

Technology
Arts Sciences
TH Köln

Faculty of Information, Media and Electrical Engineering

Bachelor Computer Science and Engineering 2020

Module Manual

Version: 3.9.2025-09-01-19-38-32

The most recent version of this handbook can be found here:

<https://f07-studieninfo.web.th-koeln.de/mhb/current/en/BaTIN2020.html>

1. Program Description

The Bachelor's degree course in Computer Science and Engineering provides students with in-depth knowledge in the development and application of complex information technology systems. Students learn to effectively combine software and hardware in order to create innovative solutions for current technological challenges. One conceivable example is the development of a system that processes and analyzes sensor data in a smart home. Computer science and systems engineering are key skills of the 21st century.

Orientation of the degree program

The course is a computer science degree program with a standard duration of seven semesters. It offers a broad range of topics from which students can choose one of three specializations:

In the "Distributed Software Systems" specialization, students focus on the architecture and implementation of complex systems such as cloud applications and networks. "Computer Engineering" focuses on the design and implementation of embedded systems, which are used, for example, in the Internet of Things, automation and robotics. The specialization "Artificial Intelligence" deals with the development of algorithms for machine learning and data processing.

In addition, modules are offered in the fields of communications engineering, acoustics and electrical engineering.

Occupational fields

Graduates of this degree course find exciting career opportunities in numerous technological fields. They are in demand in software development, where they develop tailor-made applications and systems for a wide range of industries. In the IT sector and in the development of intelligent technologies in areas such as medical technology or energy technology, embedded systems specialists find a wide range of employment opportunities. Graduates in the field of artificial intelligence are sought-after specialists who develop algorithms for machine learning and data-based decision-making. In telecommunications, they use their knowledge of networks and data transmission to improve the efficiency and security of communication systems.

The skills taught on the course make our graduates valuable team members in start-ups and established companies alike.

Course of study

At the beginning of the course, the focus is on the fundamentals of computer science in the areas of software and hardware as well as mathematics, with practical exercises and examinations consolidating the theoretical knowledge. In the following semesters, the topics expand to include advanced content such as operating systems, networks, IT security and software engineering, with a particular focus on the systematic approach to the development and optimization of technical systems. Numerous elective modules offer the opportunity to concentrate on special interests. Practical skills are intensively promoted through project work and internships. The course concludes with a practical phase and the Bachelor's thesis. This structured course provides students with both theoretical knowledge and practical experience with a particular focus on the application-oriented integration of various technologies, which are essential for success in IT and technology-related professions.

Study abroad

We actively support our students in integrating a period of study at a foreign university. The credits earned there are recognized for German studies. To this end, we have close contacts with universities in several European and non-European countries. The fifth semester can be completed at one of these universities. Our expectations of applicants If you would like to know how a complex application system, a modern operating system, a mobile application or data networks work, then a degree in Computer Engineering could suit you. The basis for admission to the course is an interest in computer science and technical subjects. It is important that you are not content with superficial descriptions, but want to get to the bottom of things.

Study requirements

Advanced technical college entrance qualification (school and practical part) or Abitur or equivalent qualification.

2. Graduate Profile

Graduates of the B. Sc. Computer Science and Engineering degree program have a sound knowledge of computer science, software development and technology-related system integration. They are able to design, develop, analyze and operate complex information technology systems. Their interdisciplinary knowledge, coupled with practical project work, qualifies them for a wide range of tasks in software and system development, depending on their choice of specialization, particularly in the areas of embedded systems, distributed applications and artificial intelligence.

The aim of the Bachelor's degree program in Computer Science and Engineering is to train students to become reflective specialists who are able to design, implement and further develop modern information technology systems in their entire breadth -- from hardware-related programming to system-related software and networked, intelligent applications.

Graduates acquire basic and advanced skills in computer science, mathematics, communication technology and electrical engineering. They can choose one of three specializations:

- Embedded Systems
- Internet of Things
- Networks and Distributed Systems
- Smart Systems
- Software Systems

Methodological, technical and social skills are promoted in equal measure through project-oriented study components, practical teaching formats and an in-depth examination of real fields of application. Graduates are familiar with modern development methods, agile processes and interdisciplinary teamwork.

Graduates of the Bachelor's degree program in Computer Science and Engineering develop their individual profile in the following areas:

- They are able to analyze, model and design complex software and hardware systems ready for implementation - especially in safety-critical and distributed system environments.
- You have application-oriented skills in programming, system specification, simulation, verification and quality assurance of IT systems.
- They are proficient in the design and operation of distributed systems, cloud applications, databases, embedded systems and artificial intelligence components.
- Through project-based training, they learn to quickly familiarize themselves with new technologies and efficiently implement them in practice as part of a team - even in intercultural contexts.
- They are able to reflect on social, ethical and legal issues and shape their technical work in a sustainable and responsible manner.
- With their communicative, interdisciplinary and organizational skills, they can take on technical responsibility and aspire to management roles in both start-ups and large companies.

The course qualifies students for direct career entry in areas such as software development, embedded systems, automation technology, telecommunications and data science as well as for a subsequent Master's degree. However, you can also work in technical sales, training, technical customer consulting and project management (e.g. as a product owner or scrum master). Computer scientists are employed in these roles in many sectors: at software manufacturers, development companies, in information and communication technology, at consulting firms or public authorities.

3. Fields of Action

Central fields of activity in the degree program are development and design, research and innovation, leadership and management as well as quality assurance and testing. The profile module matrix shows which fields of activity are addressed by which modules.

Research and development

This area covers the research and development of new technologies, algorithms, processes, devices, components and systems. This includes basic and industrial research as well as more specialized development such as in media technology, optometry, electrical engineering and technical computer science.

System and process management

This includes the planning, design, monitoring, operation and maintenance of systems and processes. This also includes the management of production processes, quality assurance and the coordination of working groups as well as IT administration and project management.

Innovation and application

The design, development and use of innovative applications and systems in technical disciplines. This also includes the creation and design of media content and products, the development of electronic, IT, media technology, acoustic or optical components and systems as well as the integration of IT solutions in technical applications.

Analysis, evaluation and quality assurance

The analysis and evaluation of procedures, systems, algorithms and devices to ensure the quality of products and processes. Includes the reflection and evaluation of media content and clinical studies as well as the investigation of visual and acoustic perception processes.

Interaction and communication

The ability for interdisciplinary collaboration and mediation between designers, technical actors, clients and users. Emphasizes the importance of soft skills such as teamwork and presentation skills in technical professions.

4. Competencies

The modules of the degree program train students in different competencies, which are described below. The profile module matrix shows which competencies are addressed by which modules.

Systems thinking and delimitation of system boundaries

Understanding and identifying the boundaries of different systems, including the delineation of relevant aspects from external, uninfluenceable factors.

Abstraction and modeling

Ability to simplify and generalize complex problems, develop and evaluate different models across disciplines.

Analyze natural and technical phenomena

Identification, naming and explanation of relevant phenomena in real-life scenarios, including scientific principles and technical contexts.

STEM competence

Knowledge and application of models and principles from mathematics, computer science, natural sciences and technology for problem solving.

Simulation and analysis of technical systems

Use of software and tools to simulate and analyze technical systems, including the development of simulation models.

Design and realization of systems and processes

Design and implementation of technical solutions and processes, taking into account technical, economic and ecological standards and principles.

Testing and evaluating systems and processes

Performing tests, including verification and validation, to ensure compliance with standards and the functionality of systems and economic aspects of processes.

Obtaining and evaluating information

Ability to systematically research, analyze and evaluate information including relevant contexts.

Communication and presentation

Effective presentation and explanation of complex technical content to different target groups in German and English.

Business and legal knowledge

Apply basic business and legal knowledge to technical and design projects and decisions.

Teamwork and interdisciplinary cooperation

Ability to work in teams, including effective communication and cooperation with professionals from other disciplines.

Decision-making in uncertain situations

Strategic decision making based on sound professional analysis, even under uncertainty.

Consideration of social and ethical values

Integration of ethical and social values in the design of systems and media and reflection on professional actions.

Learning competence and adaptability

Motivation and ability to engage in lifelong learning and to adapt to technological and methodological innovations.

Self-organization and self-reflection

Competence in the self-organization of professional and learning-related tasks as well as critical reflection of one's own actions.

Communicative and intercultural skills

Effective communication and cooperation in intercultural and international contexts as well as media skills.

Specific professional knowledge and skills

In-depth knowledge and skills tailored to the requirements and specifics of individual subject areas such as electrical engineering, media technology, optometry and computer engineering.

5. Study Plans

The following are studyable study plans. Other study plans are also possible. Please note, however, that each module is usually only offered once a year. Please also note that several modules may have to be selected in a particular semester and elective catalogs in order to obtain the total ECTS credit points shown.

5.1 Studienverlaufsplan

Sem.	Abbr.	Module Name	Mandatory (PF) Elective Catalog (WB)	ECTS
1	DR	Digitalrechner	PF	5
	EG	Elektrotechnische Grundlagen für die Technische Informatik	PF	5
	MA1	Mathematik 1	PF	10
	PI1	Praktische Informatik 1	PF	5
	PP	Programmierpraktikum	PF	5
2	FSA	Formale Sprachen und Automatentheorie	PF	5
	AD	Algorithmen und Datenstrukturen	PF	5
	MA2	Mathematik 2	PF	10
	PI2	Praktische Informatik 2	PF	5
	GSP	Grundlagen der Systemprogrammierung	PF	5
3	BVS1	Betriebssysteme	PF	6
	GUI	Graphische Oberflächen und Interaktion	PF	5
	DB1	Datenbanken 1	PF	5
	NP	Netze und Protokolle	PF	5
	SE	Software Engineering	PF	5
	SIG	Signalverarbeitung	PF	5
4	BVS2	Verteilte Systeme	PF	5
	BWR	Betriebswirtschaft und Recht	PF	5
	ITS	IT-Sicherheit	PF	5
	SWP	Softwarepraktikum	PF	6
	WM	Wahlbereich Informatik und Systems-Engineering	WB	10
5	SYP	Systementwicklungs-Projekt	PF	7
	PUK	Präsentation und Kommunikation	PF	3
	XIB	Fachübergreifende Kompetenzen und Soft-Skills	PF	5
	WM	Wahlbereich Informatik und Systems-Engineering	WB	15
6	IPM	IT-Projektmanagement	PF	5
	PRA	Praxisphase	PF	8
	WM	Wahlbereich Informatik und Systems-Engineering	WB	15
7	PRA	Praxisphase	PF	15
	BAA	Bachelorarbeit	PF	12
	KOLL	Kolloquium zur Bachelorarbeit	PF	3

5.2 Alternativer Studienverlaufsplan (verminderter Workload)

Sem.	Abbr.	Module Name	Mandatory (PF) Elective Catalog (WB)	ECTS
1	MA1	Mathematik 1	PF	10
	PI1	Praktische Informatik 1	PF	5
	PP	Programmierpraktikum	PF	5
2	MA2	Mathematik 2	PF	10
	PI2	Praktische Informatik 2	PF	5
	GSP	Grundlagen der Systemprogrammierung	PF	5
3	BVS1	Betriebssysteme	PF	6
	GUI	Graphische Oberflächen und Interaktion	PF	5
	DR	Digitalrechner	PF	5
	EG	Elektrotechnische Grundlagen für die Technische Informatik	PF	5
4	BVS2	Verteilte Systeme	PF	5
	FSA	Formale Sprachen und Automatentheorie	PF	5
	ITS	IT-Sicherheit	PF	5
	AD	Algorithmen und Datenstrukturen	PF	5
5	DB1	Datenbanken 1	PF	5
	SE	Software Engineering	PF	5
	SIG	Signalverarbeitung	PF	5
	NP	Netze und Protokolle	PF	5
6	SWP	Softwarepraktikum	PF	6
	BWR	Betriebswirtschaft und Recht	PF	5
	WM	Wahlbereich Informatik und Systems-Engineering	WB	10
7	SYP	Systementwicklungs-Projekt	PF	7
	PUK	Präsentation und Kommunikation	PF	3
	XIB1	Fachübergreifende Kompetenzen und Soft Skills 1	WB	5
	WM	Wahlbereich Informatik und Systems-Engineering	WB	5
8	IPM	IT-Projektmanagement	PF	5
	WM	Wahlbereich Informatik und Systems-Engineering	WB	15
9	PRA	Praxisphase	PF	8
	WM	Wahlbereich Informatik und Systems-Engineering	WB	10
10	PRA	Praxisphase	PF	15
	BAA	Bachelorarbeit	PF	12
	KOLL	Kolloquium zur Bachelorarbeit	PF	3

6. Modules

The modules of the degree program are described below in alphabetical order.

6.1 AD - Algorithmen und Datenstrukturen

Module ID	AD_BaTIN2020
Module Name	Algorithmen und Datenstrukturen
Type of Module	Mandatory Module
Recognized Course	AD - algorithms and data structures
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt Kenntnisse zum Umgang mit Algorithmen und Datenstrukturen sowie ein tieferes Verständnis der zugrundeliegenden Konzepte und Techniken. Im Fokus dieses steht die exemplarische Nutzung der Programmierung in C++ zur Realisierung der behandelten Datenstrukturen und Algorithmen. In praktischer Arbeit analysieren die Studierenden Problemstellungen im Systemumfeld (K1, K2, K4), implementieren Lösungen auf der Grundlage anerkannter Konzepte und Methoden (K3) mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7). Sie recherchieren dazu in Dokumentationen (K8, K15) und passen vorhandene Software an (K10).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K8, K13, K16).

Wozu: Die Verwendung und Beurteilung von Standarddatenstrukturen und Standardalgorithmen ist essentiell für heutige komplexe Softwaresysteme. Entsprechende Programmierkenntnisse und Wissen über die zugehörigen Grundlagen sind somit unverzichtbar für die Erstellung moderner Software (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2) sowie zur Ausführung von Programmen (HF3). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents**Lecture / Exercises**

concepts of object oriented programming
 basics of linear and hierarchic data structures used in programs
 linear data structures (e.g. linear lists chained lists)
 hierarchic data structures, trees
 analysis of the complexity of algorithms
 Important search algorithms
 mode of operation of important sort algorithms
 apply the concepts of object oriented programming in C++
 estimate the quality and useness of data structures and algorithms
 implement search and sort algorithms in programs

Lab

develop single handed data structures and algorithms
 implement linear and hierarchic data structures in C++
 implement sort algorithms in C++
 implement search algorithms in C++
 application of the aspects listed above to real-world scenarios in small teams

Teaching and Learning ■ Lecture / Exercises
Methods ■ Lab

Examination Types with cf. exam regulations
Weights

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended ■ Modul PI1: Sicherer Umgang mit einer Programmiersprache.
Prerequisites ■ basics of programming in high level programming languages

Mandatory Lab requires attendance in the amount of: 3 Praktikumstermine
Prerequisites

Recommended ■ Sedgewick, Robert: Algorithmen in C
Literature ■ Ottmann, Widmayer: Algorithmen und Datenstrukturen
 ■ Heun: Grundlegende Algorithmen
 ■ Wirth, Niklaus: Algorithmen und Datenstrukturen
 ■ Elektronische Verweise auf ebooks und Online Tutorials

Use of the Module in AD in Bachelor Informatik und Systems-Engineering 2024
Other Study Programs

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.2 ASN - Angewandte Statistik und Numerik

Module ID	ASN_BaTIN2020
Module Name	Angewandte Statistik und Numerik
Type of Module	Elective Modules
Recognized Course	ASN - Applied Statistics and Numerical Analysis
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Holger Weigand/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Holger Weigand/Professor Fakultät IME
Learning Outcome(s)	
Was: Das Modul vermittelt die Kompetenz, mathematische Modelle zur Beschreibung technischer Systeme zu entwerfen (K2, K5, K11), diese effizient zu implementieren und ihre Grenzen zu benennen (K1, K19). Der Studierende kann Informationen mathematisch aus- und bewerten (K12). Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs/Übungsteil und betreut parallel dazu ein Praktikum, in dem die Studierenden bekannte und selbst entwickelte Algorithmen implementieren. Wozu: Die erworbenen Kompetenzen unterstützen den Studierenden bei der Entwicklung von Algorithmen für die Forschung (HF 1). Er kann die Güte von Algorithmen bei größeren technischer Systemen abschätzen bzw. sie in solchen Systemen realisieren (HF2). Bei der Planung und Realisierung von Systemen zur Verarbeitung von Informationen für technische Anwendungen (HF3) kann er abstrakte Modelle entwerfen, speziell bei Berechnungssystemen.	
Module Contents	
Lecture / Exercises	
to do	
Lab	
Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	▪ Modul MA1: Grundlegende mathematische Kenntnisse, insbesondere Funktionen und Differentialrechnung anwenden ▪ Modul MA2: Methoden der linearen Algebra anwenden können ▪ Modul PI1: Grundbegriffe der Programmierung anwenden ▪ to do
Mandatory Prerequisites	▪ Participation in final examination only after successful participation in Lecture / Exercises ▪ Lab requires attendance in the amount of: 4 Termine

Recommended Literature	▪ Knorrenschild: Numerische Mathematik (Fachbuchverlag) ▪ Papula: Mathematik für Ingenieure und Naturwissenschaftler, Band 1+2 (Vieweg)
-------------------------------	---

Use of the Module in Other Study Programs	▪ AM in Bachelor Elektrotechnik 2020 ▪ ASN in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ ASN in Bachelor Informatik und Systems-Engineering 2024
--	--

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.3 AT - Antennentechnik

Module ID	AT_BaTIN2020
Module Name	Antennentechnik
Type of Module	Elective Modules
Recognized Course	AT - Antenna Technology
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Rainer Kronberger/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Rainer Kronberger/Professor Fakultät IME
Learning Outcome(s)	
Die Studierenden können spezielle elektromagnetische Probleme lösen, indem sie hierfür geeignete spezielle Methoden der Hochfrequenztechnik und Elektrotechnik anwenden, um später Antennen für hochfrequente Anlagen, Systeme und Baugruppen zu analysieren, entwickeln und herzustellen	
Module Contents	
Lecture	
• electromagnetic fields and waves, Maxwell Equations • wave propagation, polarization, reflection and transmission of waves on materials and boundaries - pattern, directivity, gain, impedance, efficiency - linear antennas, monopole, dipole, arrays, Yagi-Uda an-tenna, parabolic antenna - mobile antennas and terminal antennas - antenna measurements, nearfield, farfield, understand electromagnetic waves Ability to use simulation tools Learn measurment methodes and methodologies Undertand the functional principle of antennas	
Project	
Undertstand measurements at higher frequencies Learn how to use simulation tools Learn how to handle RF measurement equipment Learn how to perform antenna measurements Perform antenna simulations Write scientific reports	
Teaching and Learning Methods	▪ Lecture ▪ Project
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	34 Hours $\pm$ 3 SWS

Self-Study	116 Hours
Recommended Prerequisites	▪ Modul EG: Erweiterte Kenntnisse der Elektrotechnik und Grundlagen der Grundlegende Kenntnisse der Hochfrequenztechnik ▪ GE1, GE2, GE3, EL1, GHF, Mathematics
Mandatory Prerequisites	Project requires attendance in the amount of: 6 Termine
Recommended Literature	▪ Meinke/ Gundlach: Taschenbuch der Hochfrequenztechnik Bd. 1-3 Springer Verlag ▪ Detlefsen/Siart: Grundlagen der HF-Technik. Oldenbourg Verlag ▪ Zinke/ Brunswig: Hochfrequenztechnik 1, Filter, Leitungen, Anten-nen, Springer Verlag ▪ Kark: Antennen und Strahlungsfelder , Springer Verlag
Use of the Module in Other Study Programs	▪ AT in Bachelor Elektrotechnik 2020 ▪ AT in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ AT in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.4 ATS - Autonome Systeme

Module ID	ATS_BaTIN2020
Module Name	Autonome Systeme
Type of Module	Elective Modules
Recognized Course	ATS - Autonomous Systems
ECTS credits	5
Language	deutsch und englisch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Chunrong Yuan/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Chunrong Yuan/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt Kompetenzen zur Erstellung von autonomen Systemen (AS) in allen relevanten Aspekten und Arbeitsschritten von der Auslegung und Planung des gesamten Systems (K.1, K.4), Auswahl und Bewertung der Komponenten (K.8, K.9), Entwicklung der Software für die Sensordatenverarbeitung und intelligente Robotersteuerung unter der Verwendung von Methoden wie z.B. KI (Künstliche Intelligenz) und Robotersehen (K.2, K.3), die Integration von Software und Hardware Komponenten (K.5, K.6), bis zur Inbetriebnahme und Validierung des gesamten robotischen Systems (K.7, K.10).

Womit: Die Dozentin vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs/Übungsteil und betreut parallel dazu Miniprojekte, in denen die Studierenden ihre Kenntnisse anwenden und relevante Komponenten für AS entwickeln.

Wozu: Kompetenzen in der Entwicklung eines AS sind essentiell für technische Informatiker*innen, die im HF1 arbeiten wollen. Durch das Erlernen und die Anwendung von aktuellen Methoden und Techniken im Bereich KI und Robotik anhand robotischer Plattformen erwerben die Studierenden zudem Erfahrungen, die essentiell für das HF2 sind, u.a. Anforderungen erfassen, Konzepte zur technischen Lösung entwickeln und diese zu bewerten.

Module Contents
Lecture

Sensors
 Wheel/motor sensors
 Heading sensors
 Positioning sensors
 Cameras
 Locomotion
 Wheeled mobile robots
 Legged mobile robots
 Data processing and feature extraction
 Edge detection
 Line extraction
 Point detection and description
 Recognition and Modelling
 Object detection
 Place recognition
 3D motion and structure estimation
 Navigation
 Localization
 Mapping
 Path planning

Lab

Teamwork: Development of systems with intelligent behaviours for autonomous interpretation of sensor data and real-time robot control. The goal is to realize prototypes with the required functions.

Exercises

Sensor characterization
 Feature extraction
 Image matching and clustering
 Image based place recognition
 Motion analysis
 Programming of robot behaviour

Teaching and Learning	▪ Lecture
Methods	▪ Lab
	▪ Exercises

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites	▪ Modul AD: Kompetenz in der Analyse und Realisierung von Algorithmen ▪ Modul SIG: Kenntnisse in der Signalverarbeitung ▪ Modul SYP: Kompetenz in der Entwicklung von Software und Projekten ▪ Modul ES: Grundkenntnisse in der hardwarenahe Softwareentwicklung ▪ Capability of algorithm analysis and implementation - Knowledge of signal processing and mathematics - Capability of software and project development - Basic knowledge of embedded software
----------------------------------	--

Mandatory Prerequisites	Lab requires attendance in the amount of: 1 Termin
Recommended Literature	▪ Hertzberg: Mobile Roboter: Eine Einführung aus Sicht der Informatik, Springer Vieweg, 2012
Use of the Module in Other Study Programs	▪ ATS in Bachelor Elektrotechnik 2020 ▪ ATS in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ ATS in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.5 BAA - Bachelorarbeit

Module ID	BAA_BaTIN2020
Module Name	Bachelorarbeit
Type of Module	Mandatory Module
Recognized Course	BAA - Bachelor thesis
ECTS credits	12
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	7
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Bachelor Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	verschiedene Dozenten*innen / diverse lecturers

Learning Outcome(s)

Was: Das Modul vermittelt die Fähigkeit, eine Aufgabe der Informatik oder eine ingenieurwissenschaftliche Aufgabe selbstständig mit Hilfe wissenschaftlicher Methoden zu bearbeiten.

Womit: Auf Basis einer individuellen Vereinbarung des Studierenden mit einem Dozenten der F07 erhält der Studierende eine qualifizierte Aufgabenstellung aus dem Bereich der technischen Informatik die eigenständig und innerhalb einer beschränkten Frist erfolgreich bearbeitet und dokumentiert werden muss. Die Bachelorarbeit kann auch extern in einem Unternehmen durchgeführt werden.

Wozu: Das Berufsleben eines Informatikers ist geprägt durch die Bearbeitung komplexerer Aufgabenstellung aus dem Bereich der Informatik und/oder der Ingenieurwissenschaften. Hierbei stellt die vollumfängliche Bearbeitung (Recherche, Konzepterstellung, Implementierung, Verifikation, Validierung, Dokumentation) der Aufgabe eine wesentlichen Anforderung dar. Dies wird im Rahmen dieses Moduls quasi als "Einstieg in das Berufsleben" vermittelt.

Module Contents

Thesis

The Bachelor's thesis is a written assignment. It should show that the student is capable of independently working on a topic from his or her subject area within a specified period of time, both in its technical details and in its interdisciplinary contexts, using scientific and practical methods. Interdisciplinary cooperation can also be taken into account in the final thesis.

Teaching and Learning Methods

Examination Types with Weights

Workload 360 Hours

Contact Hours 0 Hours $\pm$ 0 SWS

Self-Study 360 Hours

Recommended Prerequisites

Mandatory Prerequisites See exam regulations §26 paragraph 1

Recommended Literature

Use of the Module in Other Study Programs	▪ BAA in Bachelor Elektrotechnik 2020 ▪ BAA in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ BAA in Bachelor Medientechnologie 2020 ▪ BAA in Bachelor Medientechnologie 2024 ▪ BAA in Bachelor Optometrie 2021 ▪ BAA in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	See also examination regulations §24ff. Contact a professor of the faculty early on for the initial supervision of the thesis.
Last Update	19.7.2025, 14:32:16

6.6 BVS1 - Betriebssysteme

Module ID	BVS1_BaTIN2020
Module Name	Betriebssysteme
Type of Module	Mandatory Module
Recognized Course	BVS1 - Operating Systems
ECTS credits	6
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every winter term
Module Coordinator	Studiengangsleiter(in) Bachelor Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	Prof. Dr. Andreas Behrend/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt Kompetenzen zum Umgang mit Betriebssystemen und Diensten in verteilten Systemen sowie ein tieferes Verständnis der zugrundeliegenden Konzepte und Techniken. Im Fokus dieses ersten Moduls (gefolgt von BVS2) steht die Nutzung von Programmier Techniken und -schnittstellen, die eine Systemsoftware typischerweise zur Realisierung nebenläufiger, kooperierender Software im lokalen und verteilten Umfeld anbietet. In praktischer Arbeit analysieren die Studierenden Problemstellungen im Systemumfeld (K1, K2, K4), implementieren Lösungen auf der Grundlage anerkannter Konzepte und Methoden (K3) mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7). Sie recherchieren dazu in Dokumentationen (K8, K15) und passen vorhandene Software an (K10).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K8, K13, K16).

Wozu: Systemsoftware, also Betriebssysteme und Dienstsoftware für verteilte Systeme, bietet die Plattform zur Erstellung von Anwendungen, die nebenläufig und verteilt arbeiten - Eigenschaften, die für heutige komplexe Softwaresysteme typisch sind. Entsprechende Programmierkenntnisse und Wissen über die zugehörigen Grundlagen sind somit essentiell für die Erstellung moderner Software (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2) sowie zur Organisation von Prozessen und zum Betrieb von Systemen, die nebenläufig und verteilt arbeiten (HF3). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents

Lecture / Exercises

- fundamentals of operating systems and distributed systems
- position and tasks of an operating system in a computer
- resources to be managed
- concurrency in hard- and software
- components and properties of distributed systems
- software structures
- operating system kernel
- hierarchical structures
- virtual machines
- client-server systems
- peer-to-peer systems
- the UNIX/Linux operating system
- history and standards
- layered structure
- kernel with programming interface
- shell with user interface
- fundamental user commands
- structure of the file system
- programming in C
- concurrency
- processes and threads
- fundamental properties
- processes in UNIX
- threads in Java
- synchronization
- fundamental conditions
- mutual exclusion
- sequencing
- mechanisms
- interrupt masking
- spinlocks
- signals
- semaphores
- monitors
- deadlocks
- communication
- fundamental terms
- storage-based vs. message-based communication
- mailboxes and ports
- synchronous vs. asynchronous communication
- local communication
- shared memory
- message queues
- pipes
- communication in distributed systems
- protocols
- sockets
- using the interfaces of an operating system:
 - user interface (console)
 - programming interface (API)
- controlling concurrent operations in an operating system
 - from the user interface
 - by API functions
- synchronizing concurrent operations by synchronization mechanisms

using various communication mechanisms
 local mechanisms
 mechanisms in computer networks

Lab

commands of the character-based Linux/UNIX command interface
 usage at the console
 usage in shell scripts
 esp. to control concurrent processes

C functions of the UNIX/Linux programming interface
 to access files and devices
 to start and control processes
 to synchronize processes
 to transfer data between processes (locally and in a network) - depending on available time

application of the aspects listed above to real-world scenarios in small teams

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights

cf. exam regulations

Workload 180 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 135 Hours

Recommended Prerequisites

- Modul PI1: Sicherer Umgang mit einer Programmiersprache.
- Modul PI2: Sicherer Umgang mit einer Programmiersprache.
- Modul DR: Kenntnisse über Aufbau und Funktionalität eines Digitalrechners.
- Modul GSP: Grundkenntnisse über die hardwarenahe Programmierung eines Digitalrechners.
- Modul NP: Grundkenntnisse über Internet-Protokolle.
- procedural programming
- architecture of a digital computer (basic knowledge)
- Internet protocols (basic knowledge)

Mandatory Prerequisites

Lab requires attendance in the amount of: 1 Termin

Recommended Literature

- Siehe http://www.nt.fh-koeln.de/vogt/bs/bvs_lit.pdf

Use of the Module in Other Study Programs

- BVS1 in Bachelor Elektrotechnik 2020
- BVS1 in Bachelor Elektrotechnik und Informationstechnik 2024
- BVS1 in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update 18.8.2025, 16:18:31

6.7 BVS2 - Verteilte Systeme

Module ID	BVS2_BaTIN2020
Module Name	Verteilte Systeme
Type of Module	Mandatory Module
Recognized Course	BVS2 - Distributed Systems
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	4
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME

Learning Outcome(s)

Was: Aufbauend auf BVS1 vermittelt das Modul vertiefend Kompetenzen zum Umgang mit Systemsoftware, insbesondere mit Diensten in verteilten Systemen. In praktischer Arbeit analysieren die Studierenden Problemstellungen im Systemumfeld (K1, K2, K4), implementieren Lösungen auf der Grundlage anerkannter Konzepte und Methoden (K3) mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7). Sie recherchieren dazu in Dokumentationen (K8, K15) und passen vorhandene Software an (K10).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K8, K13, K16).

Wozu: Systemsoftware, also Betriebssysteme und Dienstsoftware für verteilte Systeme, bietet die Plattform zur Erstellung von Anwendungen, die nebenläufig und verteilt arbeiten - Eigenschaften, die für heutige komplexe Softwaresysteme typisch sind. Entsprechende Programmierkenntnisse und Wissen über die zugehörigen Grundlagen sind somit essentiell für die Erstellung moderner Software (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2) sowie zur Organisation von Vorgängen und zum Betrieb von Systemen, die nebenläufig und verteilt arbeiten (HF3). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF4).

