4,00 Weekly Contact Hours**
Learning outcome: cf. module description	
Contents: Potential topics include: • web security (injection flaws and other issues mentioned in the OWASP Top 10) • network security (such as DNS cache poisoning and rebinding attacks) • security issues in C programs (buffer overflows, etc.) • cryptography (low-level attacks on ciphers, high-level attacks on protocols, e.g., TLS) • business logic failures • misconfigurations • attacks on availability (denial of service) • attacks on privacy (such as inference, tracking, re-identification, fingerprinting) • privacy defenses (such as k-anonymity, related concepts, differential privacy)	
Literature: Literature will be announced at the beginning of the project.	

Examination

Coursework Assignment and Colloquium / Duration of Examination: 30 minutes
Duration of Coursework: 3 months

prerequisites for module examination:

Regular attendance at project meetings.

Description:

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.

Module PSI-ProjectSP-M Project Security and Privacy *Project Security and Privacy*	6 ECTS / 180 h

(since SS21)
Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:
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.

Learning outcomes:
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.

Remark:
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:

- 60 hrs: Getting familiar with the problem and preliminaries: reading related work, and understanding potentially existing source code
- 20 hrs: Preparing the talk (including time for attendance of other talks)
- 110 hrs: Implementing tools and/or analyzing data
- 80 hrs: Writing final report with approach and methods

prerequisites for the module:
none

Recommended prior knowledge: 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.	**Admission requirements:** none

Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Project Security and Privacy **Mode of Delivery:** Practicals **Language:** English/German	**6,00 Weekly Contact Hours**

Frequency: every semester

Learning outcome:

cf. module description

Contents:

Potential topics include

- empirical studies, either manually (surveying security properties of systems) or automatically (e.g., web crawls),
- creating scanning tools and platforms where results can be published in a meaningful way (e.g., PrivacyScore.org),
- analyzing data sets for aspects of security and privacy, and
- implementing cryptographic or anonymization techniques in a secure fashion, e.g., for encrypted storage in cloud services.

Literature:

Literature will be announced at the beginning of the project.

Examination

Coursework Assignment and Colloquium / Duration of Examination: 30 minutes
Duration of Coursework: 3 months

prerequisites for module examination:

Regular attendance at project meetings.

Description:

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.

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.

Module PSI-SSSProject-B Software Systems Science Project: Security and Privacy *Software Systems Science Project: Security and Privacy*	12 ECTS / 360 h

(since WS18/19)
Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:

This project is specifically offered for Software Systems Science students. The participants of the project familiarize themselves with security and privacy issues that arise during the development of information systems.

Potential tasks during the project include the development of training scenarios for the Insekta platform, designing and/or participating in "build it – break it – fix it" challenges, and contributing to ongoing research activities of members of the Privacy and Security in Information Systems Group. Typically, participants work on their project in small groups. They carry out required research (mostly) on their own, reading about attacks and defenses in textbooks and research papers. Instructors will provide extensive and on-demand support to enable the participants.

Students who are interested in this project may approach a member of the PSI group in order to learn about currently available topics.

Learning outcomes:

Successful students will be able to explain attacks and defenses from textbooks and research papers. They will also be able to carry out selected attacks in practice and implement defenses with a programming language of their choice.

Remark:

This project is taught in English, unless all participants are fluent in German. The workload of this project is equivalent to 360 hours (spread over two semesters).

Workload breakdown:

- 20 hrs: Getting familiar with the task, obtaining preliminary knowledge
- 60 hrs: Reading papers and researching security vulnerabilities
- 30 hrs: Preparing the talks (including time for attendance of other talks)
- 200 hrs: Implementation
- 50 hrs: Writing project report

Note that there are other projects (PSI-ProjectCAD-M, PSI-ProjectPAD, PSI-ProjectSP-M) with different workloads.

prerequisites for the module:

none

Recommended prior knowledge: 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.	**Admission requirements:** none

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.		
Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 2 Semester

Module Units	
Software Systems Science Project: Security and Privacy **Mode of Delivery:** Practicals **Language:** English/German **Frequency:** every semester **Learning outcome:** cf. module description **Contents:** cf. module description **Literature:** Literature will be announced at the beginning of the project.	**8,00 Weekly Contact Hours**

Examination Coursework Assignment / Duration of Coursework: 3 months **prerequisites for module examination:** Regular attendance **Description:** The module examination consists of two module examination segments. The respective weights of the two module examination segments will be announced at the beginning of the semester in which the project starts. The first segment of the module examination consists of a written report (in English) that includes any source code, datasets, and analysis scripts. The maximum number of points that can be achieved in this part of the module examination is 100. Optionally, participants can submit intermediary results (in English) to collect up to 20 bonus points. If this part of 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 this part of 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.	

Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 3 months **prerequisites for module examination:**	

Regular attendance **Description:** The module examination consists of two module examination segments. The respective weights of the two module examination segments will be announced at the beginning of the semester in which the project starts. The second segment of the module examination consists of two parts: Firstly, the participants submit a written report (in English) that includes any 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. Optionally, participants can submit intermediary results (in English) to collect up to 20 bonus points. If this part of 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 this part of 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.	

Module PSI-Sem-B Seminar Security and Privacy Foundations *Seminar Security and Privacy Foundations*		3 ECTS / 90 h
(since SS20) Person responsible for module: Prof. Dr. Dominik Herrmann		
Contents: This seminar covers various topics related to information security and privacy. A list of available topics will be published on the website of the Privacy and Security in Information Systems Group before the first session of the seminar. Participants will form small groups and work on one of the topics.		
Learning outcomes: Participants will learn to search for, read, and discuss scientific literature in the subject area of information security and privacy. They will also learn how to write scientific texts and how to present their findings.		
Remark: This seminar will be offered in English unless all participants speak German.		
prerequisites for the module: none		
Recommended prior knowledge: Basic knowledge in the area of computer science (e.g. as covered in the module EiRBS) are helpful, but not required.		**Admission requirements:** none
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Seminar Security and Privacy Foundations **Mode of Delivery:** Seminar **Lecturers:** Prof. Dr. Dominik Herrmann **Language:** English/German **Frequency:** every summer semester **Learning outcome:** cf. module description **Contents:** cf. module description **Literature:** Relevant literature will be provided when the topics are assigned.	**2,00 Weekly Contact Hours**

Examination Coursework Assignment with presentation / Duration of Examination: 30 minutes Duration of Coursework: 3 months **prerequisites for module examination:** Continuous attendance in the seminar sessions is mandatory, cf. §9 (10) APO. **Description:** Participants write a seminar thesis and give a talk summarizing their findings.	

Module PSI-Sem-M Seminar Research Topics in Security and Privacy *Seminar Research Topics in Security and Privacy*	3 ECTS / 90 h

(since SS20)
Person responsible for module: Prof. Dr. Dominik Herrmann

Contents:
This seminar provides in-depth coverage of advanced topics in one of the fields of information security and privacy.

Participants learn to review, analyze, and discuss scientific sources (books and essays). While participants are expected to perform the actual research independently and mostly on their own, the instructors provide extensive support throughout the seminar. The instructors will provide guidance on scientific methods, e.g., how to approach a topic, how to find relevant literature, how to read a paper efficiently, how to write a seminar report, and how to give a good talk.

Participants will be asked to deliver manageable chunks of work throughout the semester (such as summarizing literature in a survey, reviewing the work of others, writing a draft of the term paper, reviewing the draft of other students, etc.). They will receive feedback by their peers and by the instructors.

The actual topics are subject to change. A list of available topics is made available before the first session via UnivIS or VC.

Learning outcomes:
The participants learn to find, read, and summarize scientific texts. They also learn to assess statements and to discuss them critically. Finally, they learn to write scientific texts and to present their results in a talk.

Students who participate in the optional peer review process will also learn techniques to give useful feedback to others as well as how to accept feedback for one's own work.

Remark:
The default language in this seminar is English, unless all participants are fluent in German.

prerequisites for the module:
none

Recommended prior knowledge: Participants should have basic knowledge in software engineering, foundations of computing, operating systems, and networks. Knowledge in information security and privacy (obtained, e.g., in PSI-IntroSP-B and by having completed a seminar or thesis in the field of information security) is strongly recommended.	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Seminar Research Topics in Security and Privacy **Mode of Delivery:** Seminar **Language:** English/German **Frequency:** every winter semester **Contents:**	**2,00 Weekly Contact Hours**

cf. module description

Literature:

- Alley: The Craft of Scientific Writing
- Anderson: Security Engineering
- Pfleeger et al.: Security in Computing
- Stallings & Brown: Computer Security: Principles and Practice
- Strunk & White: The Elements of Style

Other relevant literature is presented in the first session.

Examination

Coursework Assignment with presentation / Duration of Examination: 30 minutes
Duration of Coursework: 3 months

prerequisites for module examination:

Continuous attendance in the seminar sessions is mandatory, cf. §9 (10) APO.

Description:

The module examination consists of two parts, a term paper (in English) and a talk (in English; in German if all participants are fluent in German). The maximum number of points that can be achieved in the module examination is 100. Details regarding the number of points that can be achieved in the talk and in the report will be announced in the first session of the project.

Optionally, participants can submit intermediary results (in English) such as surveys, written reviews for the work of other participants, and a draft of the term paper. Participants can thereby earn 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.

Module SME-Projekt-B Bachelor's project on Smart Environments *Bachelorprojekt zu Smart Environments*	6 ECTS / 180 h 50 h Präsenzzeit 130 h Selbststudium

(since WS18/19)
Person responsible for module: Prof. Dr. Diedrich Wolter

Contents:
This module addresses applications of methods from the topic area Smart Environments in context of a software development project. To this end, a smart system will be develop to tackle a practical application problem, focusing on the software development. Among the methods used, artificial intelligence techniques play an important role.

Learning outcomes:

- gain skills to identify relevant methods to solve a practical problem
- gain competence to apply a basic method to a practical problem
- gain experience with problems that can arise applying a basic method to a practical problem
- improve programming skills
- learn to evaluate utility of approaches with respect to practical problems
- learn to present results in a scientific paper and defend the work in a colloquium

Remark:
The language of instruction in this course is German. However, all course materials are available in Englisch. Term papers and presentations may be delivered in either German or English.

prerequisites for the module:
none

Recommended prior knowledge: Basic skills in computer science, especially programming skills are highly recommended.		**Admission requirements:** none
Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Übung Bachelorprojekt zu Smart Environments **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Diedrich Wolter **Language:** German/English **Frequency:** every winter semester **Learning outcome:** siehe Modulbeschreibung **Contents:** The topic of the current project will tackled in small teams. In a problem-based manner, skills in scientific work and software development will be practised. **Literature:** will be announced in first meeting	**4,00 Weekly Contact Hours**

Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Umsetzung der Projektaufgabe, Dokumentation in Form eines wissenschaftlichen Aufsatzes als Hausarbeit sowie Präsentation im Kolloquium. Die Bekanntgabe der Prüfungssprache erfolgt in der ersten Sitzung der Lehrveranstaltung.	

Module SME-Projekt-M master project on smart environments *Masterprojekt zu Smart Environments*	6 ECTS / 180 h 50 h Präsenzzeit 130 h Selbststudium

(since WS17/18)
Person responsible for module: Prof. Dr. Diedrich Wolter

Contents:

This module addresses applications of advanced methods from the topic area Smart Environments. To this end, a smart system will be develop to tackle a practical application problem, focusing on the software development.

- research relevant literature
- develop own state-of-the-art approach
- system realization by implementation
- evaluation of system and its components
- presentation of results

Learning outcomes:

- gain skills to apply advanced methods from Smart Environments
- evaluate utility of approaches with respect to practical problems
- learn self-determined organisation of projects
- get acquainted with problems arising bridging theory and practice
- improve software development skills

Remark:

The main language in this course is English. Meetings may be held in German if all participating students are fluent in German. Presentations and term papers may be delivered in English or German.

prerequisites for the module:

none

Recommended prior knowledge: Basic knowledge in computer science (especially programming skills) is highly recommended, knowledge in Artificial Intelligence (AI) or Smart Environments helpful.	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Masterprojekt zu Smart Environments **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Diedrich Wolter **Language:** German/English **Frequency:** every summer semester **Learning outcome:** siehe Modulbeschreibung **Contents:**	**4,00 Weekly Contact Hours**

Im Master-Projekt werden wechselnde Themen aus dem Gebiet Smart Environments in Kleingruppen bearbeitet. Problem-basiert wird dabei wissenschaftliches Arbeiten und das Entwickeln eigener Lösungsansätze geübt. **Literature:** wird in der Lehrveranstaltung vorgestellt	

Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Umsetzung der Projektaufgabe, Dokumentation in Form eines wissenschaftlichen Aufsatzes als Hausarbeit. Die Prüfungssprache wird während der ersten Sitzung der Lehrveranstaltung bekanntgegeben.	

Module SME-STE-M Introduction to Knowledge Representation: Space, Time, Events *Introduction to Knowledge Representation: Space, Time, Events*	6 ECTS / 180 h

(since WS21/22)
Person responsible for module: Prof. Dr. Diedrich Wolter

Contents:

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:

- fundamental concepts: knowledge, abstractions, relations, logics
- syntax and semantics, formalization of knowledge
- representation and reasoning
- qualitative algebras and constraint calculi
- constraint-based reasoning
- spatial logics
- complexity and tractable subclasses

Learning outcomes:

- gain overview of formalisms for representing spatio-temporal logics
- gain skills to represent spatio-temporal knowledge symbolically
- gain overview of reasoning problems and learn to identify approaches for solving them
- learn to apply constraint-based reasoning methods
- learn to identify computational complexity of reasoning problems

Remark:

The main language of instruction in this course is English. Exams may be taken in either English or German. The lectures and tutorials may be delivered in German if all participating students are fluent in German.

prerequisites for the module:

none

Recommended prior knowledge: Basic knowledge in computer science is recommended, for example obtained in a computer science bachelor's curriculum.	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Lectures Introduction to Knowledge Representation: Space, Time, Events **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Diedrich Wolter	**2,00 Weekly Contact Hours**

Language: English/German **Frequency:** every winter semester **Learning outcome:** see description of module **Contents:** see description of module **Literature:** will be announced in first lecture	
2. Practicals Introduction to Knowledge Representation: Space, Time, Events **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Diedrich Wolter **Language:** German **Frequency:** every winter semester **Contents:** practical exercises according to the lecture	**2,00 Weekly Contact Hours**

Examination Oral examination / Duration of Examination: 20 minutes **Description:** oral examination 20 minutes about lectures and practicals	

Module SME-Sem-B Bachelor seminar on Smart Environments *Bachelorseminar zu Smart Environments*		3 ECTS / 90 h
(since WS20/21) Person responsible for module: Prof. Dr. Diedrich Wolter		
Contents: Selected topics within the area of Smart Environments are covered. Topics in this area relate to application areas such as interactive systems as well as to computer science areas such as Artificial Intelligence.		
Learning outcomes: Competences in scientific work will be acquired, in particular systematic literature research, structuring of complex topics, and (comparative) evaluation. Presentation skills to communicate specialized topics as well as scientific writing will be trained.		
Remark: The main language of instruction in this course is German. Presentations and reports may also be delivered in English.		
prerequisites for the module: none		
Recommended prior knowledge: none		**Admission requirements:** none
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** Semester

Module Units	
Bachelorseminar zu Smart Environments **Mode of Delivery:** Seminar **Lecturers:** Prof. Dr. Diedrich Wolter **Language:** German/English **Frequency:** every winter semester **Learning outcome:** see description of module **Contents:** see description of module **Literature:** will be announced in first meeting	**2,00 Weekly Contact Hours**

Examination	
Internship report / Duration of Examination: 30 minutes Duration of Coursework: 4 months	

Module SME-Sem-M master seminar on Smart Environments *Masterseminar zu Smart Environments*	3 ECTS / 90 h
(since WS20/21) Person responsible for module: Prof. Dr. Diedrich Wolter	
Contents: Selected topics within the area of Smart Environments are covered. Topics will relate to computer science areas such as Artificial Intelligence and knowledge representation.	
Learning outcomes: Competences in scientific work will be acquired, in particular systematic literature research, structuring of complex topics, and (comparative) evaluation of complex approaches. Presentation skills to communicate specialized topics as well as scientific writing will be trained.	
Remark: The main language of instruction in this course is English. However, the meetings may be held in German if all participating students are fluent in German. The written reports/seminar essay and the presentation may be delivered in English or in German.	
prerequisites for the module: none	
Recommended prior knowledge: basic knowledge in computer science (e.g., acquired in a Bachelor's curriculum)	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** Semester

Module Units	
Masterseminar Smart Environments **Mode of Delivery:** Seminar **Lecturers:** Prof. Dr. Diedrich Wolter **Language:** English/German **Frequency:** every summer semester **Learning outcome:** see description of module **Contents:** see description of module **Literature:** will be announced in first meeting	**2,00 Weekly Contact Hours**

Examination Coursework Assignment with presentation / Duration of Examination: 30 minutes Duration of Coursework: 4 months **Description:**	

Schriftliche Ausarbeitung und Vortrag zu dem im Seminar von der Teilnehmerin bzw. vom Teilnehmer bearbeiteten Thema, inkl. Diskussion.	

Module SNA-ASN-M Social Network Analysis *Analyse sozialer Netzwerke*	6 ECTS / 180 h

(since WS17/18)
Person responsible for module: Prof. Dr. Kai Fischbach

Contents:
Social network analysis focuses on relationships between or among social entities. This course presents an introduction to various concepts, methods, and applications of social network analysis. The primary focus of these methods is the analysis of relational data measured on populations of social actors.