Was: Das Modul vermittelt Wissen über die Implementation von Systemsoftware auf Grundlage einer Hardwarearchitektur. Die Studierenden lernen die Details ihrer Realisierung sowie die zugrundeliegenden Techniken, Konzepte und Strategien kennen. Sie spielen in den Übungen typische Szenarien durch und lernen dabei die Auswirkungen strategischer Entscheidungen bei Entwurf, Implementierung und Ausführung der Systemsoftware kennen (K1, K2, K3, K4, K9).

Womit: Der Dozent vermittelt das grundlegende Wissen in der Vorlesung und leitet in den Übungen zu seiner Anwendung an.

Wozu: Die Systemsoftware ist ein zentraler Bestandteil eines jeden Rechensystems und somit entscheidend für seine Einsatzmöglichkeiten und Leistung. Entsprechendes Wissen über ihre Eigenschaften und mögliche Alternativen bei ihrer Realisierung ist daher essentiell für die Erstellung (HF1), Analyse (HF2) und Organisation (HF3) informationstechnischer Systeme.

Module Contents**Lecture / Exercises**

cooperation
client-server model
examples: naming and file services
layered architectures
peer-to-peer model
procedural cooperation: remote procedure call
object-oriented cooperation
remote method invocation
object-orientierte middleware
web-based services
dynamic web pages
web services

implementation of software concurrency
management of processes
dispatching and scheduling
exceptions and interrupts
storage concepts
components of the storage hierarchy
swapping
virtual storage
processes in distributed systems
load distribution, fault tolerance, synchronization

file systems
logical and real structures
local file systems
implementation of directories
organisation of the hard disk
performance enhancement and fault tolerance
distributed file systems
file server and name server
distributed directory trees
caching and replication

assess various strategies and techniques for processor scheduling, for storage hierarchy management and for the implementation of file systems in local and distributed environments

programming of and with services in local and distributed systems

Services in distributed systems
fundamentals of cloud computing and web services
Apache-based systems
commercially available systems

Lab

C functions of the UNIX/Linux programming interface to communicate and cooperate locally and in the Internet
by using shared memory, message queues, and sockets
by using Remote Procedure Call

Java techniques for communication and cooperation
web services: SOAP, REST
others as appropriate (to be determined on short notice)

application of the aspects listed above to real-world scenarios in small teams

Teaching and Learning	▪ Lecture / Exercises
Methods	▪ Lab

Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	45 Hours $\pm$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	▪ Modul PI1: Sicherer Umgang mit einer Programmiersprache. ▪ Modul PI2: Sicherer Umgang mit einer Programmiersprache. ▪ Modul DR: Kenntnisse über Aufbau und Funktionalität eines Digitalrechners. ▪ Modul GSP: Grundkenntnisse über die hardwarenahe Programmierung eines Digitalrechners. ▪ Modul NP: Grundkenntnisse über Internet-Protokolle. ▪ Modul BVS1: Sämtliche Modulinhalte, da BVS2 eine unmittelbare Fortsetzung von BVS1 ist ▪ procedural programming - architecture of a digital computer (basic knowledge) - Internet protocols (basic knowledge) - full content of BVS1
Mandatory Prerequisites	
Recommended Literature	▪ Siehe http://www.nt.fh-koeln.de/vogt/bs/bvs_lit.pdf
Use of the Module in Other Study Programs	▪ BVS2 in Bachelor Elektrotechnik 2020 ▪ BVS2 in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ BVS2 in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.8 BWR - Betriebswirtschaft und Recht

Module ID	BWR_BaTIN2020
Module Name	Betriebswirtschaft und Recht
Type of Module	Mandatory Module
Recognized Course	BWR - Business administration and law
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4
Frequency of Course	every term
Module Coordinator	Prof. Dr. Stefan Kreiser/Professor Fakultät IME
Lecturer(s)	Dr. Diana Pülpichhuysen/Lehrbeauftragte

Learning Outcome(s)

1. Fachkompetenzen (lernergebnisorientiert)

- Die Studierenden können eine eigene Business Idee generieren, mit Hilfe von Business Modelling entwickeln und validieren.
- Sie kennen die zentralen Inhaltsfelder der BWL und deren Bedeutung für Entre- und Intrapreneure.
- Sie wissen, was notwendig ist, um ein Unternehmen funktionsfähig aufzubauen und Ziel- und zukunftsorientiert zu betreiben.
- Sie kennen die für Unternehmensgründungen relevanten rechtlichen Rahmenbedingungen und können darauf aufbauend passende Entscheidungen treffen.
- Sie sind damit grundsätzlich in der Lage, betriebswirtschaftliche Problemstellungen zu analysieren, Lösungsvorschläge zu entwickeln und (theoretisch) auszuführen.

2. Fachübergreifende Kompetenzen

: Die Studierenden können im Team projektartig vorgegebene Ziele erreichen. Sie wenden hierzu erlerntes, theoretisches Wissen auf ein Praxisbeispiel an (Transferkompetenz). Sie können:

- die notwendige Literatur recherchieren, lesen und verstehen
- mit anderen Menschen zusammenzuarbeiten und gemeinsam Ziele erreichen,
- ein komplexes Arbeitsergebnis vor Publikum präsentieren sowie
- sich selbst reflektieren und Leistungen anderer bewerten.

Die Studierenden verfügen somit über

- methodisches Grundlagenwissen der Disziplinen BWL, Recht und Entrepreneurship,
- Selbst-, Sozial und Reflexionskompetenz,
- Präsentations- und Diskussionsfähigkeit.

Module Contents**Project**

Using a fictitious business start-up (business modeling), students acquire the relevant knowledge and skills from the disciplines of business administration, law and entrepreneurship in an application-oriented manner.

Lecture

1. business ideation
2. business modeling (continuous)
3. market analysis, customer group analysis, stakeholder analysis
4. operational management processes
5. legal framework, taxes
6. cost accounting, price calculation
7. external accounting
8. business model evaluation (SWOT analysis)

Further, special teaching units on:

1. self- and team management
2. presentation techniques
3. experience report of an entrepreneur

Teaching and Learning Methods	▪ Project ▪ Lecture
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	34 Hours $\pm$ 3 SWS
----------------------	----------------------

Self-Study	116 Hours
-------------------	-----------

Recommended Prerequisites**Mandatory Prerequisites**

Recommended Literature	▪ Hölter, E. (2018): Betriebswirtschaft für Studium, Schule und Beruf. Stuttgart: Schäffer-Poeschel. ▪ Osterwalder, A. & Pigneur, Y. (2010): Business Model Generation. Hoboken, New Jersey: John Wiley & Sons.
-------------------------------	--

Use of the Module in Other Study Programs	▪ BWR in Bachelor Elektrotechnik 2020 ▪ BWR in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ BWR in Bachelor Medientechnologie 2020 ▪ BWR in Bachelor Medientechnologie 2024 ▪ BWR in Bachelor Optometrie 2021 ▪ BWR in Bachelor Informatik und Systems-Engineering 2024
--	--

Permanent Links to Organization	llu
--	---------------------

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.9 CA - Computeranimation

Module ID	CA_BaTIN2020
Module Name	Computeranimation
Type of Module	Elective Modules
Recognized Course	CA - Computer Animation
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Stefan Grünvogel/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Stefan Grünvogel/Professor Fakultät IME

Learning Outcome(s)

WAS:

Die mathematischen, algorithmischen und theoretischen Grundlagen der Computeranimation erklären können, schriftlich und mündlich, unter Verwendung der entsprechenden Fachtermini.

WOMIT:

Die entsprechenden Grundlagen werden nach dem Prinzip des Flipped Classrooms vermittelt und zunächst in Form von einfachen Aufgaben (ohne Hilfe von Software) schriftlich geübt.

WOZU:

Um Anwendungen und Software zur Computeranimation nicht nur als Black Box zu verwenden, sondern auch deren Arbeitsweise zu verstehen und sich selbstständig in weiterführende (wissenschaftliche) Themengebiete der Computeranimation einarbeiten zu können.

WAS:

Eine Problemstellung oder Aufgabenstellung aus dem Bereich der Computeranimation analysieren und die passenden Methoden und Verfahren auswählen zu können.

WOMIT:

Im Praktikum wird schrittweise an die Herangehensweise zur Lösen von Aufgabenstellungen in der Computeanimation herangeführt und typische Lösungsansätze vermittelt. Dazu notwendige fachlichen Kenntnisse werden per Flipped Classroom vermittelt.

WOZU:

Um Verfahren, Algorithmen und Geräten zur Produktion, Speicherung, Übertragung, Verarbeitung, Wiedergabe und Präsentation von Computeranimation analysieren und bewerten zu können.

WAS:

Methoden und Software der Computeranimation anwenden, weiterentwickeln oder selbst entwickeln.

WOMIT:

Im Praktikum werden schrittweise an Hand einer Game Engine oder einer Softwarebibliotheken die Kenntnisse in Form praktischer Übungsaufgaben vertieft und die Implementierung von Software zur Computeranimation geübt.

WOZU:

Um Verfahren, Algorithmen und Geräte zu Produktion und Wiedergabe von Computeranimation entwickeln und integrieren können.

Module Contents**Seminar-style Teaching**

animation systems

- Hierarchies in Scenes
- animation system
- Time and Game Loop

object animation

- Movement in space
- Time, speed and distance control
- interpolation
- rotations

Characteranimation

- kinematics
- skinning
- blend shapes
- motion capture
- Processing of transaction data

Procedural Animation

- Physically based animation
- particle systems

Teaching and Learning Methods Seminar-style Teaching

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 23 Hours $\pm$ 2 SWS

Self-Study 127 Hours

Recommended Prerequisites

- Modul MA1: Problemlösungskompetenz aus dem Bereich Analysis einer Veränderlichen. Sicheres Beherrschen der entsprechenden Symbole und Formalismen
- Modul MA2: Problemlösungskompetenz aus dem Bereich lineare Algebra und Analysis mehrerer Veränderlichen sowie Differentialgleichungen. Sicheres Beherrschen der entsprechenden Symbole und Formalismen.
- Modul PI1: Entwickeln von Programmen zur Lösung konkreter Problemstellungen, abstrahieren von Problembeschreibungen in Algorithmen und überprüfen von Programmen auf Fehler.
- Modul PI2: Entwerfen und verwenden objekt-orientierter Modelle und dynamischer Datenstrukturen zu einer gegebenen Problemstellung und Umsetzung in einer Programmiersprache. Lösen von Problemstellung mittels geeigneter Algorithmen
- Basic knowledge of computer graphics

Programming knowledge imparted in the scope of Computer Science 1 and Computer Science 2
confident handling of linear algebra as well as analysis of one and more variables by scope of knowledge from mathematics 1 and mathematics 2

Mandatory Prerequisites

Recommended Literature

- Stefan M. Grünvogel, Einführung in die Computeranimation, Springer, 2024
- Rick Parent, Computer Animation: Algorithms and Techniques, Morgan Kaufmann, 2007,
- Dietmar Jackel et. al., Methoden der Computeranimation, Springer, 2006
- Jason Gregory, Game Engine Architecture, AK Peters, 2009

Use of the Module in Other Study Programs	▪ CA in Bachelor Medientechnologie 2020 ▪ CA in Bachelor Medientechnologie 2024 ▪ CA in Bachelor Informatik und Systems-Engineering 2024
--	--

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.10 CG - Computergrafik

Module ID	CG_BaTIN2020
Module Name	Computergrafik
Type of Module	Elective Modules
Recognized Course	CG - Computer Graphics
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr.-Ing. Arnulph Fuhrmann/Professor Fakultät IME
Lecturer(s)	Prof. Dr.-Ing. Arnulph Fuhrmann/Professor Fakultät IME

Learning Outcome(s)

Das Modul vermittelt folgende Kenntnisse und Fertigkeiten:

- Beschreiben von Methoden zum geometrischen Modellieren
- Erklären von Transformationen
- Beschreiben der grundlegenden Graphikhardware
- Beschreiben der einzelnen Stufen der Rendering Pipeline
- Erklären von globalen und lokalen Beleuchtungsmodellen
- Beschreiben von Methoden zur Texturierung
- Gegenüberstellen der behandelten Beleuchtungsmodelle
- Entscheiden welches Verfahren geeignet ist, um eine konkrete Problemstellung der Computergrafik zu lösen
- Entwickeln von Computergrafikanwendungen (Verwenden eines 3D-APIs, Erstellen interaktiver 3D-Programme, Anwenden der mathematischen Basis der Computergrafik, Anwenden der grundlegenden Algorithmen der Computergrafik, Testen und Debuggen von Anwendungen)

Die Kompetenzen werden zunächst über eine Vorlesung durch den Dozenten vermittelt und danach im Praktikum von den Studierenden vertieft.

Die sichere Anwendung der Grundlagen der Computergrafik ist Voraussetzung für die Entwicklung interaktiver medientechnischer Systeme (HF1, HF2) und erlaubt die Bewertung bestehender Systeme (HF2).

Module Contents**Lecture / Exercises**

- Geometric Modeling
 - Polygonal meshes
 - subdivisional surfaces
- Transformations
 - coordinate systems
 - fundamental transformations
 - projections
- Graphics Hardware
 - raster displays
 - video cards
 - input devices
- Rendering Pipeline
 - rasterization
 - clipping
 - shading
 - visiblity
 - shader programming
- Local reflection models
 - light sources
 - reflection
 - transparency
 - BRDFs
- Textures
 - texture mapping
 - generation of texture coordinates
 - filtering
 - normal maps
 - environment maps
 - displacement maps
- Global illumination
 - rendering equation
 - raytracing
 - spatial data structures
 - shadows

Lab

- Developing computer graphics applications
- Create interactive 3D programs
- Using a 3D API
- Applying the mathematical basis of Computer Graphics
- Applying the fundamental algorithms of Computer Graphics
- Testing and debugging of own applications
- Capturing and understanding textual instructions

Teaching and Learning	▪ Lecture / Exercises
Methods	▪ Lab

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\triangleq$ 4 SWS
----------------------	-----------------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	Programming Mathematics 1 and 2
Mandatory Prerequisites	Lab requires attendance in the amount of: 3 Termine
Recommended Literature	▪ P. Shirley, S. Marschner: Fundamentals of Computer Graphics, Fifth Edition, AK Peters, 2021 ▪ T. Akenine-Möller, et al.: Real-Time Rendering, Taylor & Francis Ltd., 2018 ▪ M. Pharr, W. Jakob, and G. Humphreys, Physically Based Rendering: From Theory To Implementation, Morgan Kaufmann, 4. Edition, 2023
Use of the Module in Other Study Programs	▪ CG in Bachelor Medientechnologie 2020 ▪ CG in Bachelor Medientechnologie 2024 ▪ CG in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.11 DB1 - Datenbanken 1

Module ID	DB1_BaTIN2020
Module Name	Datenbanken 1
Type of Module	Mandatory Module
Recognized Course	DB1 - Data Base Systems 1
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Andreas Behrend/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Andreas Behrend/Professor Fakultät IME

Learning Outcome(s)

WAS? HF1: Studierende sollen den Aufbau von Datenbanksystemen zur Speicherung und Verarbeitung von Informationen kennenlernen. Sie sollen relationale Datenbanken erstellen und Anfragen mittels SQL auf diese Datenbanken programmieren können. Dabei sollen Sie auch in der Lage sein, diese Anfragen in andere Programmiersprachen einzubetten (z.B. SQL-Anfragen in Java mittels der JDBC-Schnittstelle einbetten). Sie sollen in der Lage sein, den Datenaustausch mit benachbarten Softwaresystemen über definierte Austauschformate (z.B. XML) realisieren zu können.

HF2: Gegebene Anforderungskataloge für zu entwickelnde Softwaresysteme sollen auf ihren Bedarf an persistenten Daten analysiert werden können. Dabei sollen unterschiedliche Persistenzmechanismen analysiert werden können. Hierzu sollen verschiedene Datenbankmodelle im Überblick kennengelernt werden. In Bezug auf relationale Datenbanken als Zielsysteme sollen ERD-Modelle entwickelt und normalisiert werden können.

HF3: Kleinere Datenbanksysteme, die nach analytischen Vorgaben selbst entwickelt wurden, sollen mit Schnittstellen zu Nachbarsystemen organisiert und betrieben werden können.

WOMIT? Vortrag zu HF1, HF2 und HF3. Üben an Hand praktischer Beispiele zu HF1, HF2 und HF3. Drei kleinere Projekte in Laborversuchen zu HF1 und HF3, dabei sollen die Voraussetzungen zur Erstellung der Lösungen für HF1 und HF3 mittels Analysetechniken aus HF2 spezifiziert und anschließend dokumentiert werden können.

WOZU? In Softwareabteilungen großer Industrie- und Dienstleistungsunternehmen und bei Unternehmensberatungen für Soft- und Hardwaresystemen spielt die Entwicklung von Datenbanksystemen eine sehr große Rolle. Hier werden Informatiker dringend benötigt, die Datenbanken entwerfen, hierauf bezogene Anfrageprogramme entwickeln und testen können und Datenbanken in Betrieb halten können.

Module Contents

Lecture / Exercises

General Data Base Model
 Relational data base system
 Structured Query Language
 Embedded SQL in Java (JDBC)
 Specification and design of data bases
 Entity Relationship Model
 Normal forms of a relational data base
 XML and data bases
 Document type definition

Lab

Implementation of a data base with several tables, which is specified in a requirement analysis; programming of embedded complex SQL code in JAVA; entity relationship modeling; definition of an DB/XML-interface with a document type definition

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\triangleq$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

- Modul MA1: Mengen, kartesisches Produkt, Relationen
- Basic Course Mathematics
- Basic Course Computer Science

Mandatory Prerequisites

Recommended Literature

- G. Vossen: Datenmodelle, Datenbanksprachen und Datenbankmanagementsysteme
- A. Kemper, A. Eickler: Datenbanksysteme
- C. Türker: SQL 1999 & SQL 2003

Use of the Module in Other Study Programs

- DB in Bachelor Elektrotechnik 2020
- DB1 in Bachelor Elektrotechnik und Informationstechnik 2024
- DB1 in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.12 DB2 - Datenbanken 2

Module ID	DB2_BaTIN2020
Module Name	Datenbanken 2
Type of Module	Elective Modules
Recognized Course	DB2 - Data Base Systems 2
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Andreas Behrend/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Andreas Behrend/Professor Fakultät IME

Learning Outcome(s)

WAS? HF1: Studierende sollen neben dem relationalen auch andere Datenbanksysteme kennenlernen: objektorientierte, objektrelationale und NoSQL-Datenbanksysteme. Weiterhin sollen sie neben dem ERM weitere Datenbank-Design Methoden kennenlernen, z.B. abstrakte Datentypen (ADT). Die Studierenden sollen die wichtigsten Arten von NoSQL-Datenbanksystemen kennenlernen: Key-Value-Datenbanken, Wide Column Stores, dokumentorientierte Datenbanken. Sie sollen die unterschiedlichen Schema Anforderungen analysieren lernen sowie die unterschiedlichen nicht-relationalen Datenmodelle, wie z.B. der JSON-Dokumenttyp bei dem NoSQL-Datenbanksystem CouchDB. Die Studierenden sollen objektrelationale Datenbanken und NoSQL Datenbanken erstellen und Anfragen auf diese Datenbanken programmieren können. Dabei sollen Sie auch in der Lage sein, Schnittstellen auf der Basis unterschiedlicher Austauschformate (z.B. JSON und XML) unter Berücksichtigung unterschiedlicher Grammatikmodelle (bei JSON: JSON-Schema, bei XML: XML-Schema) programmieren zu können. Weiterhin sollen Sie Bayer-Bäume als wesentliche Datenstruktur für die Sekundärspeicherverwaltung kennenlernen

HF2: Gegebene Anforderungskataloge für zu entwickelnde Datenstrukturen sollen sowohl objektrelationale als auch in Hinblick auf dokumentenorientierte NoSQL Datenbanken modelliert werden können. Hierbei soll das Konzept abstrakter Datentypen angewendet werden können. Das Leistungsverhalten von Algorithmen der Sekundärspeicherverwaltung, die auf Bayer-Bäumen basieren, sollen im Unterschied zu anderen Strukturen der Sekundärspeicherverwaltung (z.B. ISAM) bewertet werden können. Unterschiedliche Grammatikmodelle für Austauschformate sollen zum validierenden Parsen von XML- oder JSON-Daten für Datenbankschnittstellen angewendet werden können.

HF3: Objektrelationale und NoSQL Datenbanksysteme, die nach analytischen Vorgaben selbst entwickelt wurden, sollen mit Schnittstellen zu Nachbarsystemen organisiert und betrieben werden können.

WOMIT? Vortrag zu HF1, HF2 und HF3. Üben an Hand praktischer Beispiele zu HF1, HF2 und HF3. Drei kleinere Projekte in Laborversuchen zu HF1, HF2 und HF3, dabei sollen die Voraussetzungen zur Erstellung der Lösungen für HF1, HF2 und HF3 mittels der in HF2 genannten Grammatikmodelle und dem in HF1 genannten Konzept abstrakter Datentypen HF2 spezifiziert und nach Implementation dokumentiert werden können.

WOZU? In großen Unternehmen und auf größeren Internetplattformen (z.B. Google, Ebay, Amazon) spielen skalierbare, hochperformante und föderierte Cloud-Datenbanken eine große Rolle. Diese Datenbanken operieren mit NoSQL- und ADT-Konzepten. Industrie- und Dienstleistungsunternehmen, die bei der Umsetzung von Industrie 4.0 Strategien Cloud-Datenbanken anwenden oder entwickeln möchten, benötigen dringend Informatiker, die z.B. NoSQL Datenbanken entwerfen, hierauf bezogene Anfrageprogramme entwickeln, testen und in Betrieb halten können.

Module Contents**Lecture / Exercises**

XML Grammar with XML scheme
 abstract data types
 object oriented data bases
 object relational data bases
 NoSQL data bases
 Bayer trees

Lab

Definition of XML schemes, of abstract data types, of data types in object-relational data bases. Programming of CRUD Operations on object-relational data bases and on NoSQL data bases. Validating parsing of JSON documents.

Teaching and Learning ■ Lecture / Exercises
Methods ■ Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

- Modul DB1: Relationale Datenbanken, SQL, XML mit DTD, JDBC.
- Modul FSA: Klassen formaler Grammatiken nach Chomsky.
- Modul BVS1: Funktionalität und Aufbau des Dateisystems als Teil des Betriebssystemkerns.
- Modul SE: Prinzipien und Methoden der Spezifikation verteilter Softwaresysteme.
- Basic Course Mathematics
- Basic Course Computer Science
- Data Base Systems 1

Mandatory Prerequisites

Recommended Literature

- C. Türker: SQL 1999 & SQL 2003
- St. Edlich: NoSQL Datenbanken

Use of the Module in Other Study Programs DB2 in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.13 DM - Data Mining

Module ID	DM_BaTIN2020
Module Name	Data Mining
Type of Module	Elective Modules
Recognized Course	DM - Data Mining
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Beate Rhein/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Beate Rhein/Professor Fakultät IME

Learning Outcome(s)

Was:

Methoden des maschinellen Lernens auf typische Datensätze der technischen Informatik anwenden
gängige Fallstricke des Data Mining in der Vorgehensweise kennen
für eine Aufgabenstellung das geeignete Verfahren bestimmen können
Qualität von Datensätzen beurteilen
Datenschutzgesetze kennen
weit verbreitete Software hierfür anwenden
eigenverantwortliches Arbeiten lernen

Womit:

Die Methoden werden anhand eines Vortrags oder per Lernvideos vermittelt und in Vorlesung und Übung direkt angewendet. Jeder Student wird ein kleines Projekt durchführen (je nach Anzahl der Studierenden in Gruppenarbeit).

Wozu:

Data Mining wird bei den späteren Arbeitgebern immer mehr eingeführt, etwa in der Robotik, aber auch zur Überwachung und Steuerung von Produktionsprozessen oder Energiesystemen und zur Auswertung von Kundendaten, hier ist ein verantwortlicher Einsatz von Daten wichtig

Module Contents**Lecture / Exercises**

Introduction to a suitable software, e.g. Python

Introduction to descriptive statistics and possibly also probability calculation

Supervised learning:

- Classification procedure: Procedure, performance measures, application of a method of instance-based learning, e.g. k-nearest-neighbor and a method of model-based learning, e.g. decision trees
- Possible regression analysis: about machine learning and classical

Unsupervised learning:

- Cluster analysis: k-means, possibly also DBSCAN

Preprocessing of the data:

- Handling Damaged / Missing Data
- Runaway or noise - problems
- Scaling
- Visualization of data
- Possible dimension reduction
- Assessment of data quality
- possibly look at different types of data records, make reference to NoSql databases

Outlook on current research, e.g. image recognition, Natural Language Processing, Reinforcement Learning

Be able to name and apply a suitable method and overall approach to tasks

Select and evaluate a suitable performance measure

Apply Privacy Policy

Exercises / Lab

Teaching and Learning	▪ Lecture / Exercises
Methods	▪ Exercises / Lab

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	57 Hours $\triangleq$ 5 SWS
----------------------	-----------------------------

Self-Study	93 Hours
-------------------	----------

Recommended Prerequisites	▪ Modul MA1: mathematische Modelle verstehen und aufstellen Differentialrechnung ▪ Modul MA2: Funktionen mit mehreren Veränderlichen anwenden Lineare Algebra: Matrizen aufstellen und mit ihnen rechnen ▪ Modul PI1: Grundlagen der Programmierung beherrschen ▪ From Mathematics 1 and 2 the ability to construct mathematical models as well as knowledge of differential calculus and linear algebra is required.
----------------------------------	--

Mandatory Prerequisites	Participation in final examination only after successful participation in Exercises / Lab
--------------------------------	---

Recommended Literature	▪ A. Geron: Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow: Konzepte, Tools und Techniken für intelligente Systeme, Heidelberg, o'Reilly Verlag 2017, 978-3960090618 ▪ S. Raschka, V. Mirjalili: Machine Learning mit Python und Scikit-Learn und TensorFlow: Das umfassende Praxis-Handbuch für Data Science, Predictive Analytics und Deep Learning, mitp Verlag, 2018, 978-3958457331 ▪ J. Frochte, Jörg: Maschinelles Lernen, München, Carl Hanser Verlag GmbH & Co. KG, 2018, eBook ISBN: 978-3-446-45705-8, Print ISBN: 978-3-446-45291-6 ▪ A. Müller: Einführung in Machine Learning mit Python: Praxiswissen Data Science, Heidelberg, o'Reilly Verlag 2017, eBook: 978-3-96010-111-6
-------------------------------	---

Use of the Module in Other Study Programs	▪ DML in Bachelor Elektrotechnik 2020 ▪ DM in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ DM in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.14 DR - Digitalrechner

Module ID	DR_BaTIN2020
Module Name	Digitalrechner
Type of Module	Mandatory Module
Recognized Course	DR - Digital Computer
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt die grundlegenden Methoden und Systeme der Digitaltechnik sowie den Entwurf digitaltechnischer Systeme unter Verwendung programmierbarer Bausteinen. Dies geschieht insbesondere auch mit dem Ziel, dass die Studierenden das Prinzip, den Aufbau und die Funktionsweise eines Digitalrechners verstehen und in Form eines rudimentären Von-Neumann-Rechners auch selbst entwickeln und mittels Maschinensprache programmieren können. Aufbauend und vergleichend zu den rudimentären Von-Neumann-Rechner erlernen die Studierenden die grundlegende prinzipielle Funktionsweise einer gängigen CPU (z.B. IA32E-Architektur). Die Studierenden werden in die Lage versetzt, fachspezifische Begriffe, Tools und Techniken im praktischen Umfeld sicher anzuwenden. Aufbauend auf den in der Vorlesung vermittelten Kenntnissen werden komplexere Problemstellungen analysiert, auf Teilsysteme heruntergebrochen und modelliert. Darauf aufbauend wird die Problemlösung mittels Entwurfswerkzeugen implementiert, simuliert, getestet und am Zielsystem in Betrieb genommen.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. Im Praktikum erarbeiten die Studierenden in Kleingruppen Problemlösungen und verteidigen diese.

Wozu: Kompetenzen in der Entwicklung digitaltechnischer Systeme und hier insbesondere auch von Digitalrechnern sind essentiell für technische Informatiker, die im HF 1 arbeiten wollen. Durch die Entwicklung von Problemlösungen erwerben die Studierenden zudem Erfahrungen, die essentiell für das HF 2 sind. Eine projektorientierte Durchführung der Praktika in kleinen Teams mit dem Dozenten als "Auftraggeber" initiiert die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents

Lecture / Exercises

Project

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Project
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	none
----------------------------------	------

Mandatory Prerequisites	Project requires attendance in the amount of: 5 Termine
--------------------------------	---

Recommended Literature	▪ Urbanski K., Woitowikz R.: Digitaltechnik, 4. Auflage Springer 2004 ▪ Beuth K.: Elektronik Bd. 4 Digitaltechnik, Vogel Verlag 2001 ▪ Lipp H.M.: Grundlagen der Digitaltechnik, 4. Auflage Oldenbourg 2002 ▪ Tanenbaum A. S.; Austin T.. Rechnerarchitektur: Von der digitalen Logik zum Parallelrechner Pearson Deutschland 2014
-------------------------------	--

Use of the Module in Other Study Programs	DR in Bachelor Informatik und Systems-Engineering 2024
--	--

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.15 EG - Elektrotechnische Grundlagen für die Technische Informatik

Module ID	EG_BaTIN2020
Module Name	Elektrotechnische Grundlagen für die Technische Informatik
Type of Module	Mandatory Module
Recognized Course	EG - Basic Electrical Engineering for Computer Science and Engineering
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Kai Kreisköther/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Kai Kreisköther/Professor Fakultät IME ▪ Prof. Dr.-Ing. Dirk Poggemann/Professor Fakultät IME

Learning Outcome(s)

- **Was:** Die Studierenden analysieren die wesentliche Funktionsweise von elektrotechnischen und elektronischen Systemen unter Einwirkung von zeitunveränderlichen und zeitveränderlichen Spannungen und Strömen. Sie sind in der Lage, deren Verhalten einzuordnen und abzuschätzen. Dies umfasst grundlegendes Wissen über Spannung, Strom, Widerstand, Quellen, Kirchhoffsche Gesetze, Wechselstromkreise, passive und aktive Bauelemente (Diode, Transistor, Operationsverstärker), Hoch- und Tiefpässe, Schwingkreise, Transformatoren, Messtechnik elektrischer Größen, Digitaltechnik, A/D- und D/A-Wandlung, Halbleiterspeicher sowie Signalübertragung auf Leitungen. Zudem sind sie befähigt, sich in diesem Themengebiet fachlich auszutauschen.
- **Womit:** Das notwendige Wissen und grundlegende Fertigkeiten werden durch Vorlesungen und Übungen vermittelt. In betreuten Praktikumsversuchen vertiefen und wenden die Studierenden ihre Kenntnisse praktisch an.
- **Wozu:** Elektrotechnische Systeme bilden die technologische Basis für viele Informationssysteme und sind in zahlreichen technischen Anwendungsbereichen relevant. Ein grundlegendes Verständnis dieser Systeme ist für Studierende technischer Fächer unerlässlich, um moderne Technologien zu verstehen und in interdisziplinären Kontexten effektiv zu kommunizieren, auch wenn die Entwicklung solcher Systeme nicht im Fokus ihrer Tätigkeit steht.