Learning outcomes:
Erwerb vertiefter Kenntnisse der Methoden und Modelle der Netzwerkanalyse. Die Studierenden verstehen die Bedeutung der Struktur sozialer Netzwerke für die Effektivität und Effizienz betrieblicher Arbeitsprozesse. Sie erlernen methodische Grundlagen der Analyse sozialer Netzwerke und die Bewertung ihrer strukturellen Eigenschaften. Sie sind in der Lage, ihre Kenntnisse auf Forschungsfragen der Wirtschaftsinformatik anzuwenden.

Remark:
The language of instruction in this course is German. However, the exam is available in English.

prerequisites for the module:
none

Recommended prior knowledge: keine		**Admission requirements:** keine
Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Analyse sozialer Netzwerke **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Kai Fischbach **Language:** German **Frequency:** every winter semester	**2,00 Weekly Contact Hours**

Contents:
Topics include an introduction to graph theory and the use of directed graphs and matrices to study actor interrelations; structural and locational properties of actors, such as centrality, prestige, and prominence; subgroups and cliques; equivalence of actors, including structural equivalence and, blockmodels;local analyses, including dyadic and triad analysis; and introduction to statistical global analyses, using models such as p* and their relatives. Methods are illustrated on a wide range of social network examples using both standard social network analysis software and special purpose computer programs.

Literature:

- Carrington PJ, Scott J, Wasserman S (2005) Models and Methods in Social Network Analysis. Cambridge University Press, New York.
- Knoke D, Yang S (2007) Social Network Analysis, 2. Auflage. Sage Publications, Thousand Oaks

• Newman MEJ (2010) Networks. An Introduction. Oxford University Press, Oxford. • Wasserman S, Faust K (1994) Social Network Analysis: Methods and Applications. Cambridge University Press, New York.	
2. Analyse sozialer Netzwerke **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Wirtschaftsinf, Soz Netzwerke **Language:** German **Frequency:** every winter semester **Contents:** Die Inhalte der Vorlesung werden anhand von Übungsaufgaben und Fallbeispielen vertieft. Praktische Übungen werden unter Verwendung gängiger Software zur Analyse sozialer Netzwerke durchgeführt. **Literature:** • Borgatti SP, Everett MG & Freeman LC (2002) Ucinet for Windows: Software for Social Network Analysis. Analytic Technologies, Harvard. • Nooy W, Mrvar A, Batagelj V (2011) Exploratory Social Network Analysis with Pajek. Revised and Expanded Second Edition. Cambridge University Press, New York.	**2,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 90 minutes **Description:** In der Klausur werden die in Vorlesung und Übung behandelten Inhalte geprüft. Es können 90 Punkte erzielt werden. 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 bekannt gegeben. Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.	

Module SNA-NET-M Network Theory *Netzwerktheorie*	6 ECTS / 180 h

(since WS17/18)
Person responsible for module: Prof. Dr. Kai Fischbach

Contents:
Individuals and technology shape and are shaped by organizations. Individuals and organizations are also affected by sets of interlinked networks linking people, technology, organizations, knowledge and resources. In this world of networks and organizations, how do coordination, communication, power, tasks, goals, and information interact to affect group and organizational behavior and the impact of information technology on this behavior? How do we conceptualize, measure, and evaluate organizations and networks? How do we evaluate the impact of policies and technology on these organizations and networks especially given the fact that organizations and networks are dynamic?

Learning outcomes:
Die Studierenden kennen interdisziplinäre Theoriebeiträge zur Erklärung der Struktur und Dynamik sozialer Netzwerke und können das erworbene Wissen auf relevante Forschungsfragen der Wirtschaftsinformatik anwenden. Sie verstehen den Einfluss der Struktur eines Netzwerkes auf seine internen Prozesse und die Veränderung der Struktur eines Netzwerkes im Zeitverlauf.

Themenfelder:

- Theorien sozialer und komplexer Netzwerke
- Emergenz und Dynamik sozialer Netzwerke
- Agentenbasierte Modellierung und Spieltheorie
- Informationsverarbeitung in sozialen Netzwerken
- Netzwerkprozesse
- Wissensnetzwerke

Remark:
The language of instruction in this course is German. However, the exam is available in English.

prerequisites for the module:
none

Recommended prior knowledge: Kenntnisse aus dem Modul Analyse sozialer Netzwerke sind wünschenswert, jedoch nicht Voraussetzung	**Admission requirements:** keine

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Netzwerktheorie **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Kai Fischbach **Language:** German **Frequency:** every summer semester **Contents:** This course provides an overview of the dominant perspectives on organizations and networks from a macro perspective. Topics covered include knowledge	**2,00 Weekly Contact Hours**

management, organizational design, organizational learning, organizational evolution and population ecology, organizational culture, organizations as complex systems, social and organizational networks, and dynamic network analysis. **Literature:** • Easley D, Kleinberg J (2010) Networks, Crowds, and Markets. Reasoning about a Highly Connected World. Cambridge University Press, New York • Goyal S (2009) Connections: An Introduction to the Economics of Networks, Princeton University Press, Princeton und Oxford • Jackson MO (2008) Social and Economic Networks. Princeton University Press, Princeton und Oxford • Kilduff M, Tsai W (2003) Social Networks and Organizations. Sage Publications, Thousand Oaks • Monge PR, Contractor N (2003) Theories of Communication Networks. Oxford University Press, New York	
2. Netzwerktheorie **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Wirtschaftsinf, Soz Netzwerke **Language:** German **Frequency:** every summer semester **Contents:** Die Inhalte der Vorlesung werden anhand von Übungsaufgaben und Fallbeispielen vertieft. Praktische Übungen werden unter Verwendung gängiger Software zur Analyse sozialer Netzwerke durchgeführt. **Literature:** Siehe Vorlesung.	**2,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 90 minutes **Description:** In der Klausur werden die in Vorlesung und Übung behandelten Inhalte geprüft. Es können 90 Punkte erzielt werden. 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 bekannt gegeben. Eine Bewertung von 1,0 kann auch ohne Punkte aus den Studienleistungen erreicht werden.	

Module SNA-OSN-M Project Online Social Networks *Projekt zu Online Social Networks*	6 ECTS / 180 h

(since WS13/14)
Person responsible for module: Prof. Dr. Kai Fischbach
further responsible : Zylka, Matthäus, Dipl.-Wirt.-Inf.

Contents:
This module is an introduction to the analysis of online social networks. The aim is twofold: to provide students with the tools necessary to undertake research into online networks, and to give an overview of the type of questions these data can answer.

Learning outcomes:
At the conclusion of the course, students should know not only how to calculate basic network metrics on pre-existing data sets, but also how to capture an online social network efficiently with the intent of answering a specific research question.

Further goals:

- Learn how the radical innovation process in small teams works
- Learn how to collaborate in multidisciplinary intercultural virtual teams
- Learn how to find trendsetter and trends on the Internet and social media
- Learn how to predict trends using SNA und statistical forecasting techniques

Remark:
The main language of instruction in this course is English. The written reports/seminar essay and the presentation have to be delivered in English.

prerequisites for the module:
none

Recommended prior knowledge: We recommend attending at least one of the following courses: • Social Network Analysis (SNA-ASN-M) • Theories of Social Networks (SNA-NET-M)	**Admission requirements:** keine

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Online Social Networks **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Kai Fischbach **Language:** English/German **Frequency:** every winter semester	**4,00 Weekly Contact Hours**

Contents:
The course will define online networks, examine how they differ from offline social networks, and consider theoretical and methodological issues associated with their analysis. The sessions will explore different strategies to retrieve and analyze online network data, and present different empirical scenarios to which those tools have been applied.

Literature:

• Gloor, P. A. Swarm Creativity, Competitive Advantage Through Collaborative Innovation Networks. Oxford University Press, 2006 Further literature will be announced in the lecture.	

Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 4 months **prerequisites for module examination:** Regelmäßige Teilnahme an der Lehrveranstaltung **Description:** Die Gewichtung der Prüfungsleistungen Hausarbeit und Kolloquium wird zu Beginn der Lehrveranstaltung von der Dozentin bzw. dem Dozenten bekannt gegeben.	

Module SNA-WIM-B Knowledge- and Informationmanagement *Wissens- und Informationsmanagement*	6 ECTS / 180 h

(since WS18/19)
Person responsible for module: Prof. Dr. Kai Fischbach

Contents:

Die Veranstaltung bietet eine Einführung in das betriebliche Wissens- und Informationsmanagement.