Module Contents**Lecture / Exercises**

The students are able to analyze electrical and electronic systems in respect to the essential functionality and to classify and estimate their behavior.

In particular, students are able perform these analyzes according to following topics:

- resistor
- voltage and current sources
- Kirchhoff's circuit laws, serial and parallel
- electrical power and efficiency
- real electrical sources including operating point
- network analysis
- electric field
- magnetic field

- inductors and capacitors
- apparent power and reactive power
- Switching in simple RCL networks
- AC
- transformer
- generator
- DC motor

- ideal diode
- real diode (modeled using an ideal diode and voltage source and resistor)
- ideal transistor
- real transistor
- operational amplifier and corresponding basic wirings

Lab

The students carry out electrotechnical experiments in related projects. The aim of the given experiments is the understanding of the function and the measurement of an electrotechnical and / or electronic system.

Teaching and Learning	▪ Lecture / Exercises
Methods	▪ Lab

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\triangleq$ 4 SWS
----------------------	-----------------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	none
----------------------------------	------

Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 4 Termine ▪ Participation in final examination only after successful participation in Lab
--------------------------------	--

Recommended Literature	▪ Gert Hagman, Grundlagen der Elektrotechnik, AULA-Verlag, ISBN 978-3-89104-747-7
-------------------------------	---

Use of the Module in Other Study Programs	▪ EG in Bachelor Medientechnologie 2024 ▪ EG in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.16 EKS - Entwicklung komplexer Software-Systeme

Module ID	EKS_BaTIN2020
Module Name	Entwicklung komplexer Software-Systeme
Type of Module	Elective Modules
Recognized Course	EKS - Development of Complex Software Systems
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Hans Nissen/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Hans Nissen/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt unterschiedliche Kompetenzen, die für die Entwicklung und Pflege komplexer Software-Systeme erforderlich sind: die Anwendung und Beurteilung von Entwurfsmustern (K.1, K.3 K.4, K.5, K.9, K.10), die Anwendung von Ansätzen zur professionellen Code-Entwicklung (K.6, K.9), der Einsatz und die Beurteilung von Verfahren zur statischen Code-Analyse (K.4, K.7, K.9), die Beherrschung fortgeschrittene Java-Konzepte (K.5, K.6), der Entwurf und die Realisierung modularisierter Software-Architekturen (K.1, K.3, K.5, K.10), die Einordnung und der Einsatz komplexer Testverfahren (K.2, K.7, K.9), die Fähigkeit, komplexe fachbezogene Probleme zu sehen und Lösungen gegenüber Fachleuten mündlich argumentativ zu vertreten und mit anderen Studierenden weiterzuentwickeln (K.3, K.5, K.7, K.9, K.11, K.16).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs/Übungsteil unter Verwendung einer Fallstudie und verschiedenen praktischen Demonstrationen.

Im zugehörigen Praktikum erarbeiten die Studierenden in Kleingruppen Lösungen zu vorgegebenen Problemen und wenden dabei die Themenbereiche der Vorlesung praktisch an. Hierzu wird eine selbständige Vertiefung einzelner Themenbereiche, insbesondere die Verwendung typischer Werkzeuge, verlangt (K.8, K.9, K.15). An den Präsenzterminen müssen die Studierenden ihre Lösungen erläutern und verteidigen (K.16).

Wozu: Bei der Entwicklung und Pflege moderner Software muss man sich mit einer stetig zunehmenden System-Komplexität auseinandersetzen. Einige typische Gründe für eine hohe Komplexität sind: die Systeme sind sehr umfangreich, es bestehen sehr viele Schnittstellen zu anderen Systemen, es werden viele und zum Teil sehr umfangreiche Frameworks eingesetzt. Für eine erfolgreiche Tätigkeit in diesem Umfeld ist die sichere Beherrschung hierauf ausgerichteter Methoden, Konzepte und Technologien unbedingt erforderlich (HF.1). Die Erweiterung eines komplexen Systems erfordert die umfangreiche Analyse der bestehenden Abhängigkeiten und die Beurteilung der unterschiedlichen Entwurfsalternativen bezüglich ihrer Auswirkungen auf das Gesamtsystem (HF.2). Ein wesentliches Hilfsmittel zur Beherrschung der System-Komplexität stellt die Wahl einer geeigneten Organisationsform der Systems dar (HF.3). Hierfür sind umfangreiche theoretische und praktische Kenntnisse von Organisationsformen im Kleinen (z.B. Entwurfsmuster) und im Großen (z.B. Modularisierung) erforderlich.

Module Contents**Lecture / Exercises**

design patterns
 modularization
 professional code development
 advanced Java concepts
 Module-oriented architectural principles
 complex test procedures
 apply and evaluate design patterns
 Apply and evaluate approaches to professional code development
 Apply automated code analysis methods and interpret the results
 Design and implement modularized architectures
 use complex test procedures

Lab

implemtation of design pattern
 Create modularized architectures for large-scale applications
 apply automated code review and static code analysis
 Select test method and apply to programs

Teaching and Learning ■ Lecture / Exercises
Methods ■ Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

- Modul SE: Spezifikation und Modellierung von Systemen und Software mit UML, Modularisierung in Java, einfache Entwurfsmuster, grundlegende Verfahren zum Prüfen von Software, verschiedene Architekturen von Systemen und Software, Grundbegriffe der Qualitätssicherung, Kenntnisse in Versionsverwaltung
- Modul PI1: sehr gute praktische und theoretische Kenntnisse der Programmiersprache Java
- Modul PI2: sehr gute praktische und theoretische Kenntnisse der Programmiersprache Java
- Modul PP: sehr gute praktische und theoretische Kenntnisse der Programmiersprache Java
- Specification and modeling of systems and software with UML, modularization in Java, simple design patterns, basic software testing methode, various architectures of systems and software, basic quality assurance concepts, version management skills, very good practical and theoretical knowledge of the programming language Java

Mandatory Prerequisites

- Lab requires attendance in the amount of: 3 Termine
- Participation in final examination only after successful participation in Lab

Recommended Literature

- E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns, MITP Verlags GmbH & Co. KG, 2015.
- R. C. Martin: Clean Code: A Handbook of Agile Software Craftsmanship, Prentice Hall, 2008.
- S. McConnell: Code Complete, Microsoft Press, 2. Auflage, 2004.
- M. Fowler: Refactoring: Improving the Design of Existing Code. Addison-Wesley Verlag, 2. Auflage, 2018.
- A. Spillner, T. Linz: Basiswissen Softwaretest, dpunkt Verlag, 5. Auflage, 2012
- P. Liggesmeyer: Software-Qualität: Testen, Analysieren und Verifizieren von Software, Spektrum Akademischer Verlag, 2. Auflage, 2009.
- H.M. Sneed, M. Winter: Testen objektorientierter Software, Hanser Verlag, 2001.
- G. E. Thaller: Software-Metriken einsetzen–bewerten–messen, Verlag Technik, 2000.
- H. Sneed, R. Seidl, Manfred Baumgartner: Software in Zahlen - Die Vermessung von Applikationen. Carl Hanser Verlag, 2010.
- Standard ISO/IEC 5055, Software Quality Measurement, 2021.
- J. Laski, W. Stanley: Software Verification and Analysis, Springer 2019.
- J. Goll: Entwurfsprinzipien und Konstruktionskonzepte der Softwaretechnik, Springer Vieweg, 2018.
- J. Goll: Architektur- und Entwurfsmuster in der Softwaretechnik, Springer 2014.
- K. Beck: Tidy First? Mini-Refactorings für besseres software-Design, O'Reilly, 2024.
- J. Ousterhout: Prinzipien des Software-designs, O'Reilly, 2022.

Use of the Module in Other Study ProgramsEKS in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update19.7.2025, 14:32:16

6.17 EL - Elektronik

Module ID	EL_BaTIN2020
Module Name	Elektronik
Type of Module	Elective Modules
Recognized Course	EL - Electronic Circuits
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr.-Ing. Christoph Pörschmann/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Alexander Utz/Professor Fakultät IME

Learning Outcome(s)

Grundlegende Schaltungen mit passiven Bauelementen und Halbleitern (Dioden, BIP-Transistoren) kennenlernen und mit Hilfe erlernter Methoden das Verhalten der Schaltung analysieren. Zur Vermittlung dieser Kompetenzen werden Übungen an Hand exemplarischer Schaltungen durchgeführt und praktische Erfahrungen in Laborversuchen gesammelt. Dadurch wird den Studierenden ermöglicht, in der späteren beruflichen Tätigkeit schaltungstechnische Aufgabenstellungen zu analysieren, Lösungskonzepte zu erarbeiten und letztlich schaltungstechnische Systeme zu entwickeln.

Module Contents**Lecture / Exercises**

knowing and analysing of linear passive circuits
 calculation of frequency dependent behaviour
 graphical representation using the bode plot
 knowing semiconductor elements (diode, transistor) and operational amplifiers and dimensioning them

Lab

read and understand technical instructions
 connect circuits and demonstrate the function
 work on complex task in limited time
 transfer theoretic knowledge into working circuits
 discuss the results
 work with typical measurement equipment
 explain technical basics and their interdependence

Teaching and Learning ■ Lecture / Exercises
Methods ■ Lab

Examination Types with Weights cf. exam regulations

Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours

Recommended Prerequisites	▪ Modul EG: Kennen und Verstehen elektrischer Bauelemente R,L,C und einfacher passiver Schaltungen. Analysieren von Schaltungen durch Knoten- und Maschengleichungen, Technische Zusammenhänge darstellen und erläutern, Kennlinien von Bauelementen verstehen und interpretieren ▪ Modul MA1: Mathematisches Grundwissen sicher anwenden, insbesondere lineare Gleichungen, Termumformungen, Kürzen von Termen, Bruchrechnung ▪ Modul MA2: Lösen einfacher Differential- oder Integralgleichungen ▪ basic skills in calculating electric circuits, resistor, capacitor, inductor ▪ good knowledge in mathematics, linear equations, calculations with complex terms
----------------------------------	--

Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 3 Termine ▪ Participation in final examination only after successful participation in Lab
--------------------------------	---

Recommended Literature	▪ "Halbleiter-Schaltungstechnik", U. Tietze et al., ISBN: 978-3-662-48553-8 ▪ "Design of Analog CMOS Integrated Circuits", B. Razavi, ISBN: 978-0-07-252493-2
-------------------------------	---

Use of the Module in Other Study Programs	▪ EL in Bachelor Elektrotechnik 2020 ▪ EL in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ EL in Bachelor Informatik und Systems-Engineering 2024
--	--

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.18 ES - Eingebettete Systeme

Module ID	ES_BaTIN2020
Module Name	Eingebettete Systeme
Type of Module	Elective Modules
Recognized Course	ES - Embedded Systems
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Tobias Krawutschke/Professor Fakultät IME ▪ Prof. Dr. Markus Cremer/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt Kompetenzen zur Erstellung eines Eingebetteten Systems (ES) in allen Arbeitsschritten von der Auslegung und Planung des Systems (K.4), der Auswahl der Komponenten (K.8, K.10), der Entwicklung der Software und der Anschaltung an die Anlage / das Gerät (K.5) und seiner prototypischen Inbetriebnahme (K.6). Im parallel laufenden Miniprojekt werden ihre Kompetenzen zur Teamarbeit (K.13), Projektorganisation (K.11) und Kommunikation (durch Präsentationen und Bericht) (K.16) verstärkt.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs/Übungsteil und betreut parallel dazu ein Miniprojekt, in dem die Studierenden ein kleines ES entwickeln.

Wozu: Kompetenzen in der Entwicklung eines ES sind essentiell für technische Informatiker, die im HF 1 arbeiten wollen. Durch die Arbeit an einem Beispielsystem erwerben die Studierenden zudem Erfahrungen, die essentiell für das HF 2 sind, u.a. Anforderungen erfassen, Konzepte zur technischen Lösung entwickeln und diese zu bewerten. Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents

Lecture

- analysis and specification methods
- functional decomposition
- behavior description
- object oriented description
- description of parallel behavior with Petri nets
- engineering of embedded systems
- hardware aspects
- Microcontroller
- SOC system on (programmable) chip
- use of I/O controllers
- serial interface
- parallel interface
- DMA
- energy awareness
- software aspects
- choice of programming language
- Assembler
- C
- C++
- andere
- software system architecture
- singletasking
- Implementing a FSM (finite state machine)
- table based static function scheduling
- multitasking
- RTOS with an example
- Embedded Linux
- timing requirements
- Distributed embedded systems
- Basics of distributed systems
- communication system levelling
- basics of field busses
- basics of Internet of Things (IoT)
- programming distributed embedded systems

Project

Teamwork: Development of an embedded system with dedicated function, e.g. control of a mechanical model, environmental sensor etc. Aim: building a prototype

Steps

1. Description/Specification
Task description taking the client's view in communication with client (= docent)
2. Hardware architecture
recherche of suitable modules in technical documents
3. Modelling the solution
4. Implementation using modern PDE and standards, especially RTOS

- mastering complex tasks with the team
- project planning and steering
- fulfilling tasks on time

Presentation of Development

- Task description
- Project intermediate presentation
- Result
- Documentation in project report

Project description
Project implementation
User documentation
Experiences

Exercises

Modelling of an Embedded System using well-known design methods for reactive systems
Writing Software for an embedded system using C
on base of a HAL (hardware abstraction layer) or by using a RTOS

Teaching and Learning Methods

- Lecture
- Project
- Exercises

Examination Types with Weights

cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\triangleq$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

- Modul DR: Grundkenntnisse Technologie und Programmierung von digitaler Logik und Mikrocontrollern
- Modul PP: Programmiererfahrung
- Modul EG: Grundkenntnisse der Elektrotechnik für die Benutzung von Mikrocontrollern und die Erstellung einfacher Anschaltungen an Mikrocontroller, z.B. mit Spannungsteiler oder einfachem Operationsverstärker
- basic knowledge in computer engineering
FSA and FSM
Microcontroller structure and function
Imperative Programming language (pref. C)
Experiences in Program development using program development environments, e.g. Eclipse

Mandatory Prerequisites

Project requires attendance in the amount of: Projektbesprechungen und -präsentation

Recommended Literature

- W.Wolff: Computers as Compenents: Principles of Embedded System Design
- Wieringa: Design Methods for reactive Systems

Use of the Module in Other Study Programs

- ES in Bachelor Elektrotechnik 2020
- ES in Bachelor Elektrotechnik und Informationstechnik 2024
- ES in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.19 ESP - Eingebettete Systeme - Projekt

Module ID	ESP_BaTIN2020
Module Name	Eingebettete Systeme - Projekt
Type of Module	Elective Modules
Recognized Course	ESP - Embedded system project
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME

Learning Outcome(s)

Das Modul ESP vertieft die in der Lehrveranstaltung "Eingebettete Systeme" (ES) gewonnenen Kompetenzen zur Konzeption, zum Entwurf und zur Implementierung eingebetteter Systeme in einem Projekt, das über das ganze Semester von einem Team durchgeführt wird. Die Studierenden arbeiten an einer in Absprache mit dem Dozenten ausgewählten Idee für ein ES in den typischen Schritten

* Analyse

* technische Konzeption unter Benutzung verfügbarer Komponenten

* Implementation eines Prototyps

Sie präsentieren zu vorgegebenen Terminen ihre Arbeitsschritte und Ergebnisse

Module Contents

Project

ES Development
 Hardware Selection
 Device selection
 Understanding device descriptions (manuals)
 Application of modelling methods
 Generation of a system model
 Refinement of system components
 Modelling behavior
 Implementation
 Design of special components
 Integration of devices
 Development of tests, testing
 Building a prototype using mechanical/electrical parts

Handling of complex tasks with a team
 Project planning
 Contract fulfilling in time
 Presentation
 System design
 Intermediate work report
 Result presentation

Teaching and Learning Methods Project

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours	12 Hours $\triangleq$ 1 SWS
Self-Study	138 Hours
Recommended Prerequisites	▪ Modul ES: Kennen des typischen Aufbaus von ES Anwenden typischer Design- und Programmiermethoden für ES Erstellen eines kleinen ES mit HW- und SW-Anteil ▪ Modul SP: Ablauf eines größeren IT-Projekts Muster der Projektdurchführung mit Meilensteinen ▪ Course Embedded Systems
Mandatory Prerequisites	Project requires attendance in the amount of: 4 Termine
Recommended Literature	▪ W.Wolff: Computers as Compenents: Principles of Embedded System Design ▪ Wieringa: Design Methods for reactive Systems ▪ Gessler, Mahr: Hardware/Software Codesign
Use of the Module in Other Study Programs	ESP in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.20 FIT - Funksysteme für IoT

Module ID	FIT_BaTIN2020
Module Name	Funksysteme für IoT
Type of Module	Elective Modules
Recognized Course	FIT - Wireless Communications in the IoT
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Uwe Dettmar/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Uwe Dettmar/Professor Fakultät IME

Learning Outcome(s)

Was? Erlernen der Nutzung bestehender Funktechnologien zur Planung und Entwicklung von neuen Anwendungen im Mobilfunk und im Bereich des IoT (smart home, smart city, smart grid, smart farming etc.) sowie der Digitalisierung der Industrie (Industrie 4.0).
Womit? durch Beschäftigung mit existierenden und neu auf den Markt kommenden geeigneten funktechnischen Standards und Geräten und deren Grundlagen
Wozu? zum Entwurf, der anwendungsspezifischen Auswahl und Beurteilung von innovativen Lösungen zur Messwertaufnahme, sicheren Datenübertragung und Steuerung von Prozessen in den Bereichen des Internet-of-Things und von Industrie 4.0.

Module Contents

Lecture / Exercises

The underlying concept of this module is a combination of lecture and tutorial. After a lecture block of approximately 20 minutes the subjects taught are actively trained using Matlab/Octave and Python programs.

Syllabus:

- Introduction, What is IoT/Industry 4.0?
- Overview: Markets and application areas for wireless communications
- Standards, Basics on wireless communications
- sensor, actor and uC
- Multiple Access and data link Control in Sensor Networks
- Techniques for higher rates
- Network, Fog and Cloud Computing
- Standards for cellular (4G, 5G), WLAN, LPWAN, WWAN and WPAN

Students shall deepen their knowledge by self-study of literature and internet resources and discuss their results in small learning groups as teamwork.

By combination of taught subjects and small exercises and programs during the presence time, students are able to actively train their knowledge. More extensive problems are solved and discussed in the second part of the course to activate the student's capabilities to solve relevant problems.

Students further learn

- to analyze communication systems and to estimate their performance
- to select appropriate standards for specific applications
- to apply their knowledge to technical problems

Project

Students plan and work on projects in the field of the IoT in small teams. They use HW and SW to implement or evaluate wireless standards and to acquire, transfer, collect, present, and evaluate data, e.g., generated by sensors. The projects of different teams may be combined to a bigger project.

The results are presented at the end of the project and may be assessed and included into the total score by up to 30%.

Seminar

Alternatively: Seminar work with a hot topic from the course content. Written report will be rated.

Self conducted literature research, analysis of the sources, adequate and understandable presentation of the main aspects, discussion and assessment of the findings.

Lab

Alternatively: lab work related to single aspects of the lecture to deepen the understanding. This may include HW and SW problems.

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Project ▪ Seminar ▪ Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	68 Hours $\pm$ 6 SWS
----------------------	----------------------

Self-Study	82 Hours
-------------------	----------

Recommended Prerequisites	■ Modul SV: Bandbreitendefinition, Denken im Frequenzbereich, Fourier Transformation, Signale und Systeme Diskrete Fourier Transformation ■ Students should bring basic knowledge in digital communications, signal theory, and networks and protocols. They should further have basic skills from linear algebra and statistics. Basic programming capabilities are required, too.
----------------------------------	---

Mandatory Prerequisites	■ Project requires attendance in the amount of: 4 Termine ■ Participation in final examination only after successful participation in Project
--------------------------------	---

Recommended Literature	■ DAHLMAN, E. ; PARKVALL, S. ; SKÖLD, J. : 5G NR : the next generation wireless access technology. 1st. Elsevier Science, 2018 ■ FINKENZELLER, K. : RFID Handbuch. Hanser, 2008. ■ FÖRSTER, A. : Introduction to Wireless Sensor Networks. Wiley-IEEE Press, 2016. ■ GEIER, J. : Designing and deploying 802.11 wireless networks, Cisco Press, 2015. ■ LIAO, R. ; BELLALTA, B. ; OLIVER, M. ; NIU, Z. : MU-MIMO MAC Protocols for Wireless Local Area Networks: A Survey. In: IEEE Commun. Surv. Tutorials 18 (2016) ■ Mobile positioning and tracking : from conventional to cooperative techniques. Wiley-IEEE Press ■ TANENBAUM, A. S. ; WETHERALL, D. : Computer networks. Pearson Education, 2014
-------------------------------	---

Use of the Module in Other Study Programs	■ FIT in Bachelor Elektrotechnik 2020 ■ FIT in Bachelor Elektrotechnik und Informationstechnik 2024 ■ FIT in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.21 FSA - Formale Sprachen und Automatentheorie

Module ID	FSA_BaTIN2020
Module Name	Formale Sprachen und Automatentheorie
Type of Module	Mandatory Module
Recognized Course	FSA - Formal Languages and Automata Theory
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Hans Nissen/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Hans Nissen/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt verschiedene Kompetenzen für die Analyse, die Formalisierung und den Umgang mit formalen, abstrakten Strukturen in informationstechnischen Systemen:
 Formalisierung und Analyse von Systemen aus abstrakter Perspektive (K.1, K.2, K.3, K.4, K.5),
 Formalisierung einer gegebenen Struktur als formale Sprache (K.2, K.3),
 Spezifikation einer Grammatik zu einer gegebenen Sprache (K.2, K.3),
 Ableitung von akzeptierenden Automaten für gegebene Sprachen (K.1, K.2, K.5),
 Transformation einer Beschreibungsform einer formalen Sprachen in eine andere, äquivalente Beschreibungsform (K.1, K.2, K.4),
 Analyse einer Struktur bezüglich ihrer Komplexitäts-Eigenschaften (K.2, K.4, K.12).

Womit: Der Dozent vermittelt Wissen und Fertigkeiten in einem Vorlesungs/Übungsteil unter Verwendung einer Vielzahl von abstrakten und praktischen Beispielen.
 Die Analyse und die Formalisierung abstrakter Strukturen werden in der Vorlesung und der Übung intensiv geübt.

Wozu: Bei der Entwicklung von Software und Systemen werden sehr oft formale Strukturen, wie beispielsweise Automaten und Grammatiken, betrachtet.
 Diese Strukturen abstrahieren von der realen Welt und überführen insbesondere das dort beobachtbare bzw. gewünschte Verhalten in ein für die Implementierung geeignetes formalisiertes Modell.
 Für eine erfolgreiche Tätigkeit in der System- und Softwareentwicklung ist die sichere Abstraktion und Formalisierung von Strukturen und Verhalten aus der realen Welt erforderlich. Denn nur mit einer geeigneten Formalisierung ist die korrekte Verarbeitung von Informationsstrukturen und die korrekte Umsetzung von technischen Anwendung (z.B. zur Steuerung von Anlagen) möglich (HF.1).
 Diese Lehrveranstaltung liefert weiterhin die Kompetenzen für die Analyse von Systemen und den zugrundeliegenden Konzepten bezüglich ihrer Eignung zur Verarbeitung der geforderten Strukturen bzw. der Sicherstellung des gewünschten Verhaltens (HF.2).

Module Contents**Lecture / Exercises**

formal languages and Chomsky hierarchy
formalization of grammars
formalization of abstract machine models
finite automata
pushdown automata
turing machine
regular expressions
properties of formal languages
closure
decidability
Pumping Lemma
specify chomsky level of formal languages
specification of formal languages
develop grammar for given formal language
develop automata for given grammar
develop automata for given grammar
transform formal specifications
formale Beweise zu formalen Sprachen, Grammatiken und Automaten durchführen
formalize real world problems
develop abstract automata for real problems

Teaching and Learning Lecture / Exercises**Methods****Examination Types with** cf. exam regulations**Weights****Workload** 150 Hours**Contact Hours** 34 Hours $\pm$ 3 SWS**Self-Study** 116 Hours**Recommended** no requirements**Prerequisites****Mandatory
Prerequisites**

Recommended Literature	■ Uwe Schöning: Theoretische Informatik - kurzgefasst, Spektrum Akademischer Verlag, 5. Auflage, 2008
	■ Rolf Socher: Theoretische Grundlagen der Informatik Carl Hanser Verlag, 2007
	■ Gottfried Vossen, Kurt-Ulrich Witt: Grundkurs Theoretische Informatik 4. Auflage, Vieweg Verlag, 2006
	■ John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie 3. Auflage, Pearson Studium, 2011

**Use of the Module in
Other Study Programs** FSA in Bachelor Informatik und Systems-Engineering 2024**Specifics and Notes****Last Update** 19.7.2025, 14:32:16

6.22 GRT - Graphentheorie

Module ID	GRT_BaTIN2020
Module Name	Graphentheorie
Type of Module	Elective Modules
Recognized Course	GRT - Graph theory
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Hubert Randerath/Professor Fakultät IME
Lecturer(s)	

Learning Outcome(s)

- (WAS) Studierende können bestimmte Problemstellungen der Informatik abstrakt modellieren und auf Basis dieser Modelle gängige und wohlverstandene Methoden und Algorithmen zur Lösung anwenden
 - (WOMIT) indem sie
 - die Problemstellung als Graph darstellen,
 - die Graphen klassifizieren und anwendbare Algorithmen identifizieren und
 - diese Algorithmen implementieren,
 - (WOZU) damit sie effektive und effiziente Lösungen für wiederkehrende Anforderungen wie beispielsweise Wegfindungen, Optimierung von (Netzwerk-)Routing oder Ermittlungen von Abhängigkeiten in Programmen realisieren können.
-

Module Contents

Lecture

- Basics of combinatorics and asymptotic analysis
 - Basic combinatorial principles
 - Elementary counting principles
 - Permutations
 - Variations and combinations
 - Principle of inclusion and exclusion
 - Growth of functions
 - Runtime of algorithms
 - Complexity classes.
- Basics of graph theory
 - Graph definition
 - Complete and bipartite graphs
 - Isomorphism of graphs
 - Adjacency and incidence matrix
 - Paths and circles
 - Connection of graphs
 - directed graphs
 - forests
 - trees
 - elementary graph parameters
- Traversal in graphs and the shortest path problem
 - breadth-first search
 - Depth-first search
 - Topological sorting
 - Euler tours
 - Hamiltonian circles
 - Dijkstra algorithm
 - Bellman-Ford algorithm.
- Matchings and flows
 - Matchings
 - Hungarian method
 - Edmonds algorithm
 - Networks
 - Flows
 - Permissible flows
 - Max-Flow-Min-Cut
 - Ford-Fulkerson algorithm
 - Edmonds-Karp algorithm
- colorings
 - Node colorings
 - Edge coloring
 - List colorings
 - Tournaments and game plans
 - Planar graphs
 - Four-color set
 - Perfect graphs
 - Coloring algorithms

Exercises

- Students solve graph-theoretical and graph-algorithmic problems with Maple

undefined

Teaching and Learning Methods	▪ Lecture ▪ Exercises ▪ undefined
--------------------------------------	---

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	NaN Hours $\triangleq$ NaN SWS
----------------------	--------------------------------

Self-Study	NaN Hours
-------------------	-----------

Recommended Prerequisites	
----------------------------------	--

Mandatory Prerequisites	Exercises requires attendance in the amount of: 2 Termine
--------------------------------	---

Recommended Literature	▪ Diskrete Mathematik, M. Aigner, Vieweg-Verlag ▪ Graphentheorie, R. Diestel, Springer Verlag ▪ Graphentheorie, P. Tittman, Fachbuchverlag Leipzig ▪ Graphen an allen Eckern und Kanten, L. Volkmann, RWTH Aachen
-------------------------------	---

Use of the Module in Other Study Programs	GRT in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes	
----------------------------	--

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.23 GSP - Grundlagen der Systemprogrammierung

Module ID	GSP_BaTIN2020
Module Name	Grundlagen der Systemprogrammierung
Type of Module	Mandatory Module
Recognized Course	GSP - Fundamentals in System Programming
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME
Lecturer(s)	- Prof. Dr. Tobias Krawutschke/Professor Fakultät IME - Norbert Kellersohn/Lehrkraft für besondere Aufgaben

Learning Outcome(s)

Was: Das Modul vermittelt die Kompetenzen zur Verwendung von Mikrocontrollersystemen (Hardware inkl. Echtzeitbetriebssystem) für die Erarbeitung von Problemlösungen aus dem Bereich Messen-Steuern-Regeln. Darüber hinaus werden die Studierenden in die Lage versetzt eigene Treiber für vom Betriebssystem nicht unterstützte Hardwarekomponenten zu entwickeln und zu nutzen. Alle Implementierungen erfolgen in der Programmiersprache C, wodurch die Einführung in diese Programmiersprache ein integraler Bestandteil dieses Moduls ist. Die Studierenden werden in die Lage versetzt, fachspezifische Begriffe, Tools und Techniken im praktischen Umfeld sicher anzuwenden. Aufbauend auf den in der Vorlesung vermittelten Kenntnissen werden komplexere Problemstellungen analysiert, auf Teilsysteme heruntergebrochen und modelliert. Darauf aufbauend wird die Problemlösung mittels Entwurfswerkzeugen implementiert und am Zielsystem in Betrieb genommen.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. Im Praktikum erarbeiten die Studierenden in Kleingruppen Problemlösungen und verteidigen diese.

Wozu: Kompetenzen in der Verwendung von Mikrocontrollern nebst Echtzeitbetriebssystem sind essentiell für technische Informatiker, die im HF 1 arbeiten wollen. Durch die Entwicklung von Problemlösungen erwerben die Studierenden zudem Erfahrungen, die essentiell für das HF 2 sind. Eine projektorientierte Durchführung der Praktika in kleinen Teams mit dem Dozenten als "Auftraggeber" initiiert die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents

Lecture / Exercises

Lab

Teaching and Learning Methods	- Lecture / Exercises - Lab
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	45 Hours $\pm$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	basic skills in procedural programming structure and mode of operation of a simple computer basics in digital systems finite state machines and state transition diagrams

Mandatory Prerequisites	Lab requires attendance in the amount of: 2 Termine
Recommended Literature	▪ Martin: Rechnerarchitektur, Fachbuchverlag Leipzig (Carl Hanser) ▪ Oberschelp/Vossen: Rechneraufbau und Rechnerstrukturen, Oldenbourg Verlag ▪ Vogt, C: C für Java-Programmierer ▪ Tanenbaum, Goodman: Computerarchitektur, Pearson Studium (Prentice Hall)
Use of the Module in Other Study Programs	GSP in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.24 GUI - Graphische Oberflächen und Interaktion

Module ID	GUI_BaTIN2020
Module Name	Graphische Oberflächen und Interaktion
Type of Module	Mandatory Module
Recognized Course	GUI - Graphical User Interfaces
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Andreas Behrend/Professor Fakultät IME
Lecturer(s)	Ali Shafieian/wissenschaftlicher Mitarbeiter Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt Kompetenzen zur Programmierung von grafischen Benutzeroberflächen. In praktischer Arbeit analysieren die Studierenden Problemstellungen (K2), implementieren Lösungen mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7). Sie recherchieren dazu in Online-Dokumentationen (K8, K15) und passen vorhandene Software an (K10).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K8, K13).

Wozu: Grafische Benutzeroberflächen spielen im Programmierumfeld eine zentrale Rolle und somit auch Kenntnisse, sie zu programmieren und in bestehende Systeme zu integrieren (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden zudem weitere Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents

Lecture / Exercises

fundamental terms and techniques of graphical user interfaces within microsoft windows
concepts of windows and messages

using of OS-given APIs in Java
including of AWT and Swing
discussion of their pros and cons

using of Microsofts WinAPI

using of AWT und Swing classe for grafical user interfaces in Java

Lab

programming of grafical user interfaces as individual tasks - details see "Vorlesung/Übung"

using programming environments for guis

implementation of gui applications of medium complexity in small teams

Teaching and Learning Methods	■ Lecture / Exercises ■ Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
Recommended Prerequisites	▪ Modul PI2: Sicherer Umgang mit einer objektorientierten Programmiersprache. ▪ Modul AD: Tieferes Verständnis der Verwendung von C++ und Java bei der Analyse von Algorithmen ▪ procedural and object-oriented programming in C/C++ and Java structure and functions of operating systems
Mandatory Prerequisites	Lab requires attendance in the amount of: 3 Termine
Recommended Literature	▪ keine
Use of the Module in Other Study Programs	GUI in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.25 HF - Hochfrequenztechnik

Module ID	HF_BaTIN2020
Module Name	Hochfrequenztechnik
Type of Module	Elective Modules
Recognized Course	HF - High Frequency Technologies
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Rainer Kronberger/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Rainer Kronberger/Professor Fakultät IME
Learning Outcome(s)	
Die Studierenden können hochfrequenztechnische Probleme lösen, indem sie hierfür geeignete spezielle Methoden der Elektrotechnik anwenden, um später hochfrequenztechnische Schaltungen, Baugruppen zu analysieren, entwickeln und herzustellen. Die Studierenden können hochfrequenztechnische Schaltungen entwickeln, indem sie hierfür geeignete spezielle Methoden der Elektrotechnik und Elektronik anwenden, um später hochfrequenztechnische Systeme zu entwickeln und herzustellen.	
Module Contents	
Lecture	
Students will learn fundamentals in high frequency technologies in theory and application Introduction to frequency range and high frequency systems Linear, passive circuits with inductances and capacitors Currents, voltages and power in passive high frequency circuits Smith-Diagram Resonance circuits and filters Transmission line theory and application Impedance transformation circuits Scattering parameters and matrices	
Exercises / Lab	
Exercises and practical work in close relationship to lecture .Students will learn fundamentals in high frequency technologies in theory and application Students will learn fundamentals in high frequency technologies in theory and application	
Teaching and Learning Methods	▪ Lecture ▪ Exercises / Lab
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	45 Hours $\pm$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	▪ Modul EG: Grundlegende Kenntnisse der Elektrotechnik Grundlegende Kenntnisse der Wechselstromtechnik Grundlegende Kenntnisse zu elektrischen und magnetischen Feldern ▪ Modul MA1: Grundlegende Kenntnisse der Mathematik ▪ GE1-GE3, MA1, MA2

Mandatory Prerequisites	Exercises / Lab requires attendance in the amount of: Praktikumstermine
Recommended Literature	■ Meinke/ Gundlach: Taschenbuch der Hochfrequenztechnik Bd. 1-3 Springer Verlag Zinke/ Brunswig: Hochfrequenztechnik 1, Filter, Leitungen, Anten-nen, Springer Verlag Detlefsen/Siart: Grundlagen der HF-Technik. Oldenbourg Verlag
Use of the Module in Other Study Programs	■ HF in Bachelor Elektrotechnik 2020 ■ HF in Bachelor Elektrotechnik und Informationstechnik 2024 ■ HF in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.26 IAK - Ingenieurakustik

Module ID	IAK_BaTIN2020
Module Name	Ingenieurakustik
Type of Module	Elective Modules
Recognized Course	IAK - Acoustics for Engineers
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr.-Ing. Christoph Pörschmann/Professor Fakultät IME
Lecturer(s)	Prof. Dr.-Ing. Christoph Pörschmann/Professor Fakultät IME

Learning Outcome(s)

Was: Durch das Modul lernen die Studierenden die grundlegenden Konzepte und physikalischen Zusammenhänge der Akustik kennen werden in die Lage versetzt, diese zu beschreiben, zu analysieren und die Auswirkungen veränderter Einflussgrößen abzuschätzen.

Womit: Durch das Verständnis und die Anwendung der in der Vorlesung präsentierten Grundlagen erlernen die Studierenden, wie sich Schall ausbreitet, wie er erzeugt wird und welche physikalischen Phänomene dabei eine Rolle spielen. Ein weiteres Verständnis der grundlegenden Zusammenhänge wird durch das Praktikum bewirkt, in dem die Studierenden selbst Messungen vornehmen und relevante Parameter bestimmen. Sie erlernen somit, die physikalischen Zusammenhänge zu den entsprechenden Modellen und Kennziffern in Beziehung setzen.

Wozu: Akustische Zusammenhänge spielen im Alltag eines Ingenieurs an vielen Stellen eine wesentliche Rolle, vom Lärmschutz, über Grundprinzipien der Schallausbreitung in Räumen. Für medientechnische Systeme und Medienprodukte spielt die gezielte Anregung und kontrollierte Ausbreitung von Schall eine große Rolle. Die Veranstaltung vermittelt hierzu die nötigen Grundkenntnisse und Aufbaukenntnisse.

Module Contents**Lecture / Exercises**

Introduction of the basic acoustic parameters
 Sound pressure, sound velocity, flow, power
 Logarithmic quantities and levels

Mechanical and acoustic vibration systems
 Mechanical vibration systems
 Acoustic vibration systems

Sound propagation in the room
 Homogeneous plane wave
 standing waves
 resonance systems
 diffraction, refraction, reflection

point sources
 Behaviour of sound pressure and sound velocity
 elementary radiator synthesis

Concepts of loudspeakers and microphones)
 Principles of directional microphones
 Electrodynamic microphones and headphones
 Piezoelectric microphones and headphones
 Dielectric microphones

absorbers
 Porous Absorbers
 Helmholtz resonators as absorbers
 plate absorber

Analysis and description of systems with loudspeakers and microphones

Calculation and description of the entire sound propagation chain from the microphone via the mechanoelectric conversion, the transmission via a communication link as well as the conversion via an electromechanical transducer and the sound radiation.

Lab

Simulation of the propagation of sound waves

Investigation of sound reflection on Kundt's tube

Analysis of eigenmodes

perform practical measurements based on described tasks

create functional measurement setups

prepare adequate documentation for measurements carried out

Evaluate and discuss measurement results

Teaching and Learning ■ Lecture / Exercises
Methods ■ Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites Basic knowledge mechanics
 knowledge time domain / frequency domain operations
 Calculations with real and imaginary values
 Basic knowledge integral and differential mathematics

Mandatory Prerequisites	■ Lab requires attendance in the amount of: 3 Praktikumstermine und 1 Hörversuch ■ Participation in final examination only after successful participation in Lab
Recommended Literature	■ Boré, G., Peus, S. (1999). „Mikrophone für Studio und Heimstudio-Anwendungen – Arbeitsweise und Ausführungsbeispiele,“ Hrsg. Georg Neumann GmbH, Berlin. ■ Blauert, J., Xiang, N. (2008). „Acoustic for Engineers – Troy Lectures,“ Springer Verlag, Heidelberg. ■ Blauert, J., Braasch, J., Jekosch, U. (2012). „Acoustics for Communication – Dresden Lectures,“ Springer Verlag Heidelberg, in Vorbereitung. ■ Dickreiter, M., Hoeg, W., Dittel, V., Wöhr, M. (2008). „Handbuch der Tonstudioteknik,“ 7. Auflage, Saur Verlag, München. ■ Görne, T. (2011). „Tontechnik,“ Hanser Verlag München. ■ Kuttruff, H. (2004). „Akustik – Eine Einführung,“ S. Hirzel Verlag, Stuttgart. ■ Cremer, L. (1976). „Vorlesungen über Technische Akustik,“ Springer Verlag, Berlin, Heidelberg. ■ Lord Rayleigh (1896). „The Theory of Sound,“ 2nd Edition 1896, Dover Publ. New York. ■ Müller, G., Möser, M. (2004). „Taschenbuch der Technischen Akustik,“ Springer Verlag Berlin, 3. Auflage. ■ Veit, I. (2005). „Technische Akustik,“ Kamprath-Reihe, Vogel-Verlag, Würzburg. ■ Weinzierl, Stefan (2008). „Handbuch der Audiotechnik,“ Springer Verlag, Berlin. ■ Blauert, J., (2005) „Communication Acoustics,“ Springer Verlag Heidelberg, ■ Blauert, J., (2021) „Acoustics for Communication,“ Springer Verlag Heidelberg, upcoming
Use of the Module in Other Study Programs	■ IAK in Bachelor Elektrotechnik 2020 ■ IAK in Bachelor Elektrotechnik und Informationstechnik 2024 ■ IAK in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.27 IBV - Industrielle Bildverarbeitung

Module ID	IBV_BaTIN2020
Module Name	Industrielle Bildverarbeitung
Type of Module	Elective Modules
Recognized Course	BV1 - Image Processing
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Jan Salmen/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Jan Salmen/Professor Fakultät IME
Learning Outcome(s)	
Was: Die Studierenden erlernen den grundlegenden Aufbau und den software-technischen Zugriff auf digitale Bilder sowie die Methoden und Verfahren der digitalen Bildverarbeitung zur Bildverbesserung, Farbbildverarbeitung und Vermessung von Bildinhalten. Sie erlernen eine komplexe Problemstellung aus dem Bereich der Bildverarbeitung zu analysieren, auf Teilsysteme herunter zu brechen und geeignete Methoden und Verfahren auszuwählen. Darauf aufbauend wird die Problemlösung mittels Entwurfswerkzeugen implementiert und validiert. Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. Im Praktikum erarbeiten die Studierenden in Kleingruppen Problemlösungen und verteidigen diese. Wozu: Die Studierenden werden in dem Modul befähigt, im industriellen Umfeld Problemlösung berührungsloser Inspektions- und Meßaufgaben mittels Bildverarbeiten zu lösen und bestehende Lösungen zu bewerten.	
Module Contents	
Lecture	
Lab	
Teaching and Learning Methods	■ Lecture ■ Lab
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	34 Hours $\pm$ 3 SWS
Self-Study	116 Hours
Recommended Prerequisites	■ Modul PI1: Entwickeln von Programmen zur Lösung konkreter Problemstellungen, abstrahieren von Problembeschreibungen in Algorithmen und überprüfen von Programmen auf Fehler. ■ Modul MA1: Problemlösungskompetenz aus dem Bereich lineare Algebra und der Analysis. Sicheres Beherrschen der entsprechenden Symbole und Formalismen ■ Modul SV: Darstellung von zeitdiskreten Signalen im Zeit und Frequenzbereich (DFT). ■ Basic course mathematics ■ Basic course computer science ■ Basic course signal theory
Mandatory Prerequisites	Lab requires attendance in the amount of: 4 Fachgespräche

Recommended Literature

- Burger/Burge: Digitale Bildverarbeitung

Use of the Module in Other Study Programs

- IBV in Bachelor Elektrotechnik 2020
- BV in Bachelor Elektrotechnik und Informationstechnik 2024
- BV1 in Bachelor Medientechnologie 2020
- BV in Bachelor Medientechnologie 2024
- BV in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update

19.7.2025, 14:32:16

6.28 IOT - IoT Protokolle und Anwendungen

Module ID	IOT_BaTIN2020
Module Name	IoT Protokolle und Anwendungen
Type of Module	Elective Modules
Recognized Course	IOT - IoT Protocols and Applications
ECTS credits	5
Language	deutsch und englisch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Harald Elders-Boll/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Harald Elders-Boll/Professor Fakultät IME
Learning Outcome(s)	
Kennen und Anwenden die wichtigsten Protokolle, Anwendungen und Datenanalyse und Sicherheitstechniken für das Internet der Dinge (IoT) sowie der Digitalisierung der Industrie (Industrie 4.0) durch Vermittlung der zugrundeliegenden Methoden und Konzepte und deren Anwendung in Praktikumsaufgaben zur Vernetzung und Sicherheit von IoT-Endgeräten, zum Entwurf, der anwendungsspezifischen Auswahl und Beurteilung von innovativen und sicheren Anwendungen in den Bereichen des Internet of Things und der Industrie 4.0.	
Module Contents	
Lecture / Exercises	
The underlying concept of this module is a combination of lecture and tutorial. After a lecture block the subjects taught are actively trained by solving corresponding problems. Syllabus: - Introduction to IoT - Applications of IoT - Hard- and Software Fundamentals for IoT - IoT System and Architectures - IoT Communications Protocols - IoT Application Protocols (MQTT, CoAP, HTTP, REST) - Data Analytics and Machine Learning for IoT - IoT Security Distinguish different IoT architectures. Analyse IoT system using suitable tools. Connect IoT end devices to IoT systems. Assess the security of IoT systems.	
Lab	
- Connect sensors and actuators to microprocessors and single-board computers - Connect IoT devices to the cloud - Transmit measurement data to the cloud - Compromise hard- and software of IoT devices - Sniffing the communication of IoT devices	
Teaching and Learning Methods	■ Lecture / Exercises ■ Lab
Examination Types with Weights	cf. exam regulations
Workload	150 Hours

Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	▪ Modul NP: Grundlagen von Rechnernetzen Netzwerkanwendungen und Protokolle Grundlagen der Transportschicht Adressierung und Routing Grundlagen der Sicherungsschicht Grundlagen der Netzwerksicherheit ▪ Fundamentals of Computer Networks Network application and Protocols Transport Layer Fundamentals Link Layer Fundamentals Fundamentals of Network Security
Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 4 Termine ▪ Participation in final examination only after successful participation in Lab
Recommended Literature	▪ P. Lea, "Internet of Things for Architects", Pakt, 2018 ▪ A. Bahga, V. Madiseti, "Internet of Things A Hands-on Approach", Bagha & Madiseti ▪ B. Adyan, D. Obermaier, P. Fremantle, "The Technical Foundations of IoT", Artech House, 2017
Use of the Module in Other Study Programs	▪ IOT in Bachelor Elektrotechnik 2020 ▪ IOT in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ IOT in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.29 IPM - IT-Projektmanagement

Module ID	IPM_BaTIN2020
Module Name	IT-Projektmanagement
Type of Module	Mandatory Module
Recognized Course	IPM - Management of Projects in Information Technology
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Chunrong Yuan/Professor Fakultät IME
Lecturer(s)	- Prof. Dr. Chunrong Yuan/Professor Fakultät IME - Dr. Nikolaus Schmidt/Lehrbeauftragter

Learning Outcome(s)

Was: Das Modul vermittelt Kompetenzen zur Organisation und Verwaltung von IT Projekten in allen relevanten Arbeitsschritten von der Anforderungsanalyse (K.1), Projektplanung und Aufwandschätzung (K.4), Risikomanagement und Personalführung (K.9), Qualitätsmanagement und Fortschrittskontrolle (K.11), bis zu Projektabschluss, so dass die Studierenden in der Lage sind, IT-gestütztes Projektmanagement und -Governance (K.9, K.11) für eigene IT-Projekte anwenden zu können.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs/Übungsteil und die Studierenden erstellen jeweils individuell ein Konzept für die Planung und Management eines eigenen IT-Projekts.

Wozu: Kompetenzen in der Planung, Organisation und Prozessmanagement von informationstechnischen Systemen und Projekten sind essentiell für technische Informatiker*innen, die im HF3 arbeiten wollen. Durch die Planungsarbeit am eigenen IT-Projekt erwerben die Studierenden zudem Erfahrungen, die essentiell für das HF 1 und 2 sind, u.a. Anforderungen erfassen, Konzepte zu technischen Lösungen entwickeln, das zu entwickelnde Produkt bzw. System analysieren und bewerten. Die Durchführung mit beteiligten Stakeholdern vom Projekt stärkt die Vermittlungs- und Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents

Project

Planning of the bachelor thesis as an IT project together with written documentations, presentation and oral defending.

Seminar-style Teaching

Teaching and Learning Methods	- Project - Seminar-style Teaching
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	34 Hours $\pm$ 3 SWS
Self-Study	116 Hours
Recommended Prerequisites	Experience of software development
Mandatory Prerequisites	
Recommended Literature	Hindl et. al.: Basiswissen Softwareprojektmanagement, dpunkt.verlag, 2009.

**Use of the Module in
Other Study Programs**

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.30 ITS - IT-Sicherheit

Module ID	ITS_BaTIN2020
Module Name	IT-Sicherheit
Type of Module	Mandatory Module
Recognized Course	ITS - IT Security
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Heiko Knospe/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Heiko Knospe/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt die grundlegenden Konzepte und Verfahren der IT-Sicherheit, die für viele IT-Systeme und Anwendungen eine wichtige Rolle spielen (K. 4). Die Studierenden lernen die Analyse von Systemen in Bezug auf Sicherheitsanforderungen (K. 7). Hierfür ist ein Verständnis von Sicherheitsbedrohungen und Angriffen notwendig. Die Studierenden lernen die grundlegenden Verfahren und Standards der IT-Sicherheit um Systeme zu entwerfen, zu realisieren und zu prüfen (K. 8, K. 9, K. 10). Ethische Grundwerte spielen in diesem Zusammenhang eine wichtige Rolle (K. 18), z.B. beim Umgang mit personenbezogenen Daten, Womit: Der Dozent/die Dozentin vermittelt Wissen und Basisfertigkeiten in der Vorlesung. In der Übung bearbeiten die Studierenden unter Anleitung Aufgaben. Im Praktikum werden konkrete Probleme und Fragestellungen der IT-Sicherheit bearbeitet.

Wozu: Grundlegende Kenntnisse der IT-Sicherheit werden in mehreren Modulen des Studiengangs verwendet und sind anerkannter Teil der Basisausbildung in technischen Fächern (HF 1). Bei der Planung von Systemen für technische Anwendungen, der Analyse und Bewertung von Anforderungen sowie der Administration von IT-Systemen spielen Fragen der IT-Sicherheit heute eine wichtige Rolle (HF 5). Die Sicherheit von IT-Systemen ist Teil der Qualitätskontrolle und kann auch in Zertifizierungsprozessen von Bedeutung sein (HF 2).

Module Contents**Lecture / Exercises**

IT security fundamentals: standards and guidelines, taxonomy, security objectives, threats, risks, attacks, security measures.

Cryptographic methods: mathematical fundamentals, definitions of security, historical ciphers, symmetric encryption, block ciphers, operating modes, stream ciphers, hash methods, message authentication codes, asymmetric encryption, RSA, key agreement, Diffie-Hellman, signatures.

Authentication, key agreement and access control: authentication methods, passwords, key agreement, protocols, public keys and public key infrastructures (PKI), access control strategies, access matrix, Unix ACL.

Network security: TLS and SSH protocols, DNS security.

Software and web security: Basic principles and design of secure software, vulnerabilities, attacks against web applications.

Security management: risk management, security process, security standards, in particular ISO 27000 series and German IT baseline protection, data protection (privacy), laws, ethical aspects.

Lab

- Fundamentals of cyber security (e-learning).
- Development of Java software for AES encryption and decryption of files.
- Use of different operating modes for block ciphers.
- Statistical analysis of an AES ciphertext.
- Generation of key pairs, certificates and development of a public key infrastructure with open source software (optional).
- Using a Linux system for penetration testing and digital forensics (Kali Linux).
- Attacks against weak passwords.
- Attacks against web applications (test system).
- Use of software for detecting and analyzing vulnerabilities.

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	▪ Modul PI1: - ▪ Modul PI2: - ▪ Modul NP: - ▪ Modul MA1: - ▪ Modul MA2: - ▪ -
----------------------------------	--

Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 3 Termine ▪ Participation in final examination only after successful participation in Lab
--------------------------------	--

Recommended Literature

- C. Eckert, IT-Sicherheit, Oldenbourg Verlag
- D. Gollmann, Computer Security, John Wiley & Sons
- J. Schwenk, Sicherheit und Kryptographie im Internet, Springer Verlag
- G. Schäfer, M. Roßberg, Netzsicherheit, dpunkt Verlag
- W. Stallings, L. Brown, Computer Security: Principles and Practice, Pearson
- N. Pohlmann, Cyber-Sicherheit, Springer Verlag
- H. Knospe, A Course in Cryptography, American Mathematical Society
- H. Kersten, G. Klett, J. Reuter, K.-W. Schröder, IT-Sicherheitsmanagement nach der neuen ISO 27001. Springer.
- C. Paar, J. Pelzl, Kryptografie verständlich, Springer.
- P. C. van Oorschot, Computer Security and the Internet, Springer.
- C. Pfleeger et al., Security in Computing, Pearson
- J. Schwenk, Guide to Internet Cryptography, Springer

Use of the Module in Other Study Programs

- ITS in Bachelor Elektrotechnik 2020
- ITS in Bachelor Elektrotechnik und Informationstechnik 2024
- ITS in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update

19.7.2025, 14:32:16

6.31 KOAK - Kommunikationsakustik

Module ID	KOAK_BaTIN2020
Module Name	Kommunikationsakustik
Type of Module	Elective Modules
Recognized Course	KOAK - Communication Acoustics
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr.-Ing. Christoph Pörschmann/Professor Fakultät IME
Lecturer(s)	Prof. Dr.-Ing. Christoph Pörschmann/Professor Fakultät IME

Learning Outcome(s)

Was: Durch das Modul lernen die Studierenden, die grundlegenden Konzepte und physikalischen Zusammenhänge der Akustik auf diverse Anwendungen zu beziehen. Sie werden in die Lage versetzt, diese Anwendungen zu beschreiben, zu analysieren und die Auswirkungen veränderter Randbedingungen abzuschätzen.

Womit: Durch das Anwenden der Grundlagen auf diverse Problemstellungen verstehen die Studierenden viele praktische Anwendungen der Akustik. Ein weiteres Verständnis der grundlegenden Zusammenhänge wird durch das Praktikum bewirkt, in dem die Studierenden selbst einige einfache Anwendungen nutzen, erweitern und einsetzen.

Wozu: Akustische Zusammenhänge spielen im Alltag eines Ingenieurs an vielen Stellen eine wesentliche Rolle, vom Lärmschutz, über Grundprinzipien der Schallausbreitung in Räumen. Für medientechnische Systeme und Medienprodukte spielt die gezielte Anregung und kontrollierte Ausbreitung von Schall eine große Rolle. Die Veranstaltung vermittelt hierzu die Anwendungskennnisse.

Module Contents**Lecture / Exercises**

- | Room simulation method and software tools required therefor
- | Human hearing system, basic phenomena of auditory perception, psychoacoustic parameters
- | Human spatial perception
- | Principles of human speech generation, common methods of speech signal processing
- | Analyze and solve problems of sound insulation and noise.
- | relate psychoacoustic quantities to physical quantities
- | Analysis and adaptation of spatial sound reinforcement systems

Lab

- | Reverberation Time Measurement
- | Room simulation
- | Audiometry (Threshold of silence)
- | perform practical measurements based on described tasks
- | create functional measurement setups
- | prepare adequate documentation for measurements carried out
- | Evaluate and discuss measurement results

- Teaching and Learning Methods**
- Lecture / Exercises
 - Lab

Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	Basic knowledge mechanics knowledge time domain / frequency domain operations Calculations with real and imaginary values Basic knowledge integral and differential mathematics Basic knowledge Acoustics
Mandatory Prerequisites	■ Lab requires attendance in the amount of: 3 Praktikumstermine und 1 Hörversuch ■ Participation in final examination only after successful participation in Lab
Recommended Literature	■ Blauert, J.,(2005) „Communication Acoustics,“ Springer Verlag Heidelberg ■ Weinzierl, Stefan (2008). „Handbuch der Audiotechnik,“ Springer Verlag, Berlin. ■ Blauert, J.,(2021) „Acoustics for Communication,“ Springer Verlag Heidelberg, upcoming ■ Veit, I. (2005). „Technische Akustik“, Kamprath-Reihe, Vogel-Verlag, Würzburg. ■ Cremer, L. (1976). „Vorlesungen über Technische Akustik,“ Springer Verlag, Berlin, Heidelberg. ■ Kuttruff, H. (2004). „Akustik – Eine Einführung,“ S. Hirzel Verlag, Stuttgart.
Use of the Module in Other Study Programs	■ KOAK in Bachelor Elektrotechnik 2020 ■ KOAK in Bachelor Elektrotechnik und Informationstechnik 2024 ■ KOAK in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.32 KOLL - Kolloquium zur Bachelorarbeit

Module ID	KOLL_BaTIN2020
Module Name	Kolloquium zur Bachelorarbeit
Type of Module	Mandatory Module
Recognized Course	BAKOLL - Colloquium
ECTS credits	3
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	7
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Bachelor Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	verschiedene Dozenten*innen / diverse lecturers

Learning Outcome(s)

WAS:

Fachliche und außerfachliche Bezüge der eigenen Arbeit darstellen, bewerten und begründen.

WOMIT:

Präsentationstechniken (schriftlich als auch mündlich) sowie kritische Reflexion der eigenen Arbeitsergebnisse

WOZU:

Um eigene Lösungswege und gewonnene Erkenntnisse vor Fachpublikum darstellen, bewerten und diskutieren zu können.

WAS:

Eigene Arbeitsweise und Ergebnisse präsentieren.

WOMIT:

Präsentationstechniken (schriftlich als auch mündlich) sowie sowie kritische Reflexion der eigenen Arbeitsweise.

WOZU:

Um eigene Lösungswege und gewonnene Erkenntnisse vor Fachpublikum darstellen, bewerten und diskutieren zu können.

Module Contents

Colloquium

The colloquium serves to determine whether the student is able to present the results of the Bachelor's thesis, its technical and methodological foundations, interdisciplinary contexts and extracurricular references orally, to justify them independently and to assess their significance for practice

Teaching and Learning Methods Colloquium

Examination Types with Weights cf. exam regulations

Workload 90 Hours

Contact Hours 0 Hours $\pm$ 0 SWS

Self-Study 90 Hours

Recommended Prerequisites

- Mandatory Prerequisites**
- Module BAA: Die Bachelorarbeit muss abgeschlossen sein, damit sie im Kolloquium ganzheitlich und abschließend präsentiert werden kann.
 - See exam regulations §29, paragraph 2

**Recommended
Literature**

Use of the Module in Other Study Programs	▪ KOLL in Bachelor Elektrotechnik 2020 ▪ KOLL in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ KOLL in Bachelor Medientechnologie 2020 ▪ KOLL in Bachelor Medientechnologie 2024 ▪ KOLL in Bachelor Optometrie 2021 ▪ KOLL in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes	See also examination regulations §29.
----------------------------	---------------------------------------

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.33 MA1 - Mathematik 1

Module ID	MA1_BaTIN2020
Module Name	Mathematik 1
Type of Module	Mandatory Module
Recognized Course	MA1 - Mathematics 1
ECTS credits	10
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Heiko Knospe/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Heiko Knospe/Professor Fakultät IME ▪ Prof. Dr. Hubert Randerath/Professor Fakultät IME ▪ Prof. Dr. Beate Rhein/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt die grundlegenden Konzepte und Methoden der Mathematik, die in der Informatik und Technik benötigt werden (K. 3). Die Abstraktion und mathematischen Formalisierung von Problemen soll erlernt und angewendet werden (K. 2). Die Studierenden lernen in der Mathematik die Grundzüge wissenschaftlichen Arbeitens kennen (K. 12).

Womit: Der Dozent/die Dozentin vermittelt Wissen und Basisfertigkeiten in der Vorlesung. In der Übung bearbeiten die Studierenden unter Anleitung Aufgaben. Die Übung wird durch Hausaufgaben und Online-Aufgaben (E-Learning) ergänzt. Zusätzlich findet ein Tutorium statt.

Wozu: Grundlegende Mathematik-Kenntnisse werden in mehreren Modulen des Studiengangs benötigt und sind anerkannter Teil der Basisausbildung. Mathematische Methoden sind essentiell für Informatiker, die Systeme zur Verarbeitung, Übertragung und Speicherung von Informationen für technische Anwendungen planen, realisieren und integrieren (HF 1). Die Analyse und Bewertung von Anforderungen, Konzepten und Systemen erfordert häufig mathematische Methoden (HF 2).