Learning outcomes:

Ziel der Veranstaltung ist die Vermittlung folgender Kenntnisse und Fähigkeiten:

- Studierende kennen und verstehen relevante Begriffe, Modelle und Theorien des Wissens- und Informationsmanagements.
- Studierende können die Modelle und Theorien zur Analyse und Bewertung verschiedener Wissens- und Informationsmanagementaspekte anwenden.
- Studierende kennen verschiedene Wissens- und Informationsmanagementsysteme, die im inner- und überbetrieblichen Bereich zum Einsatz kommen.
- Studierende verstehen, wie Wissensmanagementsysteme geeignet gestaltet und genutzt werden können.
- Studierende verstehen die Bedeutung sozialer Netzwerke für das Wissensmanagement.

prerequisites for the module:

none

Recommended prior knowledge: none		**Admission requirements:** none
Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Wissens- und Informationsmanagement **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Kai Fischbach **Language:** German **Frequency:** every summer semester	**2,00 Weekly Contact Hours**

Contents:

Vor dem Hintergrund der Globalisierung und Digitalisierung sowie der damit einhergehenden Intensivierung und Diversifizierung der Vernetzung erlangt das effektive und effiziente Management der Ressourcen Information und Wissen in Unternehmen strategischen Rang. Die Lehrveranstaltung befasst sich in diesem Kontext mit Zielen, Aufgaben, Systemen, Theorien und Methoden des Wissens- und Informationsmanagements. Dazu werden unter anderem die Wissensentwicklung, -verteilung, -nutzung, -bewertung, -bewahrung sowie der Wissenserwerb innerhalb von Unternehmen betrachtet.

Literature:

Dalkir, K. (2017): Knowledge Management in Theory and Practice. (3. Auflage). Cambridge, Massachusetts: The MIT Press. Weitere Literatur wird in der Veranstaltung bekannt gegeben.	
2. Wissens- und Informationsmanagement **Mode of Delivery:** Practicals **Lecturers:** Diana Fischer-Preßler **Language:** German **Frequency:** every summer semester **Contents:** Die Übung Wissens- und Informationsmanagement dient der Vertiefung, Übung und Anwendung des in der Vorlesung vermittelten Stoffs. Dazu werden Aufgaben und Methoden des Wissens- und Informationsmanagements behandelt und Fallstudien in Gruppen bearbeitet. **Literature:** siehe Vorlesung	**2,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 90 minutes **Description:** 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, werden 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.	

Module SWT-ASV-M Applied Software Verification *Applied Software Verification*	6 ECTS / 180 h

(since WS19/20)
Person responsible for module: Prof. Dr. Gerald Lüttgen

Contents:

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.

Learning outcomes:

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.

Remark:

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:

- 30 hrs. attending lectures (Vorlesungen)
- 30 hrs. attending practicals (Übungen)
- 60 hrs. preparing and reviewing the lectures and practicals, including researching literature, studying material from additional sources and applying software tools
- 30 hrs. working on the assignment (Hausarbeit)
- 30 hrs. preparing for the colloquium (Kolloquium)

prerequisites for the module:

none

Recommended prior knowledge: 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.	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Applied Software Verification **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Gerald Lüttgen **Language:** English **Frequency:** every summer semester **Contents:** The lectures (Vorlesungen) will address the following topics in automated software verification: (i) state machines, assertions and algorithms for state	**2,00 Weekly Contact Hours**

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, several state-of-the-art software verification tools will be introduced.

Literature:

- Baier, C., Katoen, J.-P. Principles of Model Checking. MIT Press, 2008.
- Clarke, E., Grumberg, O., Kroening, D., Peled, D. and Veith, H. Model Checking. 3rd. ed. MIT Press, 2018.
- Huth, M. and Ryan, M. Logic in Computer Science. 2nd ed. Cambridge University Press, 2004.
- Kroening, D. and Strichman, O. Decision Procedures: An Algorithmic Point of View. Springer, 2008.
- Loeckx, J. and Sieber, K. The Foundations of Program Verification. 2nd ed. Wiley, 1987.

2. Applied Software Verification

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Lecturers: Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen
Language: English
Frequency: every summer semester

Contents:

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).

Literature:

- see the corresponding lectures -

Examination

Coursework Assignment and Colloquium / Duration of Examination: 20 minutes
Duration of Coursework: 3 weeks

Description:

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.

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.

Module SWT-FSA-B Foundations of Software Analysis

Foundations of Software Analysis

6 ECTS / 180 h

(since WS20/21)
Person responsible for module: Prof. Dr. Gerald Lüttgen

Contents:

This theory module offers students a mathematical perspective on sequential, imperative computer programs and concurrent, distributed software. It studies the formal semantics of software which enables the analysis and verification of complex digital systems.

Learning outcomes:

On completion of this module, students will be able to understand and apply core foundational concepts of, and techniques behind, program semantics and software verification, in the context of both sequential and concurrent systems.

Remark:

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:

- 60 hrs. attending lectures (Vorlesungen)
- 30 hrs. attending practicals (Übungen)
- 60 hrs. preparing and reviewing the lectures and practicals, including researching literature, studying material from additional sources
- 30 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium)

prerequisites for the module:

none

Recommended prior knowledge:	**Admission requirements:**
Basic knowledge in discrete mathematics, logics and algebra, such as acquired in the modules "Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik)" (GdI-MfI-1) and "Mathematik für Informatik 2 (Lineare Algebra)" (KTR-MfI-2).	none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

1. Foundations of Software Analysis **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Gerald Lüttgen **Language:** English/German **Frequency:** every winter semester	**4,00 Weekly Contact Hours**
Contents: Students will be introduced to the foundations of software analysis and their applications to software verification. Particular emphasis will be put on program semantics and mathematical theories for reasoning about sequential and concurrent systems. The following topics will be covered: Part I: Mathematical Foundations	

- Inductive definitions and proofs
- Orders, functions and fixed points
- Algebraic structures, equivalences and congruences
- Algebraic laws and logic systems

Part II: Sequential, Imperative Programs

- The imperative language IMP
- Natural, structural operational and denotational semantics
- The Hoare calculus

Part III: Concurrent, Distributed Software

- The process calculus CCS
- Strong and weak bisimulation
- Algebraic laws and axiomatization

Literature:

- Bruni, R. and Montanari, U., Models of Computation. Springer, 2017.
- Milner, R. Communication and Concurrency. Prentice Hall, 1989.
- Nielson, H. R. and Nielson, F., Semantics with Applications: An Appetizer. Springer, 2007.
- Loeckx, J. and Sieber, K. The Foundations of Program Verification, 2nd ed. Wiley, 1987.
- Steffen, B., Rüthing, O. and Huth, M. Mathematical Foundations of Advanced Informatics. Springer, 2018.
- Davey, B. A. and Priestley, H. A. Introduction to Lattices and Order, 2nd ed. Cambridge University Press, 2002.

2. Foundations of Software Analysis

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Lecturers: Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen
Language: English/German
Frequency: every winter semester

Contents:

The practicals (Übungen) cover pen-and-paper exercises that will deepen the concepts and techniques taught in the lectures (Vorlesungen), and apply them to the analysis and verification of small examples of software. Emphasis will be put on presenting and discussing the solutions to the exercises by and among the students.

Literature:

- see the corresponding lectures -

Examination

Coursework Assignment and Colloquium / Duration of Examination: 20 minutes
Duration of Coursework: 3 weeks

Description:

Assignment (Hausarbeit) consisting of questions practicing, reviewing and deepening the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen). 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).	

<table>
<tr><td colspan="2"><b>Module SWT-FSE-B Foundations of Software Engineering</b>
<i>Foundations of Software Engineering</i></td><td>6 ECTS / 180 h</td></tr>
<tr><td colspan="3">(since WS19/20)
Person responsible for module: Prof. Dr. Gerald Lüttgen</td></tr>
<tr><td colspan="3"><b>Contents:</b>
This module teaches the foundations of software engineering that are applicable to various kinds of software systems – from information systems to embedded systems. It focusses on technologies, notations and processes for system specification, design, implementation, and verification and validation.</td></tr>
<tr><td colspan="3"><b>Learning outcomes:</b>
Students will receive an introduction to the common problems and paradigms in, and foundations of, software development. They will also gather conceptional and practical knowledge in the analysis, design and testing of software, with an emphasis on technical aspects of specifying, designing, implementing, verifying and validating software.</td></tr>
<tr><td colspan="3"><b>Remark:</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:
• 45 hrs. attending lectures (Vorlesungen)
• 30 hrs. reviewing the lectures, including researching and studying material from additional sources
• 45 hrs. attending practicals (Übungen)
• 30 hrs. preparing and reviewing the practicals, including researching and studying material from additional sources
• 30 hrs. preparing for the written exam (Klausur)</td></tr>
<tr><td colspan="3"><b>prerequisites for the module:</b>
none</td></tr>
<tr><td colspan="2"><b>Recommended prior knowledge:</b>
Basic knowledge in Computer Science, as well as knowledge in programming in Java and in algorithms and data structures.</td><td><b>Admission requirements:</b>
none</td></tr>
<tr><td><b>Frequency:</b> every summer semester</td><td><b>Recommended semester:</b></td><td><b>Minimal Duration of the Module:</b>
1 Semester</td></tr>
</table>

<table>
<tr><td colspan="2"><b>Module Units</b></td></tr>
<tr><td><b>1. Foundations of Software Engineering</b>
<b>Mode of Delivery:</b> Lectures
<b>Lecturers:</b> Prof. Dr. Gerald Lüttgen
<b>Language:</b> English/German
<b>Frequency:</b> every summer semester
<b>Contents:</b>
The lectures (Vorlesungen) provide an introduction to the foundations of software engineering, including commonly used technologies, notations and processes for all software engineering phases. In particular, conceptual and technical aspects of software specification, architecture and design, and verification and validation</td><td><b>3,00 Weekly Contact Hours</b></td></tr>
</table>

are discussed, such as the Unified Modeling Language (UML) and its semantics, model-driven and pattern-based development, and software testing. Students are also introduced to specific aspects of agile software development. **Literature:** • Sommerville, I. Software Engineering, 10th ed. Pearson, 2016. • Robertson, S. and Robertson, J. Mastering the Requirements Process, 3rd ed. Addison-Wesley, 2012. • Cohn, M. User Stories Applied. Addison-Wesley, 2004. • Stevens, P. and Pooley, R. Using UML - Software Engineering with Objects and Components, 2nd. ed. Addison-Wesley, 2006. • Freeman, E., Robson, E., Sierra, K. and Bates, B. Head First Design Patterns. O'Reilly, 2004. • Gamma, E., Helm, R., Johnson, R. and Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Design. Prentice Hall, 1994. Further literature will be announced in the lectures.	
2. Foundations of Software Engineering **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Gerald Lüttgen, Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen **Language:** English/German **Frequency:** every summer semester **Contents:** The practicals (Übungen) exercise and deepen the conceptual knowledge transferred via the lectures (Vorlesungen), and relay practical knowledge in software engineering. **Literature:** - see the corresponding lectures -	**3,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 120 minutes **Description:** Written exam (Klausur) consisting of questions that relate to the contents of the lectures (Vorlesungen) and practicals (Übungen) of this module. The written exam is set in English, while answers may be provided in either English or German. The exam is passed if at least 50% of the available points are reached.	

Module SWT-PCC-M Principles of Compiler Construction *Principles of Compiler Construction*	6 ECTS / 180 h
(since WS20/21) Person responsible for module: Prof. Dr. Gerald Lüttgen	
Contents: The module teaches the theoretical and practical principles of compiler construction, from lexical analysis and parsing, to semantic analysis, to code generation and optimisation.	
Learning outcomes: 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.	
Remark: 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: • 30 hrs. attending lectures (Vorlesungen) • 30 hrs. reviewing the lectures, including researching and studying material from additional sources • 30 hrs. attending practicals (Übungen) • 30 hrs. preparing and reviewing the practicals, including researching and studying material from additional sources • 60 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium)	
prerequisites for the module: none	
Recommended prior knowledge: 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.	**Admission requirements:** none

Frequency: every summer semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
1. Principles of Compiler Construction **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Gerald Lüttgen **Language:** English/German **Frequency:** every summer semester **Contents:** 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	**2,00 Weekly Contact Hours**

focus on the following aspects of compiler construction: lexical analysis, parsing, abstract syntax, semantic analysis, code generation and code optimisation.

Literature:

- Louden, K. C. Compiler Construction: Principles and Practice. Course Technology, 1997.
- Aho, A. V., Lam, M. S., Sethi, R. and Ullman, J. D. Compilers: Principles, Techniques, and Tools, 2nd ed. Pearson, 2007.
- Fischer, C. N., Cytron, R. K. and LeBlanc Jr., R. J. Crafting a Compiler. Pearson, 2010.
- Muchnick, S. S. Advanced Compiler Design and Implementation, Morgan Kaufmann, 1997.

2. Principles of Compiler Construction

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Lecturers: Prof. Dr. Gerald Lüttgen, Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen
Language: English/German
Frequency: every summer semester

Contents:

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).

Literature:

- see the corresponding lectures -

Examination

Coursework Assignment and Colloquium / Duration of Examination: 20 minutes
Duration of Coursework: 3 weeks

Description:

Assignment (Hausarbeit) consisting of questions practicing, reviewing and deepening the knowledge transferred in the lectures (Vorlesungen) and practicals (Übungen). The examination is set in English language, while answers may be provided in either English or German.

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.

Module SWT-PR1-M Masters Project in Software Engineering and Programming Languages *Masterprojekt Softwaretechnik und Programmiersprachen*		6 ECTS / 180 h
(since WS19/20) Person responsible for module: Prof. Dr. Gerald Lüttgen		
Contents: **Attention: The module SWT-PR1-M cannot take place in winter semester 2019/20!** Topics in Software Engineering and Programming Languages 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. 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.		
Learning outcomes: 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.		
Remark: 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. The total workload of 180 hrs. is split approximately as follows: • 10 hrs. participating in introductions to and tutorials on methods, software tools, and giving presentations on the project status • 20 hrs. completing the exercises for bonus points • 115 hrs. researching and familiarization with the project topic and conducting the project work • 35 hrs. compilating a project report (Assignment/Hausarbeit) and preparation of the Colloquium (Kolloquium).		
prerequisites for the module: none		
Recommended prior knowledge: Basic knowledge in software engineering and programming languages, knowledge in the subject matter of the project topic.		**Admission requirements:** none
Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Masters Project in Software Engineering and Programming Languages **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Gerald Lüttgen, Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen **Language:** English/German **Frequency:** every semester **Learning outcome:** To be announced at the beginning of the project. **Contents:** Conduct of the project, accompanied by tutorials and regular project meetings. **Literature:** To be announced at the beginning of the project.	**4,00 Weekly Contact Hours**

Examination Coursework Assignment and Colloquium / Duration of Examination: 20 minutes Duration of Coursework: 12 weeks **prerequisites for module examination:** Regelmäßige Teilnahme an den zugehörigen Lehrveranstaltungen **Description:** Production of a written report on the software project carried out (Assignment/ Hausarbeit). The student may choose whether to write/compose the project report in English or German. Discussion of this project report and of the developed artefacts in the context of the wider project topic (Colloquium/Kolloquium). The examination language is either English or German and may be chosen by the student at the colloquium.	

Module SWT-PR2-M SWT Masters Project in Software Systems Science *SWT Masters Project in Software Systems Science*	9 ECTS / 270 h

(since WS19/20)
Person responsible for module: Prof. Dr. Gerald Lüttgen

Contents:

Attention: The module SWT-PR2-M cannot take place in winter semester 2019/20!

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.

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.

Learning outcomes:

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.

Remark:

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.

The total workload of 270 hrs. is split approximately as follows:

- 10 hrs. participating in introductions to and tutorials on methods, software tools, and giving presentations on the project status
- 20 hrs. completing the exercises for bonus points
- 195 hrs. researching and familiarization with the project topic and conducting the project work
- 45 hrs. compiling a project report (Assignment/Hausarbeit) and preparation of the Colloquium (Kolloquium).

The project report can be written/composed in either English or German.

prerequisites for the module:

none

Recommended prior knowledge: Basic knowledge in software engineering and programming languages, knowledge in the subject matter of the project topic.	**Admission requirements:** none

Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
SWT Masters Project in Software Systems Science **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Gerald Lüttgen **Language:** English/German **Frequency:** every semester **Learning outcome:** To be announced at the beginning of the project. **Contents:** Conduct of the project, accompanied by tutorials and regular project meetings. **Literature:** To be announced at the beginning of the project.	**6,00 Weekly Contact Hours**

Examination Coursework Assignment and Colloquium / Duration of Examination: 30 minutes Duration of Coursework: 12 weeks **prerequisites for module examination:** Regular participation in the practicals. **Description:** 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).	

<table>
<tr><td colspan="2"><b>Module SWT-RSD-B Reactive Systems Design</b>
<i>Reactive Systems Design</i></td><td>6 ECTS / 180 h</td></tr>
<tr><td colspan="3">(since WS20/21)
Person responsible for module: Prof. Dr. Gerald Lüttgen</td></tr>
<tr><td colspan="3"><b>Contents:</b>
Reactive systems are digital systems that continuously react to their environment by reading sensor values, computing output values and emitting those values to actuators. Such systems are designed using domain-specific languages, and must often satisfy stringent real-time requirements. They are embedded in many parts of our daily lives: whether it is a home automation system, a driver's assistance system in a modern car, or sophisticated medical equipment at the hospital, we depend on the reliability, correctness, and quality of these systems' software.
This module discusses the theoretical concepts and the engineering practice of the model-driven development of reactive systems software. The module's foci are on the synchronous programming paradigm, on automatic code generation from system models, on techniques for verifying and testing reactive systems, and on deploying and integrating reactive software components on a specific operating system and execution platform.</td></tr>
<tr><td colspan="3"><b>Learning outcomes:</b>
On completion of this module, students will be able to understand the context and concepts of reactive systems design. They will be able to define domain-specific languages, to employ state-of-the-art techniques for the model-driven engineering of reactive software, and to apply methods for testing and verifying reactive systems. Moreover, they will appreciate the complexities surrounding the deployment and integration of reactive software components on a physical model railway platform, taking timing requirements into account.</td></tr>
<tr><td colspan="3"><b>Remark:</b>
The main language of instruction is English.
The total workload of 180 hrs. is split approximately as follows:
• 30 hrs. attending lectures (Vorlesungen)
• 30 hrs. attending practicals (Übungen)
• 60 hrs. preparing and reviewing the lectures and practicals, including researching literature, studying material from additional sources
• 60 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium)</td></tr>
<tr><td colspan="3"><b>prerequisites for the module:</b>
none</td></tr>
<tr><td colspan="2"><b>Recommended prior knowledge:</b>
Basic knowledge in discrete mathematics and programming, e.g., acquired in the modules "<i>Mathematik für Informatik 1 (Aussagen- und Prädikatenlogik)</i>" (GdI-MfI-1) and "<i>Einführung in Algorithmen, Programmierung und Software</i>" (DSG-EiAPS-B).</td><td><b>Admission requirements:</b>
none</td></tr>
<tr><td><b>Frequency:</b> every summer semester</td><td><b>Recommended semester:</b>
4.</td><td><b>Minimal Duration of the Module:</b>
1 Semester</td></tr>
</table>