Module Contents**Lecture / Exercises**

Fundamentals

- Sets, numbers, sums, products, factorial, binomial coefficients
- Real numbers, order, intervals, completeness
- Propositional logic
- Induction
- Maps and their properties
- Real functions, boundedness, monotonicity, inverse function

Elementary functions

- Polynomials and rational functions
- Power function, root function, exponential and logarithmic functions
- Trigonometric functions

Sequences, series and continuity

- Real sequences and limits
- Series and (optional) convergence criteria
- Power series and (optional) radius of convergence
- Limits of function values
- Continuity and properties of continuous functions
- Asymptote

Differential calculus

- Differentiability and derivation
- Derivation rules
- Higher derivatives
- Extreme points and curve discussion
- Taylor polynomial, Taylor series
- Newton method
- Rule of de l'Hospital

Vectors, matrices and linear systems of equations

- Vector calculus in $\mathbb{R}^n$
- Scalar product
- Vector product
- Lines
- Planes
- Matrices and computations
- Linear systems of equations and Gaussian algorithm
- Linear independence, generating set and basis
- Rank of a matrix
- Quadratic matrices and invertible matrices
- Determinant
- Cramer's rule (optional)

Complex numbers

- Normal form and calculation rules
- Polar and exponential form
- Complex sequences, series, functions, power series, Euler's formula
- Powers and roots of complex numbers

Exercises / Lab

Online maths course OMB+ with the contents:

- Numbers, fractions
- Roots, powers, proportionality
- Equations in one unknown variable

Teaching and Learning ■ Lecture / Exercises

Methods ■ Exercises / Lab

Examination Types with Weights	cf. exam regulations
Workload	300 Hours
Contact Hours	57 Hours $\triangleq$ 5 SWS
Self-Study	243 Hours
Recommended Prerequisites	-
Mandatory Prerequisites	Participation in final examination only after successful participation in Exercises / Lab
Recommended Literature	▪ P. Hartmann, Mathematik für Informatiker, vieweg Verlag ▪ T. Westermann, Mathematik für Ingenieure, Springer Verlag ▪ T. Rießinger, Mathematik für Ingenieure, Springer Verlag ▪ M. Knorrenschild, Mathematik für Ingenieure 1, Hanser Verlag ▪ W. Schäfer, G. Trippler, G. Engeln-Müllges (Hrg.), Kompaktkurs Ingenieurmathematik, Fachbuchverlag Leipzig ▪ L. Papula, Mathematik für Ingenieure und Naturwissenschaftler, Band 1 und 2, Vieweg+Teubner Verlag ▪ G. Hoever, Höhere Mathematik kompakt, Springer Verlag ▪ O. Forster, Analysis 1, Vieweg Verlag ▪ C. Blatter, Analysis 1, Springer Verlag ▪ hm4mint.nrw, Online-Kurs Höhere Mathematik 1 ▪ M. Spivak, Calculus, Cambridge University Press ▪ G. Strang, Lineare Algebra, Springer Verlag ▪ H. Grauert, I. Lieb, Differential- und Integralrechnung I, Springer Verlag ▪ W. Walter, Analysis 1, Springer Verlag
Use of the Module in Other Study Programs	▪ MA1 in Bachelor Elektrotechnik 2020 ▪ MA1 in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ MA1 in Bachelor Medientechnologie 2020 ▪ MA1 in Bachelor Optometrie 2021 ▪ MA1 in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.34 MA2 - Mathematik 2

Module ID	MA2_BaTIN2020
Module Name	Mathematik 2
Type of Module	Mandatory Module
Recognized Course	MA2 - Mathematics 2
ECTS credits	10
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Heiko Knospe/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Heiko Knospe/Professor Fakultät IME ▪ Prof. Dr. Hubert Randerath/Professor Fakultät IME ▪ Prof. Dr. Beate Rhein/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul schließt an das Modul "Mathematik 1" an und setzt die Vermittlung grundlegender Konzepte und Methoden der Mathematik fort, die in der Informatik und Technik benötigt werden (K. 3). Die Abstraktion und mathematischen Formalisierung von Problemen soll erlernt und angewendet werden (K. 2). Die Studierenden lernen in der Mathematik die Grundzüge wissenschaftlichen Arbeitens kennen (K. 12).

Womit: Der Dozent/die Dozentin vermittelt Wissen und Basisfertigkeiten in der Vorlesung. In der Übung bearbeiten die Studierenden unter Anleitung Aufgaben. Die Übung wird durch Hausaufgaben und Online-Aufgaben (E-Learning) ergänzt. Zusätzlich findet ein Tutorium statt.

Wozu: Grundlegende Mathematik-Kenntnisse werden in mehreren Modulen des Studiengangs benötigt und sind anerkannter Teil der Basisausbildung. Mathematische Methoden sind essentiell für Informatiker, die Systeme zur Verarbeitung, Übertragung und Speicherung von Informationen für technische Anwendungen planen, realisieren und integrieren (HF 1). Die Analyse und Bewertung von Anforderungen, Konzepten und Systemen erfordert häufig mathematische Methoden (HF 2).

Module Contents**Lecture / Exercises**

Integral calculus

- Riemann integral, definition and properties
- Main theorem of differential and integral calculus
- Improper integrals
- Partial integration
- Substitution rule
- Partial fraction decomposition

Ordinary differential equations

- First-order differential equation with separable variables
- First-order linear differential equation with constant coefficients
- Second-order linear differential equation with constant coefficients

Functions of several variables

- Scalar functions and vector fields
- Limits and continuity
- Partial derivatives and gradient
- Jacobian matrix
- Higher partial derivatives
- Extreme values
- Error propagation
- Implicit functions
- Multidimensional integration

Vector spaces and linear mappings

- Groups, fields, finite fields
- Vector spaces and subvector spaces
- Linear mappings
- Linear independence, dimension and rank
- Determinant
- Euclidean and unitary vector spaces, scalar product, norm, Gram-Schmidt orthogonalization
- Orthogonal and unitary matrices
- Symmetric and Hermitian matrices
- Eigenvalues and eigenvectors
- Coordinates and change of basis
- Diagonalizable matrices and normal forms (optional)
- Matrix decompositions (optional)
- Homogeneous coordinates (optional)

Exercises / Lab

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Exercises / Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	300 Hours
-----------------	-----------

Contact Hours	57 Hours $\pm$ 5 SWS
----------------------	----------------------

Self-Study	243 Hours
-------------------	-----------

Recommended Prerequisites	▪ Modul MA1: Differential- und Integralrechnung von Funktionen einer Variablen sowie Grundlagen der Linearen Algebra. ▪ -
----------------------------------	--

Mandatory Prerequisites

Recommended Literature

- P. Hartmann, Mathematik für Informatiker, vieweg Verlag
- T. Westermann, Mathematik für Ingenieure, Springer Verlag
- T. Rießinger, Mathematik für Ingenieure, Springer Verlag
- W. Schäfer, G. Trippler, G. Engeln-Müllges (Hrg.), Kompaktkurs Ingenieurmathematik, Fachbuchverlag Leipzig
- L. Papula, Mathematik für Ingenieure und Naturwissenschaftler, Band 1 und 2, Vieweg+Teubner Verlag
- G. Strang, Lineare Algebra, Springer Verlag
- G. Fischer, Lineare Algebra, Springer Verlag
- D. C. Lay, Linear Algebra and its Applications, Addison Wesley Verlag
- C. Blatter, Analysis 1 und Analysis 2, Springer Verlag
- W. Walter, Analysis 1 und Analysis 2, Springer Verlag
- O. Forster, Analysis 1 und Analysis 2, Springer Verlag
- M. Knorrenschild, Mathematik für Ingenieure 2, Hanser Verlag

Use of the Module in Other Study Programs

- MA2 in Bachelor Elektrotechnik 2020
- MA2 in Bachelor Elektrotechnik und Informationstechnik 2024
- MA2 in Bachelor Medientechnologie 2020
- MA2 in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes**Last Update**

19.7.2025, 14:32:16

6.35 ML - Maschinelles Lernen

Module ID	ML_BaTIN2020
Module Name	Maschinelles Lernen
Type of Module	Elective Modules
Recognized Course	ML - Machine Learnig
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Beate Rhein/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Beate Rhein/Professor Fakultät IME

Learning Outcome(s)

Was: Die Studierenden erlernen die Methoden und Verfahren zum Machinellen Lernen mit Neuronalen Netzen. Darauf aufbauend wird die Problemlösung mittels Entwurfswerkzeugen implementiert und validiert.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. Im Praktikum erarbeiten die Studierenden in Kleingruppen Problemlösungen und verteidigen diese.

Wozu: Die Studierenden werden in dem Modul befähigt, im Problemlösung im Bereich des maschinellen Lernens mit neuronalen Netzen zu lösen und bestehende Lösungen zu bewerten.

Module Contents**Lecture / Exercises**

- fundamentals
- types of learning
- simple classifiers
- simple predictors (Equalizer alias Linear Regression)
- challenges in learning
- linear regression as the simple predictor
- linear regression as the simple classifier
- training data (handling, analysis, processing)
- gradient descent
- quality measures
- learning curve
- multi-class classifier based on binary classifiers
- multi-label-classification
- logistic regression
- simple neuronale Netze
- the artificial neuron as a simple classifier
- operation
- activation function
- bias
- training a neuron
- multi-layer-perceptron
- operation
- purposes of the layers
- backpropagation training algorithm
- tools for creating and training simple neural networks and handling training data
- handling, analysis and preparation of training data
- creating and configuring neural networks
- training neural networks
- verification of trained networks
- Deep Neural Networks (DNNs)
- basic problems
- vanishing or exploding gradients
- high training times
- overfitting
- solutions for the problems mentioned above
- appropriate initialization of the weights, non-saturating activation function, gradient clipping
- accelerated optimization procedures, reuse of pre-trained layers
- regularization to avoid overfitting
- tools for creating and training DNNs
- handling, analysis and preparation of training data
- creating and configuring neural networks
- training of neural networks
- verification and validation trained networks
- Convolutional Neural Networks (CNNs)
- idea
- architecture
- convolutional layer
- pooling layer
- convolution as a basic operator for training and detection
- architectures of CNNs for different problems
- tools for implementation and training CNNs
- Recurrent Neural Networks (RNNs)
- idea
- recurrent neurons
- training of RNNs and Deep RNNs

Long Short Term Memory
architectures of RNNs for different problems
tools for implementation and training deep CNNs

the presented neural networks
specify
describe
evaluate the pros and cons

solving problems using tools
for handling, analysis and preparation of the training data
for implementation, verification, validation and training of all neural presented networks

Lab

purposeful handling of the tools
deal with complex tasks in a small team
derive complex solutions that can be implemented using neural networks

Teaching and Learning ■ Lecture / Exercises
Methods ■ Lab

Examination Types with cf. exam regulations
Weights

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended ■ Modul MA1: Problemlösungskompetenz aus dem Bereich lineare Algebra und der Analysis. Sicheres
Prerequisites Beherrschen der entsprechenden Symbole und Formalismen.
 ■ basic skills in Java and/or C
 basic skills in analysis and linear algebra

Mandatory ■ Lab requires attendance in the amount of: 4 Testattermine
Prerequisites ■ Participation in final examination only after successful participation in Lab

Recommended ■ Géron, Aurélien, Hands-On Machine Learning with Scikit-Learn and TensorFlow, O'Reilly Medi
Literature

Use of the Module in ML in Bachelor Informatik und Systems-Engineering 2024
Other Study Programs

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.36 MLO - Machine Learning Operations

Module ID	MLO_BaTIN2020
Module Name	Machine Learning Operations
Type of Module	Elective Modules
Recognized Course	MLO - Machine Learning Operations
ECTS credits	5
Language	deutsch und englisch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
First-time Course	winter term 2025
Module Coordinator	Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Pascal Cerfontaine/Professor Fakultät IME

Learning Outcome(s)

(WAS) Die Studierenden erwerben Kenntnisse und Fertigkeiten für die Entwicklung, Bereitstellung, Verwaltung, Überwachung und Wartung von Machine-Learning-Systemen im produktiven Einsatz. Sie verstehen den Lebenszyklus von ML-Modellen und lernen, diesen systematisch zu gestalten und zu automatisieren,
 (WOMIT) indem sie in Vorlesungen/Übungen Konzepte, Werkzeuge und Best Practices der MLOps kennenlernen und diese in aufeinander aufbauenden praktischen Aufgaben selbst anwenden,
 (WOZU) um robuste, skalierbare und reproduzierbare ML-Anwendungen zu entwickeln, die auch unter realen Bedingungen zuverlässig funktionieren und in Produktionsumgebungen betrieben werden können.

Module Contents

Lecture / Exercises

- Introduction to machine learning methods for the creation of AI applications
- Introduction to the ML lifecycle: development, data preparation, model training and validation, deployment, monitoring, retraining
- Differences between classic software engineering and ML system development
- Basics of reproducible ML experiments and data and model management
- Version management for models and data sets (e.g. DVC, MLflow, Git)
- Containerization of ML models with Docker
- Model serving (e.g. REST APIs with FastAPI, TensorFlow Serving, TorchServe)
- Automated training and deployment
- Monitoring of ML systems: Model drift, data drift, performance metrics, logging, alerting
- Orchestration of ML workflows: Use of tools such as Kubeflow, Airflow or Prefect
- Scaling with cloud services (e.g. Vertex AI, AWS Sagemaker, Azure ML)
- Security, data protection and ethical aspects of ML operation

Lab

- Preparation and versioning of data for ML workflows
- Creating an AI application including model training
- Reproducible model training with DVC or MLflow
- Containerization and provision of an ML model as a web service
- Development of a CI/CD pipeline for an ML system (e.g. with GitHub Actions or GitLab CI)
- Setting up a monitoring system (Prometheus, Evidently, Grafana) to monitor models and data streams
- Implementation and evaluation of retraining cycles based on realistic scenarios
- Integration and deployment in a cloud environment

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
--------------------------------------	--

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\triangleq$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

- Modul DOPE: Grundlagen des Versionsverwaltungssystems Git werden beherrscht
- Modul DOPE: Containerisierung mit Docker ist bekannt
- Modul ML: Grundlagen des maschinellen Lernens sind bekannt
- - Basics of the version management system Git are mastered
- Containerization with Docker is known
- Basic knowledge of Python

Mandatory Prerequisites

Recommended Literature

- Yaron Haviv, Noah Gift: Implementing MLOps in the Enterprise, 2023
- How to Learn MLOps in 2024: Courses, Books, and Other Resources

Use of the Module in Other Study Programs MLO in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update 19.8.2025, 10:30:37

6.37 MT - Messtechnik

Module ID	MT_BaTIN2020
Module Name	Messtechnik
Type of Module	Elective Modules
Recognized Course	MT - Measurement Technology
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Kai Kreisköther/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Kai Kreisköther/Professor Fakultät IME

Learning Outcome(s)

- Messabweichungen kennenlernen und analysieren
 - Kennenlernen und Anwenden der Grundlagen der Stochastik
 - Kennenlernen und Analysieren statistischer Größen
 - Analoge Messgeräte kennenlernen und anwenden
 - Digitale Messgeräte kennenlernen und anwenden
 - Messverfahren und Sensorik verstehen und anwenden
-

Module Contents**Lecture / Exercises**

- General considerations
- Historical review
- The SI system
- Measurement techniques
- Known systematic measurement deviations
- Unknown systematic measurement deviations
- Reproduction of systematic measurement errors
- Random measurement errors
- Complete measurement result
- Random experiments
- Relative frequency
- The Laplace Experiment
- Conditional probability
- Independent events
- Random variable
- Distribution function and distribution density function
- Expected value, variance and standard deviation
- Central limit theorem, normal distribution and uniform Distribution
- Sample of a measurand
- Confidence interval for the expected value
- Propagation of random deviations
- Linear Regression
- Properties of electrical measuring instruments
- Moving-coil movement
- Electrodynamic movement
- Moving iron movement
- Measuring range extension for DC voltage measurement
- Measuring range extension for direct current measurement
- Limiters
- Alternating current and alternating voltage measurement
- Sampling and reconstruction
- A/D and D/A converters
- Digital Multimeter
- Logical basic gates
- Memory elements and counters
- Digital timing measurement
- Digital frequency measurement
- Digital Oscilloscopes
- Resistance determination
- Measuring bridges
- Sensors supplying voltage and current
- Resistive sensors
- Pulsed sensors

Lab

- Understanding and using digital oscilloscopes
- Analyzing of limiter circuits
- Analysis of galvanic, magnetic and capacitive couplings

Teaching and Learning ■ Lecture / Exercises**Methods** ■ Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

- Modul MA1: Gleichungssysteme lösen
- Modul MA2: Differentialrechnung / Integralrechnung
- MA1, MA2, GE1, GE2

Mandatory Prerequisites Lab requires attendance in the amount of: 3 Praktikumsversuche

Recommended Literature

- Schröder, E.: Elektrische Messtechnik
- Lerch, R.: Kaltenbacher, M.; Lindinger, F.: Übungen zur Elektrischen Messtechnik
- Felderhoff, R.: Elektrische und elektronische Messtechnik
- Weichert, N.: Messtechnik und Messdatenerfassung

Use of the Module in Other Study Programs

- MT in Bachelor Elektrotechnik 2020
- MT in Bachelor Elektrotechnik und Informationstechnik 2024
- MT in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.38 NDQ - Nachhaltigkeit durch Qualität

Module ID	NDQ_BaTIN2020
Module Name	Nachhaltigkeit durch Qualität
Type of Module	Elective Modules
Recognized Course	NDQ - Nachhaltigkeit durch Qualität
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Ansgar Beuten/Lehrbeauftragter
Lecturer(s)	Ansgar Beuten/Lehrbeauftragter

Learning Outcome(s)

Die Studierenden kennen die verschiedenen Formen von Nachhaltigkeit (ökologisch, ökonomisch, sozial), können diese voneinander abgrenzen und im Kontext erläutern.

Die Studierenden können für die verschiedenen Formen von Nachhaltigkeit Ziele definieren, Kennzahlen ableiten und Ansätze im Hinblick auf Nachhaltigkeit bewerten.

Die Studierenden können Nachhaltigkeit zielgruppenspezifisch argumentieren und fachlich vertreten.

Die Studierenden sind in der Lage das Mindset eines Gegenübers in Themen der Nachhaltigkeit positiv zu verändern.

Die Studierenden können verschiedene Arten von Qualität benennen, erkennen, erklären und differenzieren.

Die Studierenden können verschiedene Methoden des Qualitätsmanagements erkennen, erklären, differenzieren und anwenden.

Die Studierenden kennen verschiedene Werkzeuge des Qualitätsmanagements und können diese erklären und anwenden.

Die Studierenden sind in der Lage, Verbindung zwischen Nachhaltigkeit und Qualität herzustellen, Abhängigkeiten zu erkennen und zu analysieren. Die Studierenden können durch Anwenden der erlernten Methoden und Werkzeuge Nachhaltigkeit erzeugen und optimieren.

Module Contents

Lecture

Seminar-style Teaching

Teaching and Learning Methods	▪ Lecture ▪ Seminar-style Teaching
--------------------------------------	---

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	▪ Modul MA1: erforderlich für das Verständnis statistischer Methoden ▪ Modul MA2: erforderlich für das Verständnis statistischer Methoden
----------------------------------	--

Mandatory Prerequisites	▪ Seminar-style Teaching requires attendance in the amount of: An mindesten acht Terminen des Seminars müssen sich die Studierenden anwesend sein und sich beteiligen. ▪ Participation in final examination only after successful participation in Seminar-style Teaching
--------------------------------	--

**Recommended
Literature**

- | | |
|--|---|
| Use of the Module in
Other Study Programs | <ul style="list-style-type: none">▪ NDQ in Bachelor Elektrotechnik 2020▪ NDQ in Bachelor Elektrotechnik und Informationstechnik 2024▪ NDQ in Bachelor Medientechnologie 2020▪ NDQ in Bachelor Medientechnologie 2024▪ NDQ in Bachelor Optometrie 2021▪ NDQ in Bachelor Informatik und Systems-Engineering 2024 |
|--|---|

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.39 NP - Netze und Protokolle

Module ID	NP_BaTIN2020
Module Name	Netze und Protokolle
Type of Module	Mandatory Module
Recognized Course	NP - F07_Networks and Protocols
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Andreas Grebe/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Andreas Grebe/Professor Fakultät IME

Learning Outcome(s)

Was:

Das Modul vermittelt Wissen zu Kommunikationsprotokollen und deren Rolle und Mechanismen, Wissen zur Architektur und zum Aufbau von Computernetzen sowie ein tieferes Verständnis der zugrundeliegenden Konzepte und Techniken.

Das Modul vermittelt Kompetenzen zur Planung, Implementierung, Evaluierung und zum Betrieb von Computernetzen. Folgende Kenntnisse und Kompetenzen werden im Detail vermittelt: Grundlegende Konzepte und Technologien von Rechnernetzen benennen, strukturieren, einordnen (K.2, K.4, K.8), Strukturieren der Aufgaben in der technischen Kommunikation, zuordnen auf einschlägige Standardisierungen und übertragen auf Netzdesign und Client-/Server-Anwendungen (K.1, K.2, K.3, K.8), Protokolle (Anwendungen, Transport, Netzwerk, Ethernet, Übertragungstechnik) zuordnen und benennen, Protokoll-Mechanismen erläutern, Aufgaben und technische Parameter darlegen und strukturieren (K.1, K.2, K.8), Netze und Systeme unter Einsatz geeigneter Tools analysieren und grafisch darstellen (K.4, K.7, K.8, K.9), Systeme in Netze einbinden, Systemkonfiguration planen (K.4, K.5, K.6, K.7, K.10), Netze planen und einrichten (K.4, K.5, K.6, K.7, K.10), Leistungsfähigkeit von Rechnernetzen abschätzen und analysieren (K.2, K.3, K.7, K.8), Information aus englischen Originalquellen und Standards ableiten (K.2, K.8, K.3, K.4, K.15).

Womit:

Kenntnisse und Basisfertigkeiten werden in Vorlesung und Übung vermittelt. Darauf aufbauend werden im Praktikum Kompetenzen und Fertigkeiten ausgebaut und inhaltliche Themen vertieft. Im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K.8, K.16).

Wozu:

Computernetze sind heute die Grundlage für alle technischen Kommunikationssysteme, von der Telekommunikation über Unternehmensnetze bis hin zu Automatisierung und grundlegender Digitalisierung. Sie bilden die Kommunikationsplattform für verteilte Systeme. Entsprechende Kompetenzen und Wissen über die zugehörigen Grundlagen sind essentiell für die Erstellung (HF1), Bewertung (HF2) und Betrieb (HF3) moderner verteilter Systeme und Services. Die Verteidigung der eigenen Lösungen in der Übung und im Praktikum fördert die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents**Lecture / Exercises**

Fundamentals of architectures and topologies of computer networks, metrics, LAN, MAN WAN, communication and layer models according to ISO/OSI, IETF TCP/IP, IEEE, bit transmission and data interconnections, Ethernet technology, IP addressing and subnetting, IP routing and routing protocols, frame switching and virtual LAN, transport protocols, application protocols and communication patterns

Analyze networks and systems using suitable tools and present them graphically. Integrate systems into networks. Planning and setting up (sub)networks. Estimate and analyse the performance of computer networks. Obtaining information from original English sources.

Excerpt of the contents:

ISO/OSI reference models, TCP/IP model, IEEE model, switch, router, host, transmission media, Ethernet, 100BASE-Tx, 1000BASE-T, ARP, addressing IPv4, IPv6, DHCP, ICMP, switched LAN, virtual LAN (VLAN), static routing, RIP, OSPF, transport protocols UDP, TCP, QUIC, applications DNS, HTTP, FTP, TFTP, Telnet, SSH

Lab

Knowing, structuring, classifying, structuring basic concepts and technologies of computer networks, structuring tasks in technical communication, assigning to relevant standardizations and transferred to network design and client/server applications, assigning and naming protocols (applications, transport, network, Ethernet, transmission technology), explaining protocol mechanisms, setting out and structuring tasks and technical parameters. Master network analysis techniques and tools, know network design steps and methods for network planning.

Analyze networks and systems using suitable tools and present them graphically.

Integrate systems into networks.

Planning and setting up (sub)networks.

Estimate and analyse the performance of computer networks.

Systematic troubleshooting and correction.

Evaluate information from original sources and apply it to networks.

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	▪ Modul PI1: Sicherer Umgang mit konsolenbasierter Systemsteuerung und einer Programmiersprache inklusive Boole'scher Operationen. ▪ Modul PI2: Sicherer Umgang mit konsolenbasierter Systemsteuerung und einer Programmiersprache inklusive Boole'scher Operationen. ▪ Modul DR: Kenntnisse über Aufbau und Funktionalität eines Digitalrechners/Computers. ▪ Boole Operations, AND, OR, XOR Binary numbers Basic Computer Architecture Basic Knowledge of Operating Systems (Unix/Linux preferred) Basic programming skills
----------------------------------	--

Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 4 Praktikumsversuche ▪ Participation in final examination only after successful participation in Lab
--------------------------------	---

**Recommended
Literature**

- J. Kurose, K. Ross: Computernetzwerke - Der Top-Down-Ansatz, Pearson Studium, 6. Auflage, 2014
- A. Tanenbaum: Computernetzwerke, Pearson Studium, 5. Auflage 2012
- Douglas Comer: Computer Networks and Internets, Pearson Education Limited, 6 edition, 2015
- Internet-Standardisierung: IETF Standards (RFCs), www.ietf.org
- LAN-Standards: IEEE, ieeexplore.ieee.org (freier Zugang über TH Köln)
- Telekommunikationsstandards: ITU-T Standards, www.itu.int
- Web-Standardisierung: W3C Standards, www.w3c.org

**Use of the Module in
Other Study Programs**

- NP in Bachelor Elektrotechnik 2020
- NP in Bachelor Elektrotechnik und Informationstechnik 2024
- NP in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes

Last Update 19.7.2025, 14:32:16

6.40 NSA - Netzsicherheit und Automation

Module ID	NSA_BaTIN2020
Module Name	Netzsicherheit und Automation
Type of Module	Elective Modules
Recognized Course	NSA - Network Security and Automation
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Andreas Grebe/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Andreas Grebe/Professor Fakultät IME

Learning Outcome(s)

Das Modul vertieft Wissen und Kompetenzen zu IP-Netzen und Kommunikationsprotokollen. Das Modul vermittelt Kompetenzen zur Planung, Implementierung, Evaluierung und zum Betrieb von größeren, standortübergreifenden Computernetzen inklusive der dazugehörigen Netzsicherheitstechniken und verteilter Netzmanagementtechniken. Zu den Kenntnissen und Kompetenzen gehören: Grundlegende Konzepte und Technologien von skalierenden Rechnernetzen benennen, strukturieren, einordnen (K.2, K.4, K.8), Skalierende Netze unter Einsatz geeigneter Tools analysieren und grafisch darstellen (K.4, K.7, K.8, K.9), planen und einrichten (K.4, K.5, K.6, K.7, K.10), Leistungsfähigkeit von Rechnernetzen abschätzen und analysieren (K.2, K.3, K.7, K.8), Sicherheitsrisiken und Abwehrtechniken erläutern, implementieren und bewerten (K.1, K.2, K.3, K.7, K.8), Netzmanagementaufgaben und -techniken erläutern, implementieren und bewerten (K.1, K.2, K.3, K.7, K.8), Information aus englischen Originalquellen und Standards ableiten (K.2, K.8, K.3, K.4, K.15).

Womit:

Kenntnisse und Basisfertigkeiten werden in Vorlesung und Übung vermittelt. Darauf aufbauend werden im Praktikum Kompetenzen und Fertigkeiten ausgebaut und inhaltliche Themen vertieft. Im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K.8, K.16).

Wozu:

Computernetze sind heute die Grundlage für alle technischen Kommunikationssysteme, von der Telekommunikation über Unternehmensnetze bis hin zu Automatisierung und grundlegender Digitalisierung. Das auf dem Modul NP aufbauende Modul IN fokussiert auf Kompetenzen zur Planung, Implementierung (HF1), Betrieb (HF3) und Evaluierung (HF2) von größeren, standortübergreifenden Unternehmensnetzen. Insbesondere durch die Verbindung zum Internet und die standortübergreifenden Aspekte werden Netzsicherheit und Netzmanagement als weitere zusätzliche Schwerpunkt aufgenommen. Die Verteidigung der eigenen Lösungen in der Übung und im Praktikum fördert die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents**Lecture / Exercises**

Fundamentals for the construction of hierarchically structured networks, corporate networks with redundancy techniques, wireless LAN (WLAN), cross-location communication, WAN techniques. Introduction to network security with details on attacks, security targets, cryptographic procedures, encryption, packet filters, secure infrastructures, virtual private networks. Introduction to distributed network management and service quality techniques. Techniques for network virtualization, software-defined networking and network automation.

Students acquire the skills to analyse medium-sized, cross-location corporate networks using suitable tools, to select suitable architectures and to plan and implement corresponding networks. They name and identify hazard situations for corporate networks. Suitable security mechanisms are to be selected, designed and implemented. Tasks and methods of software-controlled networks including virtualization are known and mechanisms for network automation are planned and implemented.

Excerpt of the contents:

Hierarchical networks, redundancy, STP, EtherChannel, FHRP, Single-area and Multiarea OSPF, OSPF security technologies, WLAN, WAN connection, PPP, xDSL

Network security with security goals, cryptographic methods, algorithms, packet filters, ACL, NAT, FireWall, DMZ, VPN, IPsec
SNMP, Syslog, QoS - Quality of Service

Software Defined Networking (SDN), SDN Controller, Cloud, Virtualization, Ansible, JSON, YAML, REST API

Lab

Identify, structure and classify concepts and technologies for medium-sized, cross-location corporate networks. Master network analysis techniques and tools, know network design steps and methods for network planning. Identify security-relevant network gaps and know suitable measures for network security and their implementation. Know the tasks of network automation and virtualisation and master their implementation for suitable network areas.

Planning, implementation and analysis of VLAN architectures, WLAN networks, cross-site VPN and packet filter firewall.

Implementation and analysis of network management with SNMP and Syslog.

Implementation and analysis of network automation on network elements (e.g. router, switch, host, SDN controller) via REST API with Python scripting or Ansible YAML scripting.