Module Units

1. Reactive Systems Design

2,00 Weekly Contact Hours

Mode of Delivery: Lectures
Lecturers: Prof. Dr. Gerald Lüttgen, Eugene Yip
Language: English
Frequency: every summer semester

Learning outcome:
– see the module's learning outcomes/competences (Lernziele/Kompetenzen) listed above –

Contents:
Students are introduced to modern model-driven techniques, languages and tools for designing and programming reactive systems. The lectures first motivate reactive systems, present their basic design principles, examine the role of domain-specific languages, and study the synchronous programming paradigm. Then, techniques for verifying design properties via model checking, for automatically transforming design models into running code, and for automated testing are studied. The synchronous language and model-based development environment *KIELER SCCharts* is used for illustrating key semantic and engineering concepts.

Several topics on the deployment and integration of reactive software components on a physical execution platform are also addressed: the timing analysis problem, the mapping of components to real-time tasks, and a practical approach to execute components together in a semantics-preserving manner using the synchronous programming language ForeC.

Literature:

- Lee, E. A., and Seshia, S. A. Introduction to Embedded Systems: A Cyber-Physical Systems Approach, 2nd ed. MIT Press, 2017.
- Halbwachs, N. Synchronous Programming of Reactive Systems. Springer, 1993.
- Harel, D. and Politi, M. Modeling Reactive Systems with Statecharts. McGraw-Hill, 1998.
- Bettini, L. Implementing Domain Specific Languages with Xtext and Xtend, 2nd ed. Packt Publishing, 2016.

Further literature will be announced at the beginning of the module.

2. Reactive Systems Design

2,00 Weekly Contact Hours

Mode of Delivery: Practicals
Lecturers: Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen
Language: English
Frequency: every summer semester

Learning outcome:
– see the module's learning outcomes/competences (Lernziele/Kompetenzen) listed above –

Contents:

The practicals (Übungen) deepen the concepts and techniques taught in the lectures (Vorlesungen) and apply them to the development of reactive software. The latter involves a small programming project of a real model railway system with modern development tools such as *KIELER SCCharts* and a domain-specific language called *BahnDSL*. **Literature:** – see the corresponding lectures –	
Examination Coursework Assignment and Colloquium / Duration of Examination: 20 minutes Duration of Coursework: 3 weeks **Description:** The Assignment (Hausarbeit) consists of questions practicing, reviewing and deepening the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen); questions may also involve the practical use of the development tools introduced in the practicals. The Colloquium (Kolloquium) consists of questions testing the knowledge transferred in the lectures and practicals (Vorlesungen und Übungen), on the basis of the submitted solutions to the assignment (Hausarbeit).	

Module SWT-SEM-B Seminar in Software Engineering and Programming Languages (Bachelor) *Seminar Software Engineering and Programming Languages (Bachelor)*	3 ECTS / 90 h

(since WS17/18)
Person responsible for module: Prof. Dr. Gerald Lüttgen

Contents:
Current topics in software engineering and programming languages.

Learning outcomes:
Students will compile and acquire current topics in software engineering and programming languages by carrying out and documenting a guided literature survey, and by preparing and delivering a coherent, comprehensible presentation to their peers.

Remark:
The main language of instruction is English. The seminar may be delivered in German if all participating students are fluent in German. Regular participation in the presentations is required.

The total workload of 90 hrs. is split approximately as follows:

- 20 hrs. consultations and presentations (Referate), including discussions
- 25 hrs. literature research and familiarization and evaluation of literature
- 45 hrs. working on the assignment (Hausarbeit) and preparation for the presentation (Referat)

prerequisites for the module:
none

Recommended prior knowledge: Basic knowledge in software engineering and in programming languages.		**Admission requirements:** none
Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Software Engineering and Programming Languages (Bachelor) **Mode of Delivery:** Seminar **Lecturers:** Prof. Dr. Gerald Lüttgen, Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen **Language:** English/German **Frequency:** every semester **Contents:** Various current topics in software engineering and programming languages, which complement and/or extend the technical and methodological aspects of the degree programme's modules related to these fields. **Literature:** Literature will be allocated according to the topics to be discussed.	**2,00 Weekly Contact Hours**

Examination	

Coursework Assignment with presentation / Duration of Examination: 40 minutes Duration of Coursework: 8 weeks **prerequisites for module examination:** Regular participation in the seminar. **Description:** Assignment (Hausarbeit) consisting of a written report on the topic assigned to the student. Presentation (Referat) on the topic assigned to the student, including a discussion.	

Module SWT-SEM-M Seminar in Software Engineering and Programming Languages (Master) *Seminar Software Engineering and Programming Languages (Master)*	3 ECTS / 90 h

(since WS17/18)
Person responsible for module: Prof. Dr. Gerald Lüttgen

Contents:

Current topics in software engineering and programming languages. This may comprise the full spectrum of research topics in these fields, from the analysis, comparison and evaluation of current software technologies and tools, to the discussion and evaluation of novel research proposals.

Learning outcomes:

Students will compile and acquire current topics in software engineering and programming languages by independently carrying out and documenting a literature survey, and by preparing and delivering a coherent, comprehensible presentation to their peers. Students will also be able to scientifically discuss topics in software engineering and programming languages with their peers.

Remark:

The main language of instruction is English. The seminar may be delivered in German if all participating students are fluent in German. Regular participation in the presentations is required.

The total workload of 90 hrs. is split approximately as follows:

- 20 hrs. consultations and presentations (Referate), including discussions
- 25 hrs. literature research and familiarization and evaluation of literature
- 45 hrs. working on the assignment (Hausarbeit) and preparation for the presentation (Referat)

prerequisites for the module:

none

Recommended prior knowledge: Basic knowledge in software engineering, in programming languages and in the subject matter of the seminar. Additionally, basic knowledge of scientific methods is expected.	**Admission requirements:** none

Frequency: every semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units	
Software Engineering and Programming Languages (Master) **Mode of Delivery:** Seminar **Lecturers:** Prof. Dr. Gerald Lüttgen, Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen **Language:** English/German **Frequency:** every semester **Contents:** Various current topics in software engineering and programming languages, which complement and/or extend the technical and methodological aspects of the degree programme's modules related to these fields. **Literature:** Will be allocated according to the topics to be discussed.	**2,00 Weekly Contact Hours**

Examination Coursework Assignment with presentation / Duration of Examination: 40 minutes Duration of Coursework: 8 weeks **prerequisites for module examination:** Regular participation in the seminar. **Description:** Assignment (Hausarbeit) consisting of a written report on the topic assigned to the student. Presentation (Referat) on the topic assigned to the student, including a discussion.	

Module SWT-SWL-B Software Engineering Lab *Software Engineering Lab*	6 ECTS / 180 h

(since WS21/22)
Person responsible for module: Prof. Dr. Gerald Lüttgen

Contents:
Small teams of students will conduct a software project, starting from a brief problem description.
This involves the application of modern software engineering tools, skills in collaboration and team organisation, and knowledge of processes and techniques for producing software artefacts and associated documents.

Learning outcomes:
Students will develop a piece of medium-sized software in small teams, thereby acquiring practical expertise in software engineering and skills in working in a software development team. In addition, this module deepens the students' programming proficiency and their understanding of flexible software engineering processes and of software and process quality, and familiarises them with the deployment and use of modern software engineering tools.

Remark:
The main language of instruction is English. The practicals may be delivered in German if all participating students are fluent in German. A regular attendance of team meetings and active participation is required throughout.

The total workload of 180 hrs. is split approximately as follows:

- 20 hrs. attending meetings of the student's team with the lecturer (Dozent) on planning, coordination and feedback
- 10 hrs. attending the accompanying practicals/tutorials (Übungen/Tutorials) on software tools
- 130 hrs. conducting the team project
- 20 hrs. working on the assignment (Hausarbeit) and preparing for the colloquium (Kolloquium)

prerequisites for the module:
none

Recommended prior knowledge: Basic knowledge in Computer Science and Software Engineering, as well as knowledge in Java programming and in programming in the small.	**Admission requirements:**

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** 1 Semester

Module Units

Software Engineering Lab **Mode of Delivery:** Practicals **Lecturers:** Prof. Dr. Gerald Lüttgen, Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen **Language:** German/English **Frequency:** every winter semester **Contents:**	**4,00 Weekly Contact Hours**

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

Literature:

- Tudose, C., Tahchiev, P., Leme, F., Massol, V. and Gregory, G. JUnit in Action, 3rd ed. Manning Publications, 2020.
- Loeliger, J. and McCullough, M. Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development, 2nd ed. O'Reilly, 2012.
- Vogel, L. Eclipse IDE. Lars Vogel, 2013. ISBN 3943747042.
- Schwaber, K. and Beedle, M. Agile Software Development with Scrum, Pearson, 2001.
- Cohn, M. User Stories Applied. Addison-Wesley, 2004.

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

Examination
Coursework Assignment and Colloquium / Duration of Examination: 45 minutes
Duration of Coursework: 2 weeks
prerequisites for module examination:
Regular participation in the associated practicals, including the participation in programming tasks.
Description:
Assignment (Hausarbeit) involving the compilation of a written project report in English or German language by each team, which shall cover the following topics:

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

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

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

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

Module SWT-SWQ-M Software Quality *Software Quality*	6 ECTS / 180 h

(since WS21/22)
Person responsible for module: Prof. Dr. Gerald Lüttgen

Contents:
Software quality is fundamental for a software product's reliable, safe and secure operation, for its maintainability and reusability, and for user and customer satisfaction. Engineering high-quality software products and managing their development involves the application of advanced techniques, methods and tools for software quality assurance. This module focuses, in particular, on model-based testing, software inspection, software measurement, and static analysis, which are indispensable in today's agile software engineering practice.

Learning outcomes:
On completion of this module, students will be familiar with important concepts and techniques of software quality and their role in modern software engineering. In particular, students will be able to apply state-of-the-art methods and tools for achieving and monitoring software quality, and devise strategies for software quality assurance in different product and organizational contexts.

Remark:
The language of instruction is English.

The total workload of 180 hrs. is split approximately as follows:

- 30 hrs. attending lectures (Vorlesungen)
- 30 hrs. attending practicals (Übungen)
- 90 hrs. preparing and reviewing lectures and practicals, researching literature and studying material from additional sources
- 30 hrs. preparing for the written exam (Klausur)

prerequisites for the module:
none

Recommended prior knowledge: Basic knowledge in Software Engineering, such as gained, e.g., in the module "Foundations of Software Engineering" (SWT-FSE-B). In particular, good knowledge of the Unified Modeling Language (UML) is expected.	**Admission requirements:** none

Frequency: every winter semester	**Recommended semester:**	**Minimal Duration of the Module:** Semester

Module Units

1. Software Quality **Mode of Delivery:** Lectures **Lecturers:** Prof. Dr. Gerald Lüttgen, Alexander Kraas **Language:** English **Frequency:** every winter semester **Learning outcome:** – see the module's learning outcomes/competences (Lernziele/Kompetenzen) listed above –	**2,00 Weekly Contact Hours**

Contents: The following topics will be covered in this module: • Software quality within agile software engineering • Fundamental testing concepts and techniques • Automated, model-based testing • Inspections and reviews • Software measurement • Static analysis • Software quality management **Literature:** • Goericke, S. (editor). The Future of Software Quality Assurance. Springer, 2020. • Kramer, A. and Legeard, B. Model-Based Testing Essentials. Wiley, 2016. • Meyers, G. J. et al. The Art of Software Testing, 3rd ed. Wiley, 2012. • O’Reagan, G. Concise Guide to Software Testing. Springer, 2019. • O’Reagan, G. Introduction to Software Quality. Springer, 2014. • Utting, M. and Legeard, B. Practical Model-Based Testing – A Tools Approach. Morgan Kaufmann, 2007. • Walkinshow, N. Software Quality Assurance. Springer, 2017.	
2. Software Quality **Mode of Delivery:** Practicals **Lecturers:** Scientific Staff Praktische Informatik, insbesondere Softwaretechnik und Programmiersprachen **Language:** English **Frequency:** every winter semester **Learning outcome:** – see the module's learning outcomes/competences (Lernziele/Kompetenzen) listed above – **Literature:** – see the corresponding lectures –	**2,00 Weekly Contact Hours**

Examination Written examination / Duration of Examination: 90 minutes **Description:** Written exam (Klausur) consisting of questions that relate to the contents of the lectures (Vorlesungen) and practicals (Übungen) of this module. The exam is passed if at least 50% of the available points are reached.	

ID	Module	Semester	ECTS	Weekly Contact Hours	Examination
	International studies taught in English (on demand) **Subject Group: Applied Computer Science**				
	Subject: Smart Environments				
AI-KI-B	Introduction to Artificial Intelligence	every summer semester(1)	6	2 Lectures 2 Practicals	105 minutes
SME-Projekt-B	Bachelor's project on Smart Environments	every semester	6	4 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
SME-Projekt-M	master project on smart environments	every summer semester(2)	6	4 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
SME-STE-M	Introduction to Knowledge Representation: Space, Time, Events	every winter semester	6	2 Lectures 2 Practicals	Oral examination 20 minutes
	Subject: Cognitive Systems				
KogSys-ML-M	Machine Learning	every winter semester	6	2 Lectures 2 Practicals	Written examination 105 minutes
	Subject: Cultural Computing				
KInf-MobAss-M	Mobile Assistance Systems	every summer semester	6	2 Practicals 2 Lectures	Colloquium 20 minutes Written examination 60 minutes
KInf-SemInf-M	Semantic Information Processing	every winter semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
	Subject: Human-Computer Interaction				

HCI-Prop-M	Propaedeutic: Human-Computer-Interaction	every winter semester(1)	3	2 Seminar	Coursework Assignment with presentation 4 months 30 minutes
HCI-MCI-M	Human-Computer Interaction	every summer semester	6	2 Lectures 2 Practicals	Oral examination Written examination 90 minutes
HCI-KS-B	Cooperative Systems	every summer semester	6	2 Lectures 2 Practicals	Oral examination Written examination 90 minutes
HCI-Proj-B	Project Human-Computer Interaction	every winter semester	6	4 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
HCI-Sem-B	Bachelor-Seminar Human-Computer Interaction	every summer semester	3	2 Seminar	Coursework Assignment with presentation 4 months 30 minutes
HCI-Usab-M	Usability in Practice	every summer semester	6	4 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
HCI-Proj-M	Project Human-Computer Interaction	every summer semester	6	4 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
HCI-DISTP-B	Design of Interactive Systems: Theory and Practice	every summer semester(1)	3	1 Lectures and Practicals	Colloquium 30 minutes

HCI-Proj1-M	Research-Project Human-Computer Interaction	every summer semester(WS 2016/2017)	15	6 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
HCI-Proj2-M	Research-Project Human-Computer Interaction	every winter semester(WS 2016/2017)	15	6 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
	Subject Group: Computer Science				
	Subject: Privacy and Security in Information Systems Group				
PSI-AdvaSP-M	Advanced Security and Privacy	every summer semester(1)	6	2 Lectures 2 Practicals	Written examination 90 minutes
PSI-EDS-B	Ethics for the Digital Society	every winter semester	3	2 Lectures	Written examination 60 minutes
PSI-IntroSP-B	Introduction to Security and Privacy	every winter semester(1)	6	2 Lectures 2 Practicals	Written examination 90 minutes
PSI-ProjectCAD-M	Project Complex Attacks and Defenses	every semester(1)	9	6 Practicals	Coursework Assignment and Colloquium 3 months 30 minutes
PSI-ProjectPAD	Project Practical Attacks and Defenses	every semester(1)	6	4 Practicals	Coursework Assignment and Colloquium 3 months 30 minutes
PSI-ProjectSP-M	Project Security and Privacy	every semester(1)	6	6 Practicals	Coursework Assignment and Colloquium 3 months 30 minutes

PSI-SSSProject-B	Software Systems Science Project: Security and Privacy	every semester(1)	12	8 Practicals	Coursework Assignment 3 months Coursework Assignment and Colloquium 3 months 30 minutes
PSI-Sem-B	Seminar Security and Privacy Foundations	every summer semester(1)	3	2 Seminar	Coursework Assignment with presentation 3 months 30 minutes
PSI-Sem-M	Seminar Research Topics in Security and Privacy	every winter semester(1)	3	2 Seminar	Coursework Assignment with presentation 3 months 30 minutes
	Subject: Communication Services, Telecommunication Systems, and Computer Networks				
KTR-GIK-M	Foundations of Internet Communication	every summer semester(on demand also WS)	6	4 Lectures and Practicals	Coursework Assignment and Colloquium 4 months 30 minutes
KTR-MAKV-M	Modeling and Analysis of Communication Networks and Distributed Systems	every summer semester	6	4 Lectures and Practicals	Oral examination 30 minutes
KTR-MMK-M	Multimedia Communication in High Speed Networks	every summer semester	6	4 Lectures and Practicals	Oral examination 30 minutes
KTR-Mobi-M	Mobile Communication	every winter semester	6	4 Lectures and Practicals	Oral examination 30 minutes