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	▪ Modul NP: Fundamentale Kenntnisse über IP-basierte Netze und Kompetenzen zu Planug, Implementierung und Betrieb von IP-Netzen werden vorausgesetzt. Im Praktikum werden Kenntnisse zu Protokollanalyse (Wireshark) und Konfiguration / Programmierung von Router und Switch (Cisco IOS-Befehlssätze) erwartet. Die Voraussetzungen können u.a. durch das Testat für die ULP NP oder die nachgewiesenen CCNA Module ITN und RSE nachgewiesen werden. ▪ Knowledge and Competences of Module Networks and Protocols (NP) alternate requirements: knowledge and competences of IP networking courses or CCNA (ITN and RSE) Networking Fundamentals TCP/IP Protocol Family ISO/OSI Model and Protocols IPv4/IPv6 Addressing, Subnetting, and Routing Switching Techniques TCP/UDP Transport Techniques Application Protocols Network Implementation Competences (Client, Server, Switch, Router)
----------------------------------	--

Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 4 Termine ▪ Participation in final examination only after successful participation in Lab
--------------------------------	--

Recommended Literature

- J. Kurose, K. Ross: Computernetzwerke - Der Top-Down-Ansatz, Pearson Studium, 6. Auflage, 2014
- A. Tanenbaum: Computernetzwerke, Pearson Studium, 5. Auflage 2012
- G. Schäfer: Netzsicherheit: - Grundlagen & Protokolle - Mobile & drahtlose Kommunikation - Schutz von Kommunikationsinfrastrukturen, dpunkt.verlag, 2. Auflage 2014
- W. Stallings: Foundations of Modern Networking, Pearson Education, 2016
- J. Doherty: SDN and NFV Simplified, Pearson Education, 2016
- J. Edelman: Network Programmability and Automation, O'Reilly 2018
- Internet-Standardisierung: IETF Standards (RFCs), www.ietf.org
- LAN-Standards: IEEE, ieeexplore.ieee.org (freier Zugang über TH Köln)
- Telekommunikationsstandards: ITU-T Standards, www.itu.int
- Web-Standardisierung: W3C Standards, www.w3c.org

Use of the Module in Other Study Programs

- NSA in Bachelor Elektrotechnik 2020
- NSA in Bachelor Elektrotechnik und Informationstechnik 2024
- NSA in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes**Last Update**

19.7.2025, 14:32:16

6.41 PHTB - Philosophische Handlungstheorie Bachelor

Module ID	PHTB_BaTIN2020
Module Name	Philosophische Handlungstheorie Bachelor
Type of Module	Elective Modules
Recognized Course	PHTB - Philosophical theory of action
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Gregor Büchel/Professor Fakultät IME im Ruhestand
Lecturer(s)	Prof. Dr. Gregor Büchel/Professor Fakultät IME im Ruhestand
Learning Outcome(s)	
▪ (WAS) Studierende wenden philosophischen Theorien auf Probleme des Handelns in der heutigen Gesellschaft an, ▪ (WOMIT) indem Sie zentrale philosophische Texte studieren, seminaristisch aufarbeiten und präsentieren, ▪ (WOZU) um ihr späteres gesellschaftliches und berufliches Handeln auf philosophisch und ethisch durchdachten Grundlagen aufbauen zu können.	
Module Contents	
Lecture	
The background of philosophical theories of action is "illuminated" in the lecture	
Seminar	
The seminar will focus on the following five texts by Immanuel Kant:	
1. "Answering the Question: What is Enlightenment?" 1 "Ideas for a General History with a Cosmopolitan Intention", 1 "Groundwork for the Metaphysics of Morals", 2. the antinomy of freedom and natural necessity in the "Critique of Pure Reason", 1 "On Perpetual Peace".	
Aspects of the philosophical theory of action given in these texts are to be applied to problems of action in today's society.	
Teaching and Learning Methods	▪ Lecture ▪ Seminar
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	34 Hours $\pm$ 3 SWS
Self-Study	116 Hours
Recommended Prerequisites	
Mandatory Prerequisites	▪ Seminar requires attendance in the amount of: 6 Termine ▪ Participation in final examination only after successful participation in Seminar

Recommended Literature

- Immanuel Kant: „Beantwortung der Frage: Was ist Aufklärung? Und andere kleine Schriften“, Berlin (Sammlung Hoffenberg), 2016, ISBN: 978-3-8430-9208-1
- Immanuel Kant: „Schriften zur Geschichtsphilosophie“, Stuttgart (Reclam), 2013, ISBN: 978-3-15-009694-9
- Immanuel Kant: „Grundlegung zur Metaphysik der Sitten“, Stuttgart (Reclam), 2016, ISBN: 978-3-15-004507-7
- Immanuel Kant: „Kritik der reinen Vernunft“, Stuttgart (Reclam), 1966, ISBN: 978-3-15-006461-0
- Immanuel Kant: „Zum ewigen Frieden“, Stuttgart (Reclam), 2012, ISBN: 978-3-15-001501-8

Use of the Module in Other Study Programs

- PHTB in Bachelor Elektrotechnik 2020
- PHTB in Bachelor Elektrotechnik und Informationstechnik 2024
- PHTB in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes**Last Update**

19.7.2025, 14:32:16

6.42 PI1 - Praktische Informatik 1

Module ID	PI1_BaTIN2020
Module Name	Praktische Informatik 1
Type of Module	Mandatory Module
Recognized Course	PI1 - Practical Informatics 1
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Markus Cremer/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Markus Cremer/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt grundlegende Kompetenzen zur Nutzung von Programmiersprachen und entsprechender abstrakterer Darstellungsformen bei der algorithmischen und objektorientierten Lösung von Anwendungsproblemen. Die Studierenden sollen in die Lage versetzt werden, einschlägige Begrifflichkeiten und Techniken im praktischen Umfeld sicher anzuwenden: Aufbauend auf den in der Vorlesung vermittelten Kenntnissen (K3) analysieren die Studierenden Problemstellungen (K2), entwerfen Lösungswege dazu (K5), implementieren sie mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum bearbeiten die Studierenden in Kleingruppen Programmieraufgaben und verteidigen ihre Lösungen (K8, K13, K16).

Wozu: Kompetenzen in der Anwendung von Programmiersprachen sind essentiell für Informatiker/-innen, insbesondere in Hinblick auf die Realisierung informationstechnischer Systeme (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden zudem Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents**Lecture / Exercises****Lab**

Teaching and Learning Methods	■ Lecture / Exercises ■ Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\triangleq$ 4 SWS
----------------------	-----------------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	none
----------------------------------	------

Mandatory Prerequisites	■ Lab requires attendance in the amount of: 3 Termine ■ Participation in final examination only after successful participation in Lab
--------------------------------	--

Recommended Literature

- Ernst, H., Schmidt, J., & Beneken, G. (2023). Grundkurs Informatik: Grundlagen und Konzepte für die erfolgreiche IT-Praxis – Eine umfassende Einführung. Springer Vieweg. <https://doi.org/10.1007/978-3-658-41779-6>
- Gumm, H.-P., & Sommer, M. (2016–2019). Einführung in die Informatik (3 Bände). De Gruyter / Oldenbourg.
- Herold, H., Lurz, B., Lurz, M., & Wohlrab, J. (2023). Grundlagen der Informatik (4. Aufl.). Pearson Studium.
- C. Vogt (2004). Informatik – Eine Einführung in Theorie und Praxis, Spektrum.
- Mössenböck, H. (2014). Sprechen Sie Java (5. Aufl.). dpunkt.verlag.
- Ratz, D., Schulmeister-Zimolong, D., Seese, D., & Wiesenberger, J. (2022). Grundkurs Programmieren in Java (8. Aufl.). Hanser.
- Lo Iacono, L., Wiefling, S., & Schneider, M. (2020). Programmieren trainieren (2. Aufl.). Hanser.
- Schiedermeier, R., & Köhler, K. (2012). Das Java-Praktikum (2. Aufl.). dpunkt.verlag.
- Ullenboom, C. (2023). Java ist auch eine Insel (17. Aufl.). Rheinwerk Verlag. Openbook.

Use of the Module in Other Study Programs

PI1 in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes**Last Update**

19.7.2025, 14:32:16

6.43 PI2 - Praktische Informatik 2

Module ID	PI2_BaTIN2020
Module Name	Praktische Informatik 2
Type of Module	Mandatory Module
Recognized Course	PI2 - Practical Informatics 2
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	2
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Chunrong Yuan/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Chunrong Yuan/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt grundlegende Kompetenzen zur Nutzung von Programmiersprachen und entsprechender abstrakterer Darstellungsformen bei der algorithmischen und objektorientierten Lösung von Anwendungsproblemen. Die Studierenden sollen in die Lage versetzt werden, einschlägige Begrifflichkeiten und Techniken im praktischen Umfeld sicher anzuwenden: Aufbauend auf den in der Vorlesung vermittelten Kenntnissen (K3) analysieren die Studierenden Problemstellungen (K2), entwerfen dazu Lösungswege (K5), implementieren sie mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum bearbeiten die Studierenden in Kleingruppen Programmieraufgaben und verteidigen ihre Lösungen (K8, K13, K16)

Wozu: Kompetenzen in der Anwendung von Programmiersprachen sind essentiell für Informatikerinnen und Informatiker, insbesondere im Hinblick auf die Realisierung informationstechnischer Systeme (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden zudem Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2). Die Durchführung im Team mit dem Dozenten als „Auftraggeber“ stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents**Lecture / Exercises**

Object oriented Programming: Class structures
Object oriented Programming: Generics
Exception handling
Input and output: Streams and files
Input and output: Graphic user interfaces (GUIs)
Dynamic data structures: simple structures
Dynamic data structures: Graphs
Formal specification of syntactic structures
Object oriented programming

Lab

Object oriented implementation of dynamic data structures
Object oriented implementation of GUI components

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights

cf. exam regulations

Workload 150 Hours

Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	■ Modul PI1: Grundkenntnisse in PI1 Programmiererfahrung mit Entwicklungsumgebungen wie Eclipse ■ Basic knowledge of PI1 Capability of programming using development environments such as Eclipse
Mandatory Prerequisites	
Recommended Literature	■ Vogt: Informatik – Eine Einführung in Theorie und Praxis, Spektrum, 2004
Use of the Module in Other Study Programs	PI2 in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.44 PP - Programmierpraktikum

Module ID	PP_BaTIN2020
Module Name	Programmierpraktikum
Type of Module	Mandatory Module
Recognized Course	PP - Programming Practice
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	1
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Chunrong Yuan/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Chunrong Yuan/Professor Fakultät IME

Learning Outcome(s)

Was: Die Studierenden erwerben und vertiefen ihre Kompetenzen in der Analyse typischer Aufgabenstellungen (K.4) dem Entwurf (K.5), der Erstellung (K.6) und dem Test bzw. Prüfen (K.7, K.9) von Softwaresystemen und nebenbei in der Abstraktion und Formalisierung (K.2) fachlicher Probleme.

Womit: Die Studierenden erhalten ausgewählte Programmieraufgaben steigender Komplexität, dies sie zunächst analysieren, dann ein passendes Programm mit vorgegebenen Methoden und modernen Entwicklungsumgebungen entwerfen und programmieren, es mit vorgegebenen und selbst festzulegenden Testfällen prüfen.

Wozu: Damit wird eine Basis gelegt, auf der sie dann später, in den Veranstaltungen höherer Semester, aber auch im Berufsleben, eigenständig IT-Aufgaben analysieren, passende Systeme entwerfen, implementieren und prüfen können.

Module Contents**Lecture**

Presentation of selected standards for the design and development of programs

- * Algorithm construction with structogram / program flow plan
- * Automata
- * Structured Analysis with data flow diagrams and data dictionary

Use of a programming development environment for programming and debugging

Recursion as a means to implement series used in mathematics

Lab

Algorithm

Description formats

- description using natural language
- graphical representations (structogram, program flow plan)
- data flow diagram and data directory

Algorithms for solving some standard problems

- iteration and repetition
- recursion
- regular automaton

Implementation of algorithms using control structures (in Java and C) using

- conditionals (if, switch)
- loops (iteration, repetition)

Design and use of subroutines (in Java)

especially: implementation of predefined interfaces

Structured data types

Arrays

Java objects and classes (in Java: Public classes without methods)

Program development environment

- create projects
- debug
- test

Design algorithms solving given problems

From algorithm description to implementation

Work with program development environment

Programming as solution for scenario-based problems

Teaching and Learning ■ Lecture

Methods ■ Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 34 Hours $\triangleq$ 3 SWS

Self-Study 116 Hours

Recommended Prerequisites parallel participation of the course "Practical Informatics 1"

Mandatory Prerequisites Lab requires attendance in the amount of: Praktikumstermine

Recommended Literature	▪ Online-Dokumentation der Java-Pakete (java.sun.com) ▪ Online-Dokumentation der verwendeten Softwareentwicklungsumgebung (Eclipse) ▪ Mössenböck, Sprechen Sie Java?, dpunkt 2011 ▪ Schiedermeier/Köhler, Das Java-Praktikum, dpunkt 2011 ▪ Vogt, Informatik, Spektrum Verlag 2004
-------------------------------	--

Use of the Module in Other Study Programs	PP in Bachelor Informatik und Systems-Engineering 2024
--	--

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.45 PPRA - Parallelprogrammierung und Rechnerarchitekturen

Module ID	PPRA_BaTIN2020
Module Name	Parallelprogrammierung und Rechnerarchitekturen
Type of Module	Elective Modules
Recognized Course	PPRA - Parallel Programming and Computerarchitektur
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Markus Cremer/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Georg Hartung/Professor Fakultät IME im Ruhestand

Learning Outcome(s)

Was: Das Modul vermittelt die Kompetenzen zur Verwendung von lose und eng gekoppelten Parallelrechnersystemen für die Erarbeitung von Problemlösungen z.B. aus dem Bereich der Simulation oder der Künstlichen Intelligenz. Die Problemlösungen werden unter Verwendung von state-of-the-art Entwicklungsumgebungen (z.B. MPI und CUDA) auf Basis gängiger Programmierparadigmen und Design-Pattern erstellt. Die Studierenden werden in die Lage versetzt, fachspezifische Begriffe, Tools und Techniken im praktischen Umfeld sicher anzuwenden. Aufbauend auf den in der Vorlesung vermittelten Kenntnissen werden komplexere Problemstellungen analysiert, auf Teilsysteme heruntergebrochen und modelliert. Darauf aufbauend wird die Problemlösung mittels Entwurfswerkzeugen implementiert und am Zielsystem in Betrieb genommen.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. Im Praktikum erarbeiten die Studierenden in Kleingruppen Problemlösungen und verteidigen diese.

Wozu: Kompetenzen in der Verwendung von parallelen Rechnersystemen sind essentiell für technische Informatiker, die im HF 1 arbeiten wollen. Durch die Entwicklung von Problemlösungen erwerben die Studierenden zudem Erfahrungen, die essentiell für das HF 2 sind. Eine projektorientierte Durchführung der Praktika in kleinen Teams mit dem Dozenten als "Auftraggeber" initiiert die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents**Lecture / Exercises**

Overview parallel computer architectures and classification

Design methods of parallel applications (PCAM), Decomposition patterns (e.g. Functional, Data domain)

pattern for parallel programming, e.g. master worker, single program multiple data, multiple program multiple data and mapping to decomposition patterns

- Performance evaluation of parallel programs, e.g. with Amdahl's law
- Modelling parallel programs with Coloured Petri Nets
- Programming distributed computer systems using MPI
- Programming of Multicore/parallel computers using OpenMP and Rust

Programming of Graphic Processing Units (GPU) using CUDA

- Design of domain specific architectures
- Digital Hardware Accelerators (e.g. in signal processing)
- Tensor Processing Unit (Artificial neural network)

Project

Application of design methods for parallel applications to a self-chosen application

Modelling a parallel design of the application

implementation of parallel program for the application

Measurement of speed-up by parallelization of the application

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Project
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	basic skills in procedural programming basic skills in programming multiple tasks structure and mode of operation of a simple computer basics in digital systems (Automata, Hardware Description Language)
----------------------------------	---

Mandatory Prerequisites	Project requires attendance in the amount of: 2 Labortermine + 1 Präsentationstermin
--------------------------------	--

Recommended Literature	▪ Barlas, Gerassimos: Multicore and GPU Programming: An Integrated Approach ▪ Tanenbaum, Goodman: Computerarchitektur, Pearson Studium (Prentice Hall) ▪ Pacheco: Parallel Programming with MPI ▪ Eijkhout: Introduction to High Performance Computing
-------------------------------	---

Use of the Module in Other Study Programs	PPRA in Bachelor Informatik und Systems-Engineering 2024
--	--

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.46 PRA - Praxisphase

Module ID	PRA_BaTIN2020
Module Name	Praxisphase
Type of Module	Mandatory Module
Recognized Course	PRA - Practical phase
ECTS credits	23
Language	deutsch und englisch
Duration of Module	2 Semester
Recommended Semester	6,7
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Bachelor Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	verschiedene Dozenten*innen / diverse lecturers

Learning Outcome(s)

Was: Die Studierenden sich in eine unbekannte (außeruniversitäre) Organisation einfügen und dort adäquat agieren.

Womit: Die Studierenden solle ein eigenes Projekt in Abstimmung mit Kollegen und Vorgesetzten in der Firma organisieren und durchführen.

Dazu müssen sie selbständig wissenschaftliche Methoden und moderne Technologien in der Praxis anwenden und sich selbständig in ein neues Aufgabengebiet einarbeiten. Typischerweise wird ein System oder eine Systemkomponente selbständig und im Team konzeptionieren und entwickeln.

Wozu: Die Studierenden solle einen ersten Einblick in ihr zukünftiges Berufsumfeld und die damit verbundenen fachlichen aber auch sozialen Gegenbenheiten bekommen.

Module Contents

External Internship

markdown

- Gain an insight into the work processes and organizational structure of companies.
- Assess tasks and activities in the professional world after graduation.

markdown

- Organize and carry out own project in coordination with colleagues and superiors in the company.
- Independently apply and use scientific methods and modern technologies in practice.
- Independently familiarize yourself with a new area of responsibility.
- Fitting into an unfamiliar (non-university) organization and acting appropriately there.
- Design and/or develop and/or test a system or system component independently and as part of a team.
- Present work results orally and in writing.

Teaching and Learning Methods External Internship

Examination Types with Weights cf. exam regulations

Workload 690 Hours

Contact Hours 0 Hours $\pm$ 0 SWS

Self-Study 690 Hours

Recommended Prerequisites

- Mandatory Prerequisites**
- External Internship requires attendance in the amount of: Mit Praktikumsgeber zu vereinbaren
 - Participation in final examination only after successful participation in External Internship

Recommended Literature

Use of the Module in Other Study Programs	PRA in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	The spectrum of specific tasks covers the entire field of computer science and related engineering sciences. For this reason, all of the skills taught in the preceding course of study are potentially necessary and can be individually limited solely by the specific task that is worked on during the practical phase.
Last Update	19.7.2025, 14:32:16

6.47 PUK - Präsentation und Kommunikation

Module ID	PUK_BaTIN2020
Module Name	Präsentation und Kommunikation
Type of Module	Mandatory Module
Recognized Course	PUK - Presentation and Communication
ECTS credits	3
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	5
Frequency of Course	every winter term
Module Coordinator	Studiengangsleiter(in) Bachelor Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	NN/Lehrbeauftragter
Learning Outcome(s)	
Die Studierenden können moderne Präsentations- und Argumentationstechniken anwenden, indem Sie - Typen von Kommunikationsprozessen bestimmen, - Konzepte zur Vermittlung von Kommunikationsinhalten einordnen, - Grundzüge der Rhetorik sowie - moderne Präsentations- und Moderationstechniken anwenden, um in begleitenden und späteren Modulen, sowie im späteren Berufsleben technische und wissenschaftliche Themen zielgruppengerecht präsentieren, technische-wissenschaftliche Texte verfassen und sachgerecht argumentieren können.	
Module Contents	
Seminar	
- communication process types - concepts for communicating content - rhetoric basics - modern presentation techniques - chairing groups	
Teaching and Learning Methods	Seminar
Examination Types with Weights	cf. exam regulations
Workload	90 Hours
Contact Hours	12 Hours $\pm$ 1 SWS
Self-Study	78 Hours
Recommended Prerequisites	■ Modul SYP: SYP ist keine direkte Voraussetzung, aber die in PUK vermittelten Kompetenzen werden im Rahmen von SYP bei Projektpräsentation geprüft. ■ none
Mandatory Prerequisites	Seminar requires attendance in the amount of: 3 Termine
Recommended Literature	
Use of the Module in Other Study Programs	PUK in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	

Last Update

19.7.2025, 14:32:16

6.48 QKC - Quellen- und Kanalcodierung

Module ID	QKC_BaTIN2020
Module Name	Quellen- und Kanalcodierung
Type of Module	Elective Modules
Recognized Course	QKC - Source and Channel Coding
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Uwe Dettmar/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Uwe Dettmar/Professor Fakultät IME

Learning Outcome(s)

Was? Die in gespeicherten oder aktuell entstehenden Daten befindliche Information extrahieren und gegen Fehler bei Übertragung über einen gestörten Kanal und Abhören durch Dritte schützen und zugehörige Verfahren analysieren und bewerten.
Womit? Durch Anwendung von Verfahren und Algorithmen der Quellen- und Kanalcodierung und der Kryptographie.
Wozu? Zur Gewährleistung einer vertraulichen, effizienten und sicheren Speicherung und Übertragung von Daten mit Hilfe von Nachrichtentechnischen Systemen.

Module Contents**Lecture / Exercises**

The underlying concept of this module is a combination from lecture and tutorial. After a lecture block of approximately 20 minutes) the subjects taught are actively trained using Matlab/Octave and Python programs.

Syllabus:

- basics on source and channel coding and cryptology
- system theoretical description of a transmission system
- basics math for source and channel coding and cryptography
- basics on information theory related to source and channel coding
- examples for source and block codes
- aspects of IT security
- public and private key cryptography
- cryptological protocols

These subjects are presented during the lecture. Students shall deepen their knowledge by self-study of literature and internet resources and discuss their results in small learning groups as a teamwork.

By the help of small exercises and programs during the presence time, students are able to actively train their knowledge. More extensive problems are solved and discussed in the second part of the course to activate the student's capabilities to solve relevant problems.

Students further learn

- to analyze communication systems and to estimate their performance
- to compare and rate algorithms and methods
- to apply their knowledge to technical problems

Lab

Students accomplish lab problems from source and channel coding using Jupyter Notebooks.

Simulations are provide as complete or partly complete Python programs. Students collect and present the data by graphical representation.

Matlab in combination with the Communications Toolbox ist used for more extensive simulation tasks.

- students train their capabilities to solve technical problems by programming
- students analyze and simulate digital communication systems and rate their performance
- they train their self-management and their problem focused thinking and acting
- students train solving problems in a team and their communication skills

Teaching and Learning	▪ Lecture / Exercises
Methods	▪ Lab

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	▪ Modul MA1: Kenntnisse in Lineare Algebra, Funktionentheorie, Algebra Fähigkeit, diese Kenntnisse in praktischen Problemen anzuwenden. ▪ Modul MA2: Kenntnisse: Reihen und Folgen, Fehlerrechnung Fähigkeit, diese Kenntnisse in praktischen Problemen anzuwenden. ▪ Modul PI1: Algorithmen zur Lösung vorgegebener Probleme formulieren Beherrschung grundlegender Programmierfähigkeiten ▪ Students should have basic knowldege in algebra, linear algebra, and stochastics as well as the capabilities to write small programs in a standard programming language. In the framework of this module Python and Matlab/Octave is used for simulations.
----------------------------------	---

Mandatory	■ Lab requires attendance in the amount of: 4 Praktikumstermine
Prerequisites	■ Participation in final examination only after successful participation in Lab
Recommended Literature	■ BOSSERT, M. : Einführung in die Nachrichtentechnik. Oldenbourg Verlag, 2012. ■ BOSSERT, M. : Kanalcodierung. Oldenbourg Wissenschaftsverlag GmbH, 2013. ■ NEUBAUER, A. : Informationstheorie und Quellencodierung. Wilburgstetten : Schlembach, 2006. ■ PROAKIS, J. G. ; SALEHI, M. : Digital Communications. 5. McGraw-Hill, 2008. ■ SAYOOD, K. : Introduction to data compression. third. Elsevier Morgan Kaufmann, 2000. ■ MEYER, M. : Kommunikationstechnik. 4. Vieweg und Teubner, 2019. ■ SKLAR, B. : Digital Communications. Prentice Hall PTR, 2001
Use of the Module in Other Study Programs	■ QKC in Bachelor Elektrotechnik 2020 ■ QKC in Bachelor Elektrotechnik und Informationstechnik 2024 ■ QKC in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.49 RT - Regelungstechnik

Module ID	RT_BaTIN2020
Module Name	Regelungstechnik
Type of Module	Elective Modules
Recognized Course	RT - Control Engineering
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Jens Onno Krah/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Jens Onno Krah/Professor Fakultät IME

Learning Outcome(s)

Statisches Verhalten von Regelstrecken und Regelkreisen analysieren.
 Üben anhand von Kennlinienfeldern und Linearisierungen.
 Dynamisches Verhalten von Regelstrecken kennenlernen
 Empirische Betrachtungen durchführen, Differentialgleichungen aufstellen, Laplace-Transformation verwenden, Übertragungsfunktionen berechnen, Frequenzgang und Bode-Diagramm erstellen.
 Stabilität von Regelkreisen
 Algebraische Stabilitätskriterien anwenden, Nyquist-Kriterium verwenden.
 Parametrierung von Reglern
 Anwenden von Entwurfsverfahren, Entwerfen mit Frequenzkennlinien / Bode-Diagramm, Parametrieren durch Polvorgabe
 Gerätetechnik, zeitdiskreter Regelkreis
 Kennlernen von dedizierten Reglern und Differenzengleichungen
 Algorithmische Abtastregelungen parametrieren.
 Vermaschte Regelkreise
 Kennlernen von Kaskadenregelung, optional mit Vorsteuerung bzw. Störgrößenaufschaltung.
 Split-Range-Regelungen anwenden.
 Technisches Englisch
 Beispielsweise Fachgespräche sollen auf Englisch geführt werden können.

Module Contents

Lecture / Exercises

Transfer function of the closed control loop
 Selection of a appropriate controller for a given system
 Calculation of the stability of control loops

Lab

Handling and correct application of engineering tools
 Use and evaluation of the controllers
 Structure of control equipment
 Solving control problems

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights cf. exam regulations

Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	MA1, MA2, GE1, GE2, ASS, MT
Mandatory Prerequisites	Lab requires attendance in the amount of: 3 Praktikumstermine
Recommended Literature	■ Skript ■ Lutz, Wendt: Taschenbuch der Regelungstechnik, Verlag Harri Deutsch.
Use of the Module in Other Study Programs	■ RT in Bachelor Elektrotechnik 2020 ■ RT in Bachelor Elektrotechnik und Informationstechnik 2024 ■ RT in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.50 SE - Software Engineering

Module ID	SE_BaTIN2020
Module Name	Software Engineering
Type of Module	Mandatory Module
Recognized Course	SE - Software Engineering
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Hans Nissen/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Hans Nissen/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt Kompetenzen zur Erstellung von Software-Systemen in allen Projektphasen von der Dokumentation von Systemanforderungen (K.1, K.4), der Spezifikation und Modellierung von Systemeigenschaften (K.1, K.2, K.3, K.5, K.9), der Erstellung und Bewertung eines Systementwurfs für vorgegebene Qualitätsziele (K.1, K.2, K.5, K.9, K.10), der Prüfung von Systemeigenschaften (K.7, K.9, K.10) und der Erstellung leserlichen Programmcode (K.6). Im parallel laufenden Praktikum werden die Kompetenzen zur Erstellung umfangreicher technischer Texte (K.2, K.4), zum Entwurf eines Software-Systems unter Berücksichtigung von Qualitätszielen (K.1, K.3, K.5) zur Realisierung von Systemmodellen in Programmcode (K.6), zur Prüfung von Programmcode (K.6, K.7), zur Prüfung eines erstellten Software-Systems (K.7, K.9, K.10) und die Fähigkeit, komplexe fachbezogene Probleme zu sehen und Lösungen gegenüber Fachleuten mündlich argumentativ zu vertreten und mit anderen Studierenden weiterzuentwickeln (K.3, K.5, K.7, K.9, K.11, K.16) verstärkt.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs/Übungsteil unter Einbeziehung einer umfangreichen Fallstudie und betreut parallel dazu ein Praktikum, in dem die Studierenden ein kleines Software-System planen, entwerfen, realisieren und analysieren.

Wozu: Kompetenzen in der Entwicklung eines Software-Systems sind essentiell für technische Informatiker, die im HF 1 arbeiten wollen. Durch die Arbeit an einem komplexeren Beispielsystem erwerben die Studierenden zudem Erfahrungen, die essentiell für das HF 2 sind, u.a. Anforderungen spezifizieren, Systeme und Software-Architekturen entwerfen, realisieren und bewerten. Die Entwicklung eines Software-Systems über mehrere Praktikumstermine hinweg, vermittelt den Studierenden erste Erfahrungen in der Organisation umfangreicherer Systeme unter Berücksichtigung vorgegebener Qualitätsziele (HF 3).

Module Contents**Lecture / Exercises**

- | overview of Software Engineering
- | software development process models
- | requirements engineering
- | system and software specification techniques
- | modelling in UML
- | modern system architectures
- | quality assurance methods
- | tasks and methods of configuration management
- | documentation of requirements
- | evaluation of process models
- | development of system specifications
- | design and evaluation of different system architectures
- | design and evaluation of different software architectures
- | design of logical test cases and precise test procedures
- | Creation of readable program code

Lab

- | text understanding
- | usage of modelling tools
- | development of models
- | writing object-oriented programs in Java
- | testing of programs
- | transform model into code
- | develop system models from requirements
- | develop system design from system model
- | implementation of system models
- | verification of program code

Teaching and Learning ■ Lecture / Exercises
Methods ■ Lab

Examination Types with cf. exam regulations
Weights

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended ■ Modul PI2: Fundierte Kenntnisse in der Programmierung.
Prerequisites ■ Modul PP: Praktische Erfahrungen in der Programmierung.
 ■ programming skills in Java

Mandatory ■ Lab requires attendance in the amount of: 3 Testattermine
Prerequisites ■ Participation in final examination only after successful participation in Lab

Recommended Literature

- I. Sommerville: Software Engineering, Addison-Wesley, 2018.
- H. Balzert: Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering, Spektrum Akademischer Verlag, 3. Auflage, 2009.
- B. Oestereich: Analyse und Design mit der UML 2.5: Objektorientierte Softwareentwicklung, Oldenbourg Verlag, 11. Auflage, 2013.
- B. Brügge, A.H. Dutoit: Objektorientierte Softwaretechnik mit UML, Entwurfsmustern und Java, Pearson Studium, 2006.
- H. Balzert: Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb, Spektrum Akademischer Verlag, 3. Auflage, 2012.
- T. Posch, K. Birken, M. Gerdorf: Basiswissen Softwarearchitektur, 3. Auflage, dpunkt Verlag, 2011.
- A. Spillner, T. Linz: Basiswissen Softwaretest, 4. Auflage, dpunkt Verlag, 2010.
- B. Hindel, K. Hörmann, M. Müller, J. Schmied: Basiswissen Software-Projektmanagement, 3. Auflage, dpunkt Verlag, 2009.
- G. Popp: Konfigurationsmanagement, 3. Auflage, dpunkt Verlag, 2009.
- R. Oechsle: Java-Komponenten Grundlagen, prototypische Realisierung und Beispiele für Komponentensysteme, Hanser Verlag, 2013.

Use of the Module in Other Study Programs

SE in Bachelor Informatik und Systems-Engineering 2024

Specifics and Notes**Last Update**

19.7.2025, 14:32:16

6.51 SIG - Signalverarbeitung

Module ID	SIG_BaTIN2020
Module Name	Signalverarbeitung
Type of Module	Mandatory Module
Recognized Course	SIG - Signal Processing
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	3
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Harald Elders-Boll/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Harald Elders-Boll/Professor Fakultät IME

Learning Outcome(s)

Die Studierenden lernen die grundlegenden Verfahren und Algorithmen zur Analyse und Verarbeitung von diskreten Signalen und Systemen im Zeit- und Frequenzbereich anzuwenden, wie die diskrete Faltung, die DTFT, die z-Transformation und die DFT/FFT, die Eigenschaften zeitdiskreter Signale und Systeme im Zeit- und Frequenzbereich zu ermitteln, darzustellen und zu interpretieren, um analoge Signale digitalisieren, analysieren und mit Hilfe von zeitdiskreten Systemen verarbeiten zu können.