KTR-Proj	Project Communication Networks and Services	every winter semester(nach Bedarf auch SS)	6	4	Coursework Assignment and Colloquium 4 months 30 minutes
KTR-Sem-M	Master Seminar Communication Systems and Computer Networks	winter or summer semester, on demand(Regelturnus: WS)	3	2 Advanced seminar	Coursework Assignment with presentation 4 months 40 minutes
KTR-Sem-B	Bachelor Seminar Communication Systems and Computer Networks	winter and summer semester, on demand(Regelturnus: SS)	3	2 Seminar	Coursework Assignment with presentation 4 months 30 minutes
KTR-SSSProj-B	KTR Bachelor Project Software Systems Science	every semester(Turnusbeginn SS)	12	8	Coursework Assignment 4 months Coursework Assignment and Colloquium 4 months 30 minutes
KTR-SSSProj-M	KTR Master Project Software Systems Science	every semester(Beginn WS)	9	6	Coursework Assignment and Colloquium 4 months 30 minutes

Subject: Distributed Systems Group

DSG-DistrSys-M	Distributed Systems	every summer semester(2020)	6	2 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 months 15 minutes

DSG-IDistrSys-B	Introduction to Distributed Systems	every summer semester(2020)	6	2 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 months 10 minutes
DSG-Sem-M	Master Seminar in Distributed Systems	every semester	3	2 Key competence	Coursework Assignment with presentation 4 months 30 minutes
DSG-Project-B	Bachelor Project in Distributed Systems	every summer semester	6	4 Practicals	Coursework Assignment and Colloquium 2 months 10 minutes
DSG-DSAM-M	Distributed Systems Architectures and Middleware	every winter semester	6	2 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 months 15 minutes
DSG-Project-M	Master Project Distributed Systems	every semester	9	6 Practicals	Coursework Assignment and Colloquium 3 months 10 minutes
DSG-SOA-M	Service-Oriented Architecture and Web Services	every summer semester	6	2 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 months 15 minutes
	Subject: Foundations of Computer Science				
GdI-CSNL-M	Computational Semantics of Natural Language	every summer semester(1)	6	4	Portfolio 45 minutes
GdI-FPRS-M	Functional Programming of Reactive Systems		6	2 Lectures 2 Practicals	Written examination 90 minutes

		every summer semester			Oral examination 30 minutes
GdI-IFP-B	Introduction to Functional Programming	every winter semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
GdI-MTL	Modal and Temporal Logic	every winter semester	6	4 Lectures and Practicals	Written examination 90 minutes Oral examination
GdI-GTI-B	Machines and Languages	every summer semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
	Subject: Mobile Software Systems /Mobility				
MOBI-ADM-M	Advanced Data Management	every summer semester(1)	6	2 Lectures 2 Practicals	Written examination 75 minutes
MOBI-DSC-M	Data Streams and Complex Event Processing	every winter semester(1)	6	2 Lectures 2 Practicals	Oral examination 15 minutes Written examination 60 minutes
MOBI-MSS-B	Mobility in Software Systems	every winter semester(1)	6	4 Lectures and Practicals	Written examination 105 minutes
MOBI-PRAI-B	Bachelor Project Mobile Software Systems (AI)	every summer semester(1)	6	4 Practicals	Coursework Assignment and Colloquium
MOBI-PRAI-M	Master Project Mobile Software Systems (AI)	every winter semester(1)	6	4 Practicals	Coursework Assignment and Colloquium
MOBI-PRS-B	Bachelor Project Mobile Software Systems (SoSySc)	every semester(1)	12	8 Practicals	Coursework Assignment and Colloquium
MOBI-PRS-M	Master Project Mobile Software Systems (SoSySc)		9	6 Practicals	Coursework Assignment and Colloquium

		every summer semester(1)			
	Subject: Software Technologies Research Group				
SWT-FSA-B	Foundations of Software Analysis	every winter semester	6	4 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 weeks 20 minutes
SWT-PR1-M	Masters Project in Software Engineering and Programming Languages	every semester	6	4 Practicals	Coursework Assignment and Colloquium 12 weeks 20 minutes
SWT-PR2-M	SWT Masters Project in Software Systems Science	every semester	9	6 Practicals	Coursework Assignment and Colloquium 12 weeks 30 minutes
SWT-SWQ-M	Software Quality	every winter semester(1)	6	2 Lectures 2 Practicals	Written examination 90 minutes
SWT-FSE-B	Foundations of Software Engineering	every summer semester	6	3 Lectures 3 Practicals	Written examination 120 minutes
SWT-SEM-B	Seminar in Software Engineering and Programming Languages (Bachelor)	every semester	3	2 Seminar	Coursework Assignment with presentation 8 weeks 40 minutes
SWT-SWL-B	Software Engineering Lab	every winter semester	6	4 Practicals	Coursework Assignment and Colloquium 2 weeks 45 minutes

SWT-PCC-M	Principles of Compiler Construction	every summer semester	6	2 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 weeks 20 minutes
SWT-ASV-M	Applied Software Verification	every summer semester	6	2 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 weeks 20 minutes
SWT-SEM-M	Seminar in Software Engineering and Programming Languages (Master)	every semester	3	2 Seminar	Coursework Assignment with presentation 8 weeks 40 minutes
SWT-RSD-B	Reactive Systems Design	every summer semester(1)	6	2 Lectures 2 Practicals	Coursework Assignment and Colloquium 3 weeks 20 minutes
	Subject Group: Information Systems				
	Subject: Information Systems Management				
ISM-FIISM-B	Fundamentals of International IS Management	every summer semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
	Subject: Energy Efficient Systems				
EESYS-ADAML-M	Applied Data Analytics and Machine Learning in R	every winter semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
EESYS-BIA-M	Business Intelligence & Analytics	every winter semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
EESYS-ES-M	Energy Efficient Systems	every summer semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
EESYS-IITP-B	International IT Project Management		6	2 Lectures	Written examination

		every summer semester		2 Practicals	90 minutes
	Subject: Information Systems and Services				
	Subject: Social Networks				
SNA-OSN-M	Project Online Social Networks	every winter semester	6	4 Practicals	Coursework Assignment and Colloquium 4 months 30 minutes

ID	Module	Semester	ECTS	Weekly Contact Hours	Examination
	Course language German, exams in English on demand, course material may be available in English **Subject Group: Applied Computer Science**				
	Subject: Media Informatics				
	Subject: Smart Environments				
SME-Sem-B	Bachelor seminar on Smart Environments	every winter semester(1)	3	2 Seminar	Internship report 4 months 30 minutes
SME-Sem-M	master seminar on Smart Environments	every summer semester(1)	3	2 Seminar	Coursework Assignment with presentation 4 months 30 minutes
	Subject Group: Computer Science				
	Subject: Distributed Systems Group				
DSG-Project-2-SoSySc-B	DSG Bachelorproject Software Systems Science	every winter semester	12	8 Practicals	Coursework Assignment 4 months Coursework Assignment and Colloquium 4 months 10 minutes
DSG-Project-B	Bachelor Project in Distributed Systems	every summer semester	6	4 Practicals	Coursework Assignment and Colloquium 2 months 10 minutes
DSG-Sem-B	Bachelor Seminar in Practical Computer Science	every semester	3	2 Introductory seminar	Coursework Assignment with presentation 4 months 20 minutes
	Subject: Foundations of Computer Science				

GdI-MfI-1	Propositional and Predicate Logic	every winter semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
	Subject Group: Information Systems				
	Subject: Information Systems Management				
ISM-IOM-M	International Outsourcing Management	every winter semester	6	4	Written examination 90 minutes
	Subject: Energy Efficient Systems				
	Subject: Information Systems and Services				
ISDL-DEXP-B	Digital Experimentation	every winter semester	6	2 Lectures and Practicals	Written examination 90 minutes
	Subject: Industrial Information Systems				
IIS-Sem-B	Bachelor Seminar Industrial Information Systems	every winter semester	3	2 Introductory seminar	Coursework Assignment with presentation 3 months 30 minutes
IIS-Sem-M	Master Seminar Industrial Information Systems	every winter semester	3	2 Introductory seminar	Coursework Assignment with presentation 3 months 30 minutes
	Subject: Social Networks				
SNA-WIM-B	Knowledge- and Informationmanagement	every summer semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
SNA-ASN-M	Social Network Analysis	every winter semester	6	2 Lectures 2 Practicals	Written examination 90 minutes
SNA-NET-M	Network Theory	every summer semester	6	2 Lectures 2 Practicals	Written examination 90 minutes