Module Contents

Lecture / Exercises

Basic Concepts: Classification of signals and systems, stability, causality
 LSI and LTI Systems: convolution, impulse response, stability, causality
 Fourier Series: real-valued and complex-valued coefficients, Gibbs' Phenomenon
 Fourier Transform: derivation, properties, calculation, transfer function and frequency response, energy density spectrum
 Sampling: sampled vs. discrete time signals, sampling theorem, aliasing
 DTFT: derivation, properties, calculation of the DTFT, frequency response
 DFT: derivation, properties, leakage effect

Evaluation of the stability of LTI and LSI systems in the time domain
 Calculation of the output signal of LSI systems using convolution
 Calculation of the Fourier transform and the DTFT
 Implementation of FIR systems by programming the discrete convolution
 Implementation of simple IIR systems
 Evaluation of the filter characteristics based on the frequency response and the auditory impression

Lab

sampling input and output signals of a continuous-time (CT) system
 basic algorithms of signal processing

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites	▪ Modul MA1: trigonometrische, exp., log-Funktionen; Grenzwerte; komplexe Rechnung ▪ Modul MA2: Integral- und Differentialrechnung; unendliche Reihen; Reihenentwicklung ▪ Modul PI1: Prozedurale Programmierung ▪ Knowledge of the following mathematical subjects: Trigonometric functions, exponential function, logarithm, complex calculus, integral and differential calculus, series expansion, geometric series, partial fraction expansion. Knowledge of the following physical subjects: power and energy.
----------------------------------	--

Mandatory Prerequisites	▪ Lab requires attendance in the amount of: 2 Praktikumstermine ▪ Participation in final examination only after successful participation in Lab
--------------------------------	---

Recommended Literature	▪ Jens Rainer Ohm und Hans Dieter Lüke, Signalübertragung, Springer, 2014 ▪ Martin Meyer, Signalverarbeitung, Springer Vieweg, 2014 ▪ Martin Werner, Signale und Systeme, Springer Vieweg, 2008 ▪ Bernd Girod u.a., Einführung in die Systemtheorie, Springer Vieweg, 2007
-------------------------------	--

Use of the Module in Other Study Programs	SIG in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.52 SM - Software Management

Module ID	SM_BaTIN2020
Module Name	Software Management
Type of Module	Elective Modules
Recognized Course	DOPE - DevOps and Platform-Engineering
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. René Wörzberger/Professor Fakultät IME
Lecturer(s)	Prof. Dr. René Wörzberger/Professor Fakultät IME

Learning Outcome(s)

(WAS) Die Studierenden erlernen und wenden Werkzeuge und Methoden an zur (a) professionellen Entwicklung von Systemen im Team, (b) zur Qualitätssicherung von Systemen, (c) zur Automatisierung von Entwicklungsprozessen und (d) zum Betrieb von Systemen in Cloud-Infrastrukturen ,
(WOMIT) indem ihnen besagte Inhalte und Fertigkeiten in Vorlesungen/Übungen vermittelt werden und indem sie sie in einer Reihe von vorzubereitenden, aufeinander aufbauenden Praktikumsaufgaben praktisch anwenden,
(WOZU) um später in hochdynamischen Entwicklungsumfeldern qualitativ hochwertige Software-Systeme über den gesamten Lebenszyklus verantworten zu können.

Module Contents

Lecture / Exercises

- inner workings of the source code management systems Git
- organizing teams with GitLab functions
- automate builds with Apache Maven
- continuous integration and delivery (CI/CD) with GitLab Runner
- test automation with JUnit
- developing mocks with Mockito
- automating web ui tests with Selenium
- automating web ui tests with Selenium
- measuring code quality with Sonarqube
- on-prem and cloud infrastructures
- creating a system cluster in the Google Cloud
- container virtualization with Docker
- container orchestration with Kubernetes

Lab

- how to develop in teams with GitLab
- adding and developing a code base with Git
- creating build scripts with Maven
- Implementation of tests with JUnit, Mockito, Selenium, and JMeter
- containerization and deployment with Docker and Kubernetes
- set-up of a system cluster in the Google Cloud including (continuous) deployment of releases

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	▪ Modul PI1: Ein fortgeschrittenes Verständnis des "gemanagten" Subjekts im Java-Source-Code ist erforderlich. ▪ Modul PI2: Ein fortgeschrittenes Verständnis des "gemanagten" Subjekts im Java-Source-Code ist erforderlich. ▪ Modul SP: Ein fortgeschrittenes Verständnis des "gemanagten" Subjekts im Java-Source-Code ist erforderlich. Zudem sind für die Bearbeitung von Praktikumsaufgaben fortgeschrittene Kenntnisse mit der Arbeit im Team erforderlich. ▪ (1) advanced programming skills in Java ▪ (2) experiences with development projects in teams ▪ (3) basic knowledge in software engineering
Mandatory Prerequisites	
Recommended Literature	▪ H. Atkins et al.: Building Secure and Reliable Systems, O'Reilly, 2020 ▪ B. Betsy et al.: Site Reliability Engineering, O'Reilly, 2020 ▪ J. Humble et al.: Accelerate - The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations, IT Revolution, 2018
Use of the Module in Other Study Programs	▪ SM in Bachelor Elektrotechnik 2020 ▪ DOPE in Bachelor Informatik und Systems-Engineering 2024
Permanent Links to Organization	<u>Ilu course</u>
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.53 SMO - Smart Mobility Components

Module ID	SMO_BaTIN2020
Module Name	Smart Mobility Components
Type of Module	Elective Modules
Recognized Course	SMO - Smart Mobility Components
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Kai Kreisköther/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Kai Kreisköther/Professor Fakultät IME

Learning Outcome(s)

WAS

Die Studierenden können

- konzeptionelle und technologische Entscheidungen für Komponenten und Gesamtsystemarchitekturen von smarten Mobilitäts- und Logistiksystemen treffen,

WOMIT

indem sie

- Ausprägungen und Bausteine von smarten Mobilitäts- und Logistiksystemen kennenlernen,
- die Vielfalt der Komponenten in smarten Mobilitäts- und Logistiksystemen und deren prinzipielle technologische Umsetzung kennenlernen,
- im Rahmen einer Projektarbeit eigenständig eine technologische Komponente eines smarten Mobilitäts- oder Logistiksystems konzipieren, spezifizieren und entwickeln/testen,

WOZU

um

- im Rahmen der konzeptionellen und technologischen Gestaltung von smarten Mobilitäts- und Logistiksystemen als mündiges Projektmitglied auftreten und mitarbeiten zu können und
- technologische Komponenten (bspw. Sensoren) sowie Gesamtsystemarchitekturen von smarten Mobilitäts- und Logistiksystemen konzipieren, spezifizieren und entwickeln/testen zu können.

Module Contents

Lecture / Exercises

Project

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Project
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
Recommended Prerequisites	
Mandatory Prerequisites	Project requires attendance in the amount of: 8 Review-Termine á 90 Minuten
Recommended Literature	■ Barbara Flügge: Smart Mobility, 2020 ■ Alaa Khamis: Smart Mobility, 2021 ■ Fateh Belaïd und Anvita Arora: Smart Cities, 2024 ■ Michèle Finck et al: Smart Urban Mobility, 2020 ■ Vincenzo Piuri et al: AI and IoT for Smart City Applications, 2022 ■ Dagmar Cagáňová und Natália Horňáková: Industry 4.0 Challenges in Smart Cities, 2023
Use of the Module in Other Study Programs	■ SMO in Bachelor Elektrotechnik 2020 ■ SMO in Bachelor Elektrotechnik und Informationstechnik 2024 ■ SMO in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 15:19:39

6.54 SMP - Signalverarbeitung mit Matlab/Python und μ C

Module ID	SMP_BaTIN2020
Module Name	Signalverarbeitung mit Matlab/Python und μ C
Type of Module	Elective Modules
Recognized Course	SMP - Signalprocessing using Matlab/Python and Microprocessors
ECTS credits	5
Language	deutsch und englisch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. Harald Elders-Boll/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Harald Elders-Boll/Professor Fakultät IME ▪ Prof. Dr. Uwe Dettmar/Professor Fakultät IME ▪ Prof. Dr.-Ing. Christoph Pörschmann/Professor Fakultät IME

Learning Outcome(s)

Entwurf, Analyse und Implementierung von Systemen und Algorithmen zur Signalverarbeitung in Software und Hardware durch praktische Übungen und das selbstständige Bearbeiten von Hard- und/oder Software-Projekten, um erfolgreich neue Systeme und Anwendungen der Signalverarbeitung in unterschiedlichen Anwendungsbereichen entwickeln zu können

Module Contents**Lecture**

Principles of Digital Signal Processing:

Sampling and Reconstruction

Digital Filters

DFT and FFT

Fast FFT-based Convolution

Spectral Analysis

Signal Generation

Real-time Signal Processing:

Interrupt and Polling

Block-based Signal Processing

Apply fundamentals of digital signal processing:

Understanding of and ability to explain the fundamental principles of digital signal processing

Ability to compare and evaluate different digital filter types and different implementations

Implementation of real-time DSP:

Ability to explain the general problem of real-time DSP

Ability to name aspects influencing the processing speed

Understanding of and ability to explain the fundamental methods of real-time digital signal processing

Lab

Implementation of fundamental methods and procedures for signal processing in Python/Matlab and on microprocessors.

Project

Implementation Python/Matlab:

Program, debug and optimize algorithm in Python Matlab.

Implementierung on microprocessor:

Port algorithm to target microprocessor platform

Familiarity with development environment

Optimize algorithm for target platform

Solve complex tasks in team work:

Plan simple projects

Keep agreements and deadlines

Schedule and carry out reviews

Implementation of DSP algorithm on microprocessor platform:

Understand given methods for digital signal processing

Obtain required references for given methods

Translate mathematical methods to program code

Test, verify, and optimize program code

Presentation of results:

Presentation of project results

Teaching and Learning ■ Lecture

Methods ■ Lab

■ Project

Examination Types with cf. exam regulations

Weights

Workload 150 Hours

Contact Hours	45 Hours $\triangleq$ 4 SWS
Self-Study	105 Hours
Recommended Prerequisites	▪ Modul SIG: Grundbegriffe von zeitdiskreten Signale und Systemen, Stabilität, Kausalität LSI-Systeme: zeitdiskrete Faltung zeitdiskreter Signale, FIR und IIR Filter Abtastung, Abtasttheorem, Aliasing DTFT, Frequenzgang z-Transformation, Zusammenhang zwischen Frequenzgang und Übertragungsfunktion, Blockschaltbilder DFT, Leakage-Effekt ▪ Basic procedural programming skills - Basic knowledge of digital signal processing: Sampling Theorem, Digital Filter, Fourier Transform
Mandatory Prerequisites	▪ Project requires attendance in the amount of: 8 Termine ▪ Participation in final examination only after successful participation in Project
Recommended Literature	▪ Welch, Wright, Morrow: Real-Time Digital Signal Processing (CRC Press)
Use of the Module in Other Study Programs	▪ SMP in Bachelor Elektrotechnik 2020 ▪ SMP in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ SMP in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.55 SOP - Systems on Programmable Chips

Module ID	SOP_BaTIN2020
Module Name	Systems on Programmable Chips
Type of Module	Elective Modules
Recognized Course	SOP - Systems on Programmable Chips
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Tobias Krawutschke/Professor Fakultät IME

Learning Outcome(s)

Die Studierenden erwerben die Kompetenz zum Entwurf, Implementierung und Test eines modernen signalverarbeitenden Systems, indem sie an einfachen Beispielen die FPGA-Technologie mittels Hardware-Beschreibungssprache benutzen lernen, dies dann auf eine komplexere Aufgabenstellung aus der Audio-Signalverarbeitung anwenden, damit sie später FPGAs als "Problemlöser" für leistungsfähige Verarbeitung von Signalen einsetzen können.

Die Studierenden erwerben die Kompetenz zum Entwurf eines Hardware-Software-Systems, indem sie auf der Basis ihrer Kenntnisse in hardwarenaher Programmierung und der Erstellung programmierter digitaler Systeme ein Beispielsystem auf einem SoPC (System on Programmable Chip) erstellen, damit sie später diese Technologie für verschiedenste Aufgaben, bei denen viele Daten in kürzester Zeit bearbeitet werden müssen, anwenden können.

Module Contents**Lecture / Exercises**

- 1) Digital system modelling using
 - Boolean algebra
 - Schematic (using digital basic components)
 - Finite State Automata (FSA)
 - Extended FSA, Statecharts
 - Controlflow/Dataflow systems
 - VHDL
- 2) Digital technology
 - Understanding of typical digital circuits (CMOS technology)
 - Understanding, description and classification of runtime effects
 - Knowledge and variants of programmable units (PLD, FPGA)
- 3) SoC/SoPC-Systeme
 - System construction
 - IO access using machine-near programming
 - Interrupts, alarm
 - Programming automata / CFDF systems
 - Rules to partition hardware and software components
 - Design of coupling of HW/SW components

Lab

- Getting competencies in analysis, modelling and implementation of the hardware part of an audio signal processing system
- 1) Analysis of interface to the CoDec and creation of a system reading in and writing out samples (copy-machine)
 - 2) Development of a FIR filter working on the samples
 - 3) Development of a simple echo producer working in the time domain

- Getting competencies in analysis, modelling and implementation of an audio signal processing system in software
- 1) Analysis of interface to the CoDec and creation of a system reading in and writing out samples (copy-machine)
 - 2) Development of a N-stage averaging mean filter working on the samples
 - 3) Development of a simple echo producer working in the time domain
 - 4) Measurement and optimization of the system since it reaches the performance limit of standard microcontrollers

- Realization of the example system as a HW/SW system with input of parameter values for echo and FIR filter unit
- 1) System partition HW/SW
 - 2) Protocoll specification between HW and SW components
 - 3) Realization of User Interface (Input of echo and filter parameters, general system control)
 - 4) Realization of protocoll components
 - 5) Validation with FPGA Board
 - 6) Comparison of solutions HW / SW / SoPC in report

Teaching and Learning ■ Lecture / Exercises

Methods ■ Lab

Examination Types with Weights cf. exam regulations

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\pm$ 4 SWS
----------------------	----------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	■ Modul DR: Grundlagen Digitale Logik Grundlagen Automaten Grundlagen Mikroprozessor Grundlagen Hardwarenahe Programmierung in C ■ Modul PP: Programmier-Kompetenzen Kompetenz zur Textanalyse und Extraktion der Informationen für einen Programmwurf Strukturierte Analyse ■ Modul BVS1: Konzepte des Multitasking ■ Fundamentals of digital systems * Design Methods (Boolean Algebra, Automata) * Basic knowledge of digital technology including hardware description language ■ Fundamentals Programming * Hardwareoriented Programming with C * Programming experience * Knowledge and first experiences in reactive programming, especially using interrupts ■ Fundamentals of signal processing, esp. digital filters (FIR)
----------------------------------	--

Mandatory Prerequisites

Recommended Literature	■ Hamblen, Furman: Rapid Prototyping of Digital Systems, Kluwer Academic Publishing ■ Wakerly: Digital Design: Principles and Practices, Prentice Hall ■ D. Gajski: Embedded System Design, Springer Verlag New York ■ U. Meyer-Baese: Digital Signal Processing with Field Programmable Gate Arrays
-------------------------------	--

Use of the Module in Other Study Programs	■ SOP in Bachelor Medientechnologie 2024 ■ SOP in Bachelor Informatik und Systems-Engineering 2024
--	--

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.56 SWP - Softwarepraktikum

Module ID	SWP_BaTIN2020
Module Name	Softwarepraktikum
Type of Module	Mandatory Module
Recognized Course	SWP - Software Lab
ECTS credits	6
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Hans Nissen/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Hans Nissen/Professor Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt unterschiedliche Kompetenzen, die für die team-orientierte Entwicklung komplexer Software-Systeme erforderlich sind:
 der Entwurf von System-Komponenten gemäß Spezifikation und textuellen Anforderungen im Team (K.1, K.4, K.2, K.3, K.5, K.10, K.13),
 die Implementierung von System-Komponenten gemäß Entwurf im Team (K.13, K.16, K.6, K.10),
 die Prüfung einer implementierten Komponente auf Korrektheit (K.7),
 die technische Dokumentation einer Implementierung (K.8, K.16),
 die Integration von Komponenten eines Systems zusammen mit anderen Entwicklerteams (K.1, K.10, K.13, K.16),
 die Prüfung eines integrierten System auf Korrektheit (K.7),
 das zeitliche und inhaltliche Strukturieren und Organisieren eines eigenen Projekt (K.11, K.13),
 das (teilweise) eigenständige Erlernen und Anwenden verschiedener typischer Werkzeuge in der Praxis (K.9, K.15).

Womit: Die Veranstaltung besteht aus einem einsemestrigen Projekt, welches die Studierenden in einem Team aus drei bis fünf Mitgliedern bearbeiten.
 Der Schwerpunkt der Projektarbeit der Studierenden liegt dabei auf der Umsetzung von Anforderungen und Spezifikationen in einem lauffähigen Gesamtsystem.
 Der Dozent gibt die Anforderungsspezifikation, die System-Spezifikation und den Grobentwurf des Gesamtsystems vor. Jedes Team entwirft, realisiert und testet eine Komponente des Gesamtsystems und integriert die eigenen Komponenten mit den Komponenten anderer Teams zu einem lauffähigen Gesamtsystem.
 Jedes Team organisiert ihr Entwicklungsprojekt selbst. Vorgegeben durch den Dozenten werden die wesentlichen Meilensteine des Projekts, an denen die Ergebnisse mit den Teams besprochen und bewertet werden.

Wozu: In dem praxisnahen Projekt in dieser Veranstaltung sammeln die Studierenden praktische, realitätsnahe Erfahrungen in allen typischen Arbeitsfeldern eines Bachelorabsolventen:
 die Studierenden entwickeln eine Teilkomponente eines spezifizierten Gesamtsystems selbständig in einem Teams (HF.1),
 sie analysieren und bewerten die entwickelte Komponente bezüglich einer vorgegebenen Spezifikation und das Gesamtsystem bezüglich der Kundenanforderungen (HF.2),
 sie organisieren ihr eigenes Entwicklungsprojekt zeitlich und (teilweise) inhaltlich selbst und übernehmen selbstverantwortlich die Strukturierung einer Komponente des Gesamtsystems (HF.3),
 die Studierenden kommunizieren während des gesamten Projekts intensiv und über verschiedene Kanäle mit dem Dozenten als dem Auftraggeber und späterem Anwender des Projekts, den Mitgliedern des eigenen Teams und in der Phase der Integration auch mit den Mitgliedern mehrerer anderer Teams (HF.4).

Module Contents**Project**

handling of semi-formal specifications
 team-oriented software development
 structure and organize a project in terms of time and content
 application of tools
 integrated development environment
 version management
 error management
 test tools
 cooperation and communication tools
 multiple Java-APIs
 data bases
 design of system component according to specification and requirement document
 implementation of system component according to design in a team
 verify correctness of system components
 technical documentation of system components
 integrate components into system in cooperation with other design teams
 verify integrated system

Teaching and Learning Methods

Project

Examination Types with Weights

cf. exam regulations

Workload 180 Hours

Contact Hours 12 Hours $\pm$ 1 SWS

Self-Study 168 Hours

Recommended Prerequisites

- Modul PI1: sehr gute praktische Beherrschung der Programmiersprache Java
- Modul PP: Das Programmierpraktikum vermittelt bestimmte, spezifische softwarearchitektonisch Voraussetzungen, Stile und Idiome, auf denen das Softwarepraktikum aufbaut.
- Modul DB1: Grundlagen der Speicherung in einer Datenbank und der Anfrage von Daten
- very good programming skills
 Knowledge in Software Engineering
 Knowledge in Data Bases

Mandatory Prerequisites

- Module SE: Spezifikation und Modellierung von Systemen und Software mit UML, Modularisierung in Java, einfache Entwurfsmuster, grundlegende Verfahren zum Prüfen von Software, verschiedene Architekturen von Systemen und Software, Grundbegriffe der Qualitätssicherung, Kenntnisse in Versionsverwaltung
- Module PI2: Das Softwarepraktikum setzt Konzepte für das Programmieren im Großen voraus, die in PI2 vermittelt werden. Fehlen diese Voraussetzungen bei einzelnen Team-Mitgliedern, ist der Team-Erfolg im Softwarepraktikum gefährdet.
- Project requires attendance in the amount of: 7 Termine

Recommended Literature

- I. Sommerville: Software Engineering, Addison-Wesley, 2007.
- H. Balzert: Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb, Spektrum Akademischer Verlag, 3. Auflage, 2012.
- B. Brügge, A.H. Dutoit: Objektorientierte Softwaretechnik mit UML, Entwurfsmustern und Java, Pearson Studium, 2006.
- H. Balzert: Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering, Spektrum Akademischer Verlag, 3. Auflage, 2009.

Use of the Module in Other Study Programs	SWP in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.57 SYP - Systementwicklungs-Projekt

Module ID	SYP_BaTIN2020
Module Name	Systementwicklungs-Projekt
Type of Module	Mandatory Module
Recognized Course	SYP - System Development Project
ECTS credits	7
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	5
Frequency of Course	every winter term
Module Coordinator	Prof. Dr. René Wörzberger/Professor Fakultät IME
Lecturer(s)	Prof. Dr. René Wörzberger/Professor Fakultät IME

Learning Outcome(s)

(WAS) Studierende entwerfen und entwickeln im Rahmen eines eigenen Software-Entwicklungsprojekts im Team ein Software-System (WOMIT) indem sie das Projekt geeignet planen, Anforderungen systematisch erheben, das System nach geeigneten Qualitätskriterien entwerfen, zunächst prototypische implementieren und präsentieren, durch geeignete Maßnahmen (Tests etc.) qualitätssichern und in benutzbarer und dokumentierter Form an ihren Auftraggeber übergeben
(WOZU) damit sie im späteren Beruf als Mitglied eines Entwicklungsteams wirksam mitarbeiten können.

Module Contents**Lab**

- | team-oriented software development
- | communication with customers
- | coverage of whole software lifecycle
- | organize project regarding schedule and content
- | give presentations
- | develop schedule for development project
- | elicit and document customer requirements
- | specify and model system according to requirements
- | design system according to specification and quality requirements
- | implement system
- | document technical system details
- | verify implemented system
- | create user documentation
- | present own solutions to customers

Teaching and Learning Lab**Methods**

Examination Types with Weights cf. exam regulations

Workload 210 Hours

Contact Hours 12 Hours $\pm$ 1 SWS

Self-Study 198 Hours

Recommended Prerequisites	▪ Modul PI1: Die zwingende Voraussetzung PI2 baut auf PI1 auf. ▪ Modul SWP: Das Arbeiten im Team sollte zunächst bei der enger gefassten Aufgabenstellung des SWP eingeübt werden.
Mandatory Prerequisites	▪ Module PI2: Programmierkenntnisse auf dem Niveau von PI2 sind unabdingbar für einen ausreichenden Beitrag im jeweiligen SYP-Team ▪ Module SE: In SYP sollen die Kenntnisse in sämtlichen Arbeitsbereichen des Software-Engineerings (SE) angewendet werden, die zuvor in SE vermittelt wurden. Es reicht für die Teilnahme in SYP, wenn das Praktikum in SE bestanden wurde. ▪ Lab requires attendance in the amount of: 5 Termine
Recommended Literature	▪ keine
Use of the Module in Other Study Programs	▪ SYP in Bachelor Informatik und Systems-Engineering 2024 ▪ PLSYP in Master Technische Informatik 2020 ▪ PLSYP in Master Informatik und Systems-Engineering 2024
Permanent Links to Organization	<u>llu course</u>
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.58 UT - Übertragungstechnik

Module ID	UT_BaTIN2020
Module Name	Übertragungstechnik
Type of Module	Elective Modules
Recognized Course	UT - Digital Communications
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Uwe Dettmar/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Uwe Dettmar/Professor Fakultät IME

Learning Outcome(s)

Was? Die Übertragung von in binärer Form vorliegender Daten über gestörte Kanäle durch Modulation inklusive des Entwurfs von Modulator und Demodulator
Womit? Unter Anwendung von Verfahren und Algorithmen der digitalen Übertragungstechnik
Wozu? Zur Realisierung einer an die Eigenschaften des Kanals angepassten zuverlässigen Datenübertragung in kommunikationstechnischen Systemen.

Module Contents**Lecture / Exercises**

The underlying concept of this module is a combination of lecture and tutorial. After a lecture block the subjects taught are actively trained using Matlab/Octave and Python programs.

Syllabus:

- Short history of telecommunications
- models and content of communications
- basic terms
- signals, systems, and modulation
- multicarrier
- channels and elements of a transmission link
- decision theory
- link budget calculation

Students shall deepen their knowledge by self-study of literature and internet resources and discuss their results in small learning groups as teamwork.

By working with small exercises and programs during the presence time students are able to actively train their knowledge. More extensive problems are solved and discussed in the second part of the lesson to activate the student's capabilities to solve relevant problems.

Students further learn

- to analyze communication systems and to estimate their performance
- to compare and rate algorithms and methods
- to apply their knowledge to technical problems

Lab

Students accomplish lab problems using Jupyter Notebooks.

Simulations are run with the help of complete or partly complete Python programs. Students collect and present the data by graphical representation.

Matlab in combination with the Communications Toolbox ist used for more extensive simulation tasks.

- students train their capabilities to solve technical problems by the help of programmes
- students analyze and simulate digital communication systems and rate their performance
- they train their self-management and their problem focused thinking and acting
- students practice solving problems in a team and train their communication skills.

Teaching and Learning ■ Lecture / Exercises

Methods ■ Lab

Examination Types with Weights cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites	■ Modul MA1: Elementare Funktionen, Differentialrechnung, Lineare Gleichungssysteme ■ Modul MA2: Komplexe Rechnung, Integralrechnung, Lineare Algebra ■ Modul EG: Grundbegriffe, elektrische und magnetische Feldgrößen, Komplexe Wechselstromrechnung ■ Modul SIG: Signale, Impulsantwort, Faltung, Fourier Transformation und Spektren ■ Students should have basic knowledge in algebra, linear algebra, and stochastics as well as the capabilities to write small programs in a standard programming language. In the framework of this module Python and Matlab/Octave is used for simulations. They should understand physical quantities and units and have basic capabilities in the application of complex AC calculation. Basic knowledge from signal theory and Fourier transform are required, too.
Mandatory Prerequisites	■ Lab requires attendance in the amount of: 4 Praktikumstermine ■ Participation in final examination only after successful participation in Lab
Recommended Literature	■ BOSSERT, M. : Einführung in die Nachrichtentechnik. Oldenbourg Verlag, 2012. ■ MEYER, M. : Kommunikationstechnik. 4. Vieweg und Teubner, 2019. ■ JOHNSON, SETHARES, KLEIN: Software Receiver Design, Cambridge 2011 ■ PROAKIS, J. G. ; SALEHI, M. : Digital Communications. 5. McGraw-Hill, 2008.
Use of the Module in Other Study Programs	■ UT in Bachelor Elektrotechnik 2020 ■ UT in Bachelor Elektrotechnik und Informationstechnik 2024 ■ UT in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.59 VDS - Verteilte Datenverarbeitungssysteme

Module ID	VDS_BaTIN2020
Module Name	Verteilte Datenverarbeitungssysteme
Type of Module	Elective Modules
Recognized Course	VDS - Verteilte Datenverarbeitungssysteme
ECTS credits	5
Language	deutsch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every term
Module Coordinator	Prof. Dr. Andreas Behrend/Professor Fakultät IME
Lecturer(s)	Prof. Dr. Andreas Behrend/Professor Fakultät IME

Learning Outcome(s)

Was: Im Rahmen der Vorlesung sollen prinzipielle Big Data Konzepte vermittelt werden. Das beinhaltet Fault Tolerance in verteilten Systemen, verschiedene Fehlerbehandlungsmechanismen (z.B. Replikation, Wiederherstellung) sowie Bloom Filter und N-Grams als Techniken zur Optimierung von Datenabfragen. Zudem werden Konzepte wie die Jaccard Similarity sowie Min/LS Hashing vorgestellt, um große Datenmengen effizient vergleichen und analysieren zu können sowie die grundlegenden Prinzipien von NoSQL-Systemen, einschließlich Replikation, CAP-Theorem, Multiversion- und BASE-Konsistenz, Consistent Hashing, Casual Consistency, Time-to-Live (TTL), Lamport- und Vector Clocks vermittelt.

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum erarbeiten die Studierenden Softwarelösungen auf Basis von NoSQL-Systemen in Kleingruppen. Im Praktikum analysieren die Studierenden dafür Problemstellungen zur Datenspeicherung und -analyse (K1, K2, K4), programmieren Lösungen auf der Grundlage anerkannter Konzepte und Methoden (K3) mit Hilfe von aktuellen Systemen und Werkzeugen (K6, K9) und evaluieren die Softwarelösung (K7). Dazu verwenden sie verfügbare Dokumentationen (K8, K15) und passen die Softwaresysteme bzw. Werkzeuge entsprechend an (K10). Zum Abschluss des Praktikums müssen die Studierenden ihre Lösungen vorstellen und verteidigen (K8, K13, K16).

Wozu: In komplexen Softwaresystemen muss die Verarbeitung großer, heterogener Datenmengen oft nebenläufig und verteilt erfolgen. Kenntnisse über die Grundlagen und die Programmierung verteilter Datenverarbeitungssysteme sind somit essentiell für die Erstellung moderner Software (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2) sowie zur Organisation bzw. zum Betrieb von Systemen, die nebenläufig und verteilt arbeiten (HF3). Die Durchführung im Team und mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Durch die Auseinandersetzung mit diesen Themen entwickeln die Studierenden das nötige Wissen und die Fähigkeiten, um verteilte Datenverarbeitungssysteme zu entwerfen, zu implementieren und zu optimieren, die für moderne Big Data-Anwendungen erforderlich sind.

Module Contents

Lecture / Exercises

Lab

Teaching and Learning Methods	▪ Lecture / Exercises ▪ Lab
--------------------------------------	--

Examination Types with Weights	cf. exam regulations
---------------------------------------	----------------------

Workload	150 Hours
-----------------	-----------

Contact Hours	45 Hours $\triangleq$ 4 SWS
----------------------	-----------------------------

Self-Study	105 Hours
-------------------	-----------

Recommended Prerequisites	▪ Modul PI1: Sicherer Umgang mit einer Programmiersprache. ▪ Modul PI2: Sicherer Umgang mit einer Programmiersprache. ▪ Modul DB1: Datenanalyse mittels SQL sowie Architektur von Datenverarbeitungssystemen
Mandatory Prerequisites	Participation in final examination only after successful participation in Lab
Recommended Literature	▪ Martin Kleppmann: "Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems" ▪ Pramod J. Sadalage und Martin Fowler: "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence" ▪ Lena Wiese: "NoSQL-Datenbanken: Konzepte – Technologien – Anwendungen"
Use of the Module in Other Study Programs	VDS in Bachelor Informatik und Systems-Engineering 2024
Specifics and Notes	
Last Update	19.7.2025, 14:32:16

6.60 VMA - Programmierung verteilter und mobiler Anwendungen

Module ID	VMA_BaTIN2020
Module Name	Programmierung verteilter und mobiler Anwendungen
Type of Module	Elective Modules
Recognized Course	MPR - Mobile Device Programming
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr. Cartsten Vogt/Professor Fakultät IME
Lecturer(s)	▪ Prof. Dr. Cartsten Vogt/Professor Fakultät IME ▪ Marcel Henk/ wissenschaftlicher Mitarbeiter Fakultät IME

Learning Outcome(s)

Was: Das Modul vermittelt Kompetenzen zur Programmierung von Mobilgeräten, insbesondere von Smartphones. In praktischer Arbeit analysieren die Studierenden Problemstellungen (K2), implementieren Lösungen mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7). Sie recherchieren dazu in Online-Dokumentationen (K8, K15) und passen vorhandene Software an (K10). Darüber hinaus befähigt das Modul die Studierenden, die Folgen bei der Programmierung und beim Einsatz von Mobilgeräten einzuschätzen (K14).

Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K8, K13, K16).

Wozu: Mobilgeräte spielen im privaten und professionellen Umfeld eine zentrale Rolle und somit auch Kenntnisse, sie zu programmieren und in verteilte Systeme zu integrieren (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden zudem weitere Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

Module Contents**Lecture / Exercises**

fundamental terms and techniques
 characteristic properties of mobile devices
 overview of current mobile operating systems and programming platforms
 steps of mobile device programming (code development, emulation, and installation)

Mobile device programming with one or multiple current systems (Remark: This main part of the course will be continuously adapted to the current state of the art and the market. This document therefore lists only the main topics that will probably be covered.)

components of a mobile application
 graphical user interfaces
 data storage
 concurrency
 data communication, esp. Internet access
 location-based services
 security

using programming environments for mobile devices

programming smartphone applications of medium complexity

assessing the risks in the programming and usage of mobile devices

Lab

Smartphone programming on a selected system - details see "Vorlesung/Übung"

using programming environments for smartphones

implementation of smartphone applications of medium complexity in small teams

Teaching and Learning Methods

- Lecture / Exercises
- Lab

Examination Types with Weights

cf. exam regulations

Workload 150 Hours

Contact Hours 45 Hours $\pm$ 4 SWS

Self-Study 105 Hours

Recommended Prerequisites

- Modul PI1: Sicherer Umgang mit einer objektorientierten Programmiersprache.
- Modul PI2: Sicherer Umgang mit einer objektorientierten Programmiersprache.
- Modul BVS1: Struktur und Funktionalität von Betriebssystemen. Grundkenntnisse in der nebenläufigen Programmierung (Threading) und in der Netzwerkprogrammierung (Sockets).
- Modul DB1: Grundkenntnisse in relationalen Datenbanken, auch Programmierung damit.
- Modul NP: Grundkenntnisse in Internet-Protokollen.
- object-oriented programming (ideally Java)
- structure and functions of operating systems
- programming with concurrency / threading and with sockets
- communication protocols for data networks
- relational databases

Mandatory Prerequisites

Lab requires attendance in the amount of: 1 Testattermin

Recommended Literature

- Aufgrund des sehr dynamischen Fachgebiets können Literaturangaben immer nur jeweils unmittelbar zu Veranstaltungsbeginn gemacht werden. Aktuell: <http://developer.android.com>; Künneth, Android 8 - Das Praxisbuch für Entwickler, Rheinwerk 2018

Use of the Module in Other Study Programs	▪ VMA in Bachelor Elektrotechnik 2020 ▪ MPR in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ MPR in Bachelor Informatik und Systems-Engineering 2024
--	---

Specifics and Notes

Last Update	19.7.2025, 14:32:16
--------------------	---------------------

6.61 XIB - Fachübergreifende Kompetenzen und Soft-Skills

Module ID	XIB_BaTIN2020
Module Name	Fachübergreifende Kompetenzen und Soft-Skills
Type of Module	Mandatory Module
Recognized Course	XGA - Participation in appointment committees
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	5
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Bachelor Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	
Learning Outcome(s)	Die Studierenden lernen, über die fachbezogenen Grenzen ihres Studiums hinweg zu schauen. Sie sind in der Lage, internationale, inter-/transdisziplinäre und/oder interkulturelle Aspekte ihres zukünftigen Berufs zu erkennen, einzuordnen, ihr Verhalten darauf einzustellen und auch in fremdem Kontext sicher zu agieren. Das konkrete Lehrangebot wird in der Regel erst kurzfristig zu Beginn des jeweiligen Semesters festgelegt. Es kann unterschiedlichste Themen behandeln; eine Zusammenarbeit mit anderen Fakultäten oder Instituten ist vorgesehen. Je nach konkret gewähltem Lehrangebot werden die u.a. Kompetenzen unterschiedlich intensiv vermittelt. Das Modul kann auch durch Teilnahme an mehreren verschiedenen kleineren Lehrveranstaltungen erfüllt werden, sofern diese zu den Modulzielen beitragen und die erforderlichen ECTS-Punkte in Summe erreicht sind.
Module Contents	
Project	
Teaching and Learning Methods	Project
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	12 Hours $\pm$ 1 SWS
Self-Study	138 Hours
Recommended Prerequisites	
Mandatory Prerequisites	
Recommended Literature	
Use of the Module in Other Study Programs	▪ XIB in Bachelor Elektrotechnik 2020 ▪ XGA in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ XGA in Bachelor Informatik und Systems-Engineering 2024 ▪ XGA in Master Informatik und Systems-Engineering 2024
Specifics and Notes	Participation in appointment committees as a student member is recognized. The number of ECTS points recognized depends on the number of hours of committee work demonstrated. One ECTS point is recognized for every 25 hours of committee work. The chairperson of the appointment committee awards the ECTS and certifies them. The student is expected to be actively involved in the work.
Last Update	19.7.2025, 14:32:16

6.62 XIB2 - Fachübergreifende Kompetenzen und Soft-Skills

Module ID	XIB2_BaTIN2020
Module Name	Fachübergreifende Kompetenzen und Soft-Skills
Type of Module	Elective Modules
Recognized Course	XGA - Participation in appointment committees
ECTS credits	5
Language	deutsch, englisch bei Bedarf
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every term
Module Coordinator	Studiengangsleiter(in) Bachelor Technische Informatik / Informatik und Systems-Engineering
Lecturer(s)	
Learning Outcome(s)	Die Studierenden lernen, über die fachbezogenen Grenzen ihres Studiums hinweg zu schauen. Sie sind in der Lage, internationale, inter-/transdisziplinäre und/oder interkulturelle Aspekte ihres zukünftigen Berufs zu erkennen, einzuordnen, ihr Verhalten darauf einzustellen und auch in fremdem Kontext sicher zu agieren. Das konkrete Lehrangebot wird in der Regel erst kurzfristig zu Beginn des jeweiligen Semesters festgelegt. Es kann unterschiedlichste Themen behandeln; eine Zusammenarbeit mit anderen Fakultäten oder Instituten ist vorgesehen. Je nach konkret gewähltem Lehrangebot werden die u.a. Kompetenzen unterschiedlich intensiv vermittelt. Das Modul kann auch durch Teilnahme an mehreren verschiedenen kleineren Lehrveranstaltungen erfüllt werden, sofern diese zu den Modulzielen beitragen und die erforderlichen ECTS-Punkte in Summe erreicht sind.
Module Contents	
Project	
Teaching and Learning Methods	Project
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	12 Hours $\pm$ 1 SWS
Self-Study	138 Hours
Recommended Prerequisites	
Mandatory Prerequisites	
Recommended Literature	
Use of the Module in Other Study Programs	▪ XIB in Bachelor Elektrotechnik 2020 ▪ XGA in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ XGA in Bachelor Informatik und Systems-Engineering 2024 ▪ XGA in Master Informatik und Systems-Engineering 2024
Specifics and Notes	Participation in appointment committees as a student member is recognized. The number of ECTS points recognized depends on the number of hours of committee work demonstrated. One ECTS point is recognized for every 25 hours of committee work. The chairperson of the appointment committee awards the ECTS and certifies them. The student is expected to be actively involved in the work.
Last Update	19.7.2025, 14:32:16

6.63 XPSS - Praxisorientierte Summer School

Module ID	XPSS_BaTIN2020
Module Name	Praxisorientierte Summer School
Type of Module	Elective Modules
Recognized Course	XPSS - Practically based Summer School
ECTS credits	5
Language	englisch
Duration of Module	1 Semester
Recommended Semester	4-6
Frequency of Course	every summer term
Module Coordinator	Prof. Dr.-Ing. Ulrich Reiter/Professor Fakultät IME
Lecturer(s)	
Learning Outcome(s)	In dem Modul lernen die Studierenden die Zusammenarbeit in kleinen internationalen Teams. Dazu analysieren sie Problemstellungen und erstellen geeignete kreative Lösungskonzepte, die in Form von Vorträgen präsentiert werden. Sie sammeln praktische Erfahrung in der Realisierung kleinerer Projekte und der Präsentation der erreichten Ergebnisse. Durch die Arbeit in internationalen Teams vertiefen die Studierenden ihre interkulturellen Fähigkeiten. Als Ergebnis des Moduls sind die Teilnehmer und Teilnehmerinnen in der Lage, fachspezifische Aufgabenstellungen zu analysieren, Lösungskonzepte zu entwickeln und technische Systeme in einer internationalen Umgebung zu erstellen.
Module Contents	
Project	Working in small teams, self organisation, project planing, project realisation, presentation
Teaching and Learning Methods	Project
Examination Types with Weights	cf. exam regulations
Workload	150 Hours
Contact Hours	12 Hours $\pm$ 1 SWS
Self-Study	138 Hours
Recommended Prerequisites	▪ Modul EG: Anwenden einfacher elektrischer Schaltungen, ▪ Modul PI1: Erstellen von funktionsfähigen Programmen ▪ Modul MA1: Anwenden grundlegender mathematischer Kenntnisse ▪ Modul MA2: Anwenden fortgeschrittener mathematischer Kenntnisse ▪ Good knowledge in programming microcontroller. Good understanding of electronic elements and electronic components. Experience or skill in developing electronic circuits.
Mandatory Prerequisites	
Recommended Literature	
Use of the Module in Other Study Programs	▪ XPSS in Bachelor Elektrotechnik 2020 ▪ XPSS in Bachelor Elektrotechnik und Informationstechnik 2024 ▪ XIM in Master Technische Informatik 2020
Specifics and Notes	The Summer School will be offered together with collaborating universities. A fixed date can not be guaranteed. Informations to the start of the Summer School will be announced early.

Last Update

19.7.2025, 14:32:16

7. Electives Catalogs

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

7.1 AUS - Auslandssemester

Hier werden an einer ausländischen Hochschule erbrachte Leistungen nach vorheriger Absprache anerkannt, wenn ihr Umfang dem eines Semesters entspricht. Das konkrete Lehrangebot richtet sich nach der ausländischen Hochschule.

7.2 WM - Wahlbereich Informatik und Systems-Engineering

In diesem Wahlbereich können Module der Fakultät 07 für Informations-, Medien- und Elektrotechnik mit Bezug zur Technischen Informatik gewählt werden.

You must select modules of 35 ECTS credit points in total out of this catalog.

This elective catalog particularly includes all modules from the following areas:

- Specialization ES - Eingebettete Systeme
- Specialization IOT - Internet of Things
- Specialization NVS - Netze und Verteilte Systeme
- Specialization SMS - Smart Systems
- Specialization SOS - Software-Systeme

Modules from these other areas are printed normally in the following, original modules from this elective area are printed in bold.

Modules of the faculty:

Module ID	Module Name	ECTS	included in Specialization			
ASN	Angewandte Statistik und Numerik	5				
AT	Antennentechnik	5				
ATS	Autonome Systeme	5	ES		SMS	SOS
BV	Bildverarbeitung	5				
CA	Computeranimation	5			SMS	
CG	Computergrafik	5			SMS	
DB2	Datenbanken 2	5				SOS
DM	Data Mining	5		IOT	SMS	
EKS	Entwicklung komplexer Software-Systeme	5			NVS	SOS
EL	Elektronik	5	ES			
ES	Eingebettete Systeme	5	ES			
ESP	Eingebettete Systeme - Projekt	5	ES			
FIT	Funksysteme für IoT	5		IOT		
HF	Hochfrequenztechnik	5				
IAK	Ingenieurakustik	5				
IOT	IoT Protokolle und Anwendungen	5		IOT	NVS	
KOAK	Kommunikationsakustik	5				
ML	Maschinelles Lernen	5			SMS	
MLO	Machine Learning Operations	5				
MPR	Mobilgeräteprogrammierung	5				
MT	Messtechnik	5				
NSA	Netzsicherheit und Automation	5			NVS	
PPRA	Parallelprogrammierung und Rechnerarchitekturen	5				
QKC	Quellen- und Kanalcodierung	5		IOT		
SM	Software Management	5			NVS	SOS
SMO	Smart Mobility Components	5		IOT		
SMP	Signalverarbeitung mit Matlab/Python und µC	5		IOT		
SOP	Systems on Programmable Chips	5	ES	IOT		

Module ID	Module Name	ECTS	included in Specialization	
UT	Übertragungstechnik	5		
VDS	Verteilte Datenverarbeitungssysteme	5	SMS	
VMA	Programmierung verteilter und mobiler Anwendungen	5	NVS	SOS

7.3 XIB1 - Fachübergreifende Kompetenzen und Soft Skills 1

Die Module zu unterschiedlichen Themen werden in Zusammenarbeit mit anderen Fakultäten oder Institutionen angeboten. Das konkrete Lehrangebot wird in der Regel erst kurzfristig zu Beginn des jeweiligen Semesters festgelegt.

You must select modules of 5 ECTS credit points in total out of this catalog.

Modules of the faculty:

Module ID	Module Name	ECTS	included in Specialization
PHTB	Philosophische Handlungstheorie Bachelor	5	
XIB	Fachübergreifende Kompetenzen und Soft-Skills	5	
XPSS	Praxisorientierte Summer School	5	

8. Specializations

The following section outlines the major fields of study defined in this degree program (see also §24 of the examination regulations). The following information and regulations apply to all major fields of study:

- A major field of study is considered successfully completed if at least 4 of the included modules have been successfully accomplished.
- The completed areas of specialization are listed in a separate appendix to the degree certificate; if there is more than one, only parts of it may be listed upon request to the Examination Office.
- Upon request, a major field of study can be supplemented with additional suitable modules. Such a request must be submitted informally to the program director at least six months before planned participation in a module to be supplemented. The examination board decides on the acceptance of the request in consultation with the program director and appropriately qualified teaching staff.

8.1 ES - Eingebettete Systeme

Der Schwerpunkt "Eingebettete Systeme" befasst sich mit Informationstechnik, die in Objekte aller Art integriert wird und dort in Analogie zu biologischen Systemen das Nervensystem des Geräts bildet. Viele Gegenstände des Alltags sind inzwischen mit eingebetteten Systemen ausgerüstet, von der Kaffeemaschine über alle Arten von Fahrzeugen bis hin zu Gebäuden, nicht zuletzt auf Grund der hohen Verfügbarkeit und niedrigen Kosten der Mikrocontrollertechnik. Der Studienschwerpunkt vermittelt Wissen zu eingebetteten Systemen sowie zu Komponenten eingebetteter Systeme (Sensoren, Aktoren) und zu autonomen Systemen.

Modules of the faculty:

Module ID	Module Name	ECTS
ATS	Autonome Systeme	5
EL	Elektronik	5
ES	Eingebettete Systeme	5
ESP	Eingebettete Systeme - Projekt	5
SOP	Systems on Programmable Chips	5

8.2 IOT - Internet of Things

Das Internet der Dinge (englisch Internet of Things, Kurzform: IoT) ist ein Sammelbegriff für Technologien einer globalen Infrastruktur der Informationsgesellschaften, die es ermöglicht, physische und virtuelle Gegenstände miteinander zu vernetzen und sie durch Informations- und Kommunikationstechniken zusammenarbeiten zu lassen. (Quelle Wikipedia)

Modules of the faculty:

Module ID	Module Name	ECTS
DM	Data Mining	5
FIT	Funksysteme für IoT	5
IOT	IoT Protokolle und Anwendungen	5
QKC	Quellen- und Kanalcodierung	5
SMO	Smart Mobility Components	5
SMP	Signalverarbeitung mit Matlab/Python und μ C	5
SOP	Systems on Programmable Chips	5

8.3 NVS - Netze und Verteilte Systeme

Heute sind Computer meistens vernetzt, also über Kommunikationsnetze, WLANs oder Mobilfunknetze miteinander verbunden. Die Software, die diese Computer ausführen, ist typischerweise verteilt. Der Begriff des vernetzten Computers geht dabei weit über traditionelle Desktop- und Netbook-Computer oder Serversysteme hinaus: Er umfasst beispielsweise auch Mobiltelefone ("Smartphones"), eingebettete Systeme, z.B. in vernetzten Fahrzeugen, oder sensorgestützte Systeme zur Überwachung und Steuerung von Gebäuden und Anlagen ("Smart Home", "Smart Factory"). Die Themengebiete dieses Schwerpunkts erstrecken sich von den Grundlagen der Datenübertragung, über Architekturen und Modelle für Computernetze sowie ihre Realisierung durch Protokollhierarchien, bis hin zur Programmierung von Mobilgeräten und Webanwendungen.

Modules of the faculty:

Module ID	Module Name	ECTS
EKS	Entwicklung komplexer Software-Systeme	5
IOT	IoT Protokolle und Anwendungen	5
NSA	Netzsicherheit und Automation	5
SM	Software Management	5
VMA	Programmierung verteilter und mobiler Anwendungen	5

8.4 SMS - Smart Systems

Smart Systems sind „intelligente Systeme“, die in der Lage sind, auf Basis einer Datenanalyse selbstständig Entscheidungen, Prognosen oder Empfehlungen zu erzeugen, die dann sowohl zur autonomen Reaktion eines Systems oder auch als Hilfestellung für den Menschen herangezogen werden können. Smart-Systems kommen in verschiedensten Bereichen zum Einsatz. Im Bereich Automotiv z.B. für Einparkhilfen, Abstandsregeltempomaten oder Spurwechselassistenten. Im Bereich der Medizintechnik z.B. für die Auswertung von Röntgenaufnahmen zur Unterstützung des Arztes. Im industriellen Bereich z.B. zur Überwachung von Produktionsprozesse und zur Qualitätskontrolle. Im kaufmännischen Bereich z.B. zur Analyse von Kundenbestellungen zur automatischen Generierung kundenspezifischer Werbemaßnahmen.

Modules of the faculty:

Module ID	Module Name	ECTS
ATS	Autonome Systeme	5
CA	Computeranimation	5
CG	Computergrafik	5
DM	Data Mining	5
ML	Maschinelles Lernen	5
VDS	Verteilte Datenverarbeitungssysteme	5

8.5 SOS - Software-Systeme

In der heutigen Informationsgesellschaft wird in fast allen Bereichen komplexe Anwendungssoftware eingesetzt. Die Anwendungen treten dabei in unterschiedlichen Rollen auf: als zentrale Applikationen auf einem Server, als leistungsstarke Klienten, als autonome Systeme sowie als verteilte Anwendungen. In diesem hochdynamischen Umfeld entwickeln sich ständig neue Trends und interessante Technologien. Um solche komplexen Software-Systeme zu realisieren und zu betreiben, sind vielfältige Kenntnisse, Fähigkeiten und Fertigkeiten erforderlich. Die im Schwerpunkt Software-Systeme behandelten Themengebiete umfassen alle Bereiche moderner Anwendungssysteme: Architektur und Design komplexer Software-Systeme, intuitive Benutzerführung, effektive Realisierung der Geschäftslogik, effiziente Datenspeicherung, Programmierung mobiler und autonomer Geräte.

Modules of the faculty:

Module ID	Module Name	ECTS
ATS	Autonome Systeme	5
DB2	Datenbanken 2	5
EKS	Entwicklung komplexer Software-Systeme	5
SM	Software Management	5
VMA	Programmierung verteilter und mobiler Anwendungen	5

9. Examination Types

The forms of examination referenced in the module descriptions are explained in more detail below. The explanations are taken from the examination regulations, §19ff. In case of deviations, the text of the examination regulations applies.

(Digital) Written exam

Written, paper-based or digitally supported examination. Details are regulated in §19 of the examination regulations.

Oral examination

Examination to be taken orally. Details are regulated in §21 of the examination regulations.

Oral contribution

See §22, para. 5 of the examination regulations: An oral contribution (e.g. paper, presentation, negotiation, moderation) serves to determine whether students are capable of independently working on a practice-oriented task within a specified period of time using scientific and practical methods and presenting it in a technically appropriate manner by means of verbal communication. This also includes answering questions from the auditorium regarding the oral presentation. The duration of the oral presentation is determined by the examiner at the beginning of the semester. The facts relevant to the grading of the oral presentation are to be recorded in a protocol; students should also submit the written documents relating to the oral presentation for documentation purposes. Students must be notified of the grade no later than one week after the oral presentation.

Technical discussion

See §22, Para. 8 of the examination regulations: A technical discussion serves to determine professional competence, understanding of complex technical contexts and the ability to solve problems analytically. Students and examiners have roughly equal speaking time in the technical discussion in order to enable a discursive technical exchange. One or more discussions are held with an examiner during the semester or in summary form. Students should present and explain practice-related technical tasks, problems or project plans from the degree program and explain the relevant technical background, theoretical concepts and methodological approaches for processing the tasks. Possible solutions, procedures and considerations for solving the problem must be discussed and justified. The facts relevant to the grading of the technical discussion must be recorded in a protocol.

Project work

See §22, Para. 6 of the examination regulations: The project work is an examination that consists of independently working on a specific problem under supervision using scientific methodology and documenting the results. In addition to the quality of the answer to the question, the organizational and communicative quality of the implementation, such as slides, presentations, milestones, project plans, meeting minutes, etc., are also relevant for assessment.

Lab report

See §22, para. 10 of the examination regulations: An internship report (e.g. experimental protocol) serves to determine whether students are capable of independently carrying out a practical laboratory task within a specified period of time, as well as documenting, evaluating and reflecting on the process and results in writing. Preparatory homework may be required before the actual experiment is carried out. Technical discussions may take place during or after the experiment. Internship reports can also be admitted to the examination in the form of group work. Students must be notified of the assessment of the practical placement report no later than six weeks after submission of the report.

Exercise lab

See §22, para. 11 of the examination regulations: The examination form "practical training" tests the technical skills in the application of the theories and concepts learned in the lecture as well as practical skills, for example the use of development tools and technologies. For this purpose, several tasks are set during the semester, which are to be solved either alone or in group work, on site or as homework by a given deadline. The solutions to the tasks must be submitted by the students in (digital) written form. The exact criteria for passing the examination will be announced at the beginning of the corresponding course.

Exercise lab under examination conditions

See §22, para. 11, sentence 5 of the examination regulations: A "practical training course under examination conditions" is a practical training course in which the tasks are to be completed within the time frame and under the independent conditions of an examination.

Term paper

See §22, para. 3 of the examination regulations: A term paper (e.g. case study, research) serves to determine whether students are capable of independently completing a specialist task in written or electronic form using scientific and practical methods within a specified period of time. The topic and scope (e.g. number of pages of the text section) of the term paper are determined by the examiner at the beginning of the semester. A declaration of independence must be signed and submitted by the candidate. In addition, technical discussions may be held.

Learning portfolio

A learning portfolio documents the student competence development process by means of presentations, essays, excerpts from internship reports, tables of contents of term papers, notes, to-do lists, research reports and other performance presentations and learning productions, summarized as so-called "artefacts". The learning portfolio only becomes an examination item in conjunction with the student's reflection (in writing, orally or in a video) on the use of these artifacts to achieve the learning objective previously made transparent by the examiner. During the creation of the learning portfolio, feedback on development steps and/or artifacts is given over the course of the semester. A revised form of the learning portfolio - in handwritten or electronic form - is submitted as the examination result following the feedback.

Single / Multiple choice

See §20 of the examination regulations.

Access colloquium

See §22, para. 12 of the examination regulations: An entrance colloquium serves to determine whether the students fulfill the specific requirements to be able to work independently and safely on a defined practical laboratory task using scientific and practical methods.

(Intermediate) Certificate

See §22, para. 7 of the examination regulations: A test/intermediate test certifies that the student has completed a piece of coursework (e.g. draft) to the required standard. The scope of work to be completed and the required content and requirements can be found in the respective module description in the module handbook and in the assignment.

Open book preparation

The open book assignment (OBA) is a short term paper and therefore an unsupervised written or electronic examination. It is characterized by the fact that, according to the examiner's declaration of aids, all aids are generally permitted. Special attention is drawn to the safeguarding of good scientific practice through proper citation etc. and the requirement of independence in the performance of each examination.

Thesis

Bachelor's or Master's thesis as defined in the examination regulations §25ff: The Master's thesis is a written assignment. It should show that the student is capable of independently working on a topic from their subject area within a specified period of time, both in its technical details and in its interdisciplinary contexts, using scientific and practical methods. Interdisciplinary cooperation can also be taken into account in the final thesis.

Colloquium

Colloquium for the Bachelor's or Master's thesis as defined in the examination regulations §29: The colloquium serves to determine whether the student is able to present the results of the Master's thesis, its technical and methodological foundations, interdisciplinary contexts and extracurricular references orally, to justify them independently and to assess their significance for practice.

10. Profile Module Matrix

The following section describes the extent to which the modules of the degree program support and develop the competencies and fields of action of the study program as well as certain study program criteria as defined by the University of Applied Science TH Köln.

Abbr.	Module Name	HF1 - Systeme zur Verarbeitung,...	HF2 - Anforderungen, Konzepte u...	HF3 - Informationstechnische Sy...	HF4 - Mit Auftraggebern, Anwend...	K.1 - In Systemen denken	K.2 - fachliche Probleme abstra...	K.3 - Konzepte und Methoden der...	K.4 - Systeme analysieren	K.5 - Systeme entwerfen	K.6 - Systeme realisieren	K.7 - Systeme prüfen	K.8 - Informationen beschaffen ...	K.9 - Typische Werkzeuge, Stand...	K.10 - In vorhandene Systeme ein...	K.11 - Projekte organisieren	K.12 - Grundzüge wissenschaftlic...	K.13 - Komplexe technische Aufga...	K.14 - Gesellschaftliche und eth...	K.15 - Befähigung zum lebenslang...	K.16 - Kommunikative und interku...	SK.1 - Global Citizenship	SK.2 - Internationalisierung	SK.3 - Interdisziplinarität	SK.4 - Transfer
AD	Algorithmen und Datenstrukturen	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●		●	●	●				
ASN	Angewandte Statistik und Numerik	●	●	●	●	●	●	●	●	●			●								●				
AT	Antennentechnik	●	●	●	●				●		●	●	●	●		●	●				●	●			
ATS	Autonome Systeme	●		●	●	●	●	●	●	●	●	●	●								●				
BAA	Bachelorarbeit	●	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●	●	●	●			
BVS1	Betriebssysteme	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●		●	●	●				
BVS2	Verteilte Systeme	●	●	●	●	●	●	●	●	●	●	●	●			●	●				●				
BWR	Betriebswirtschaft und Recht				●		●										●		●	●		●		●	
CA	Computeranimation	●		●	●	●	●	●	●	●	●	●	●							●				●	●
CG	Computergrafik	●			●	●	●	●	●		●	●	●						●	●				●	●
DB1	Datenbanken 1	●	●	●	●	●	●	●	●	●	●			●							●				
DB2	Datenbanken 2	●	●	●	●	●	●				●	●													
DM	Data Mining	●		●	●		●		●				●					●	●	●					
DR	Digitalrechner	●		●	●	●	●	●	●	●	●	●	●		●	●			●	●					
EG	Elektrotechnische Grundlagen für die Technische Informatik	●	●			●	●	●	●				●					●		●	●				
EKS	Entwicklung komplexer Software-Systeme	●	●	●	●	●	●	●	●	●	●	●	●	●					●	●	●				
EL	Elektronik	●	●	●	●	●	●		●	●	●	●		●		●			●	●					
ES	Eingebettete Systeme	●		●	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●				

Abbr.	Module Name	HF1 - Systeme zur Verarbeitung,...	HF2 - Anforderungen, Konzepte u...	HF3 - Informationstechnische Sy...	HF4 - Mit Auftraggebern, Anwend...	K.1 - In Systemen denken	K.2 - fachliche Probleme abstra...	K.3 - Konzepte und Methoden der...	K.4 - Systeme analysieren	K.5 - Systeme entwerfen	K.6 - Systeme realisieren	K.7 - Systeme prüfen	K.8 - Informationen beschaffen ...	K.9 - Typische Werkzeuge, Stand...	K.10 - In vorhandene Systeme ein...	K.11 - Projekte organisieren	K.12 - Grundzüge wissenschaftlic...	K.13 - Komplexe technische Aufga...	K.14 - Gesellschaftliche und eth...	K.15 - Befähigung zum lebenslang...	K.16 - Kommunikative und interku...	SK.1 - Global Citizenship	SK.2 - Internationalisierung	SK.3 - Interdisziplinarität	SK.4 - Transfer
ESP	Eingebettete Systeme - Projekt	●	●	●			●		●	●	●	●			●	●	●		●	●					
FIT	Funksysteme für IoT	●	●	●	●		●	●	●							●	●								
FSA	Formale Sprachen und Automatentheorie	●	●	●	●	●	●	●	●	●	●		●												
GRT	Graphentheorie	●					●		●																
GSP	Grundlagen der Systemprogrammierung	●	●	●	●	●	●	●	●	●	●	●	●	●					●	●	●				
GUI	Graphische Oberflächen und Interaktion	●	●	●		●	●			●	●	●				●	●			●					
HF	Hochfrequenztechnik	●	●	●	●				●	●				●											
IAK	Ingenieurakustik	●	●	●	●	●		●		●									●						
IBV	Industrielle Bildverarbeitung	●	●	●	●	●	●	●	●	●	●	●	●			●	●		●						
IOT	IoT Protokolle und Anwendungen	●	●	●	●	●	●	●	●	●	●														
IPM	IT-Projektmanagement	●	●	●	●	●			●					●		●									
ITS	IT-Sicherheit	●	●	●	●					●	●	●						●							
KOAK	Kommunikationsakustik	●	●	●	●	●		●	●	●									●						
KOLL	Kolloquium zur Bachelorarbeit													●					●		●		●	●	●
MA1	Mathematik 1	●	●	●		●		●						●											
MA2	Mathematik 2	●	●	●		●		●						●											
ML	Maschinelles Lernen	●	●	●	●	●	●	●	●	●	●	●	●			●	●		●						
MLO	Machine Learning Operations	●	●	●	●	●	●	●	●	●	●	●	●			●	●		●						
MT	Messtechnik	●	●	●	●	●	●	●	●		●					●	●	●							

Abbr.	Module Name	HF1 - Systeme zur Verarbeitung,...	HF2 - Anforderungen, Konzepte u...	HF3 - Informationstechnische Sy...	HF4 - Mit Auftraggebern, Anwend...	K.1 - In Systemen denken	K.2 - fachliche Probleme abstra...	K.3 - Konzepte und Methoden der...	K.4 - Systeme analysieren	K.5 - Systeme entwerfen	K.6 - Systeme realisieren	K.7 - Systeme prüfen	K.8 - Informationen beschaffen ...	K.9 - Typische Werkzeuge, Stand...	K.10 - In vorhandene Systeme ein...	K.11 - Projekte organisieren	K.12 - Grundzüge wissenschaftlic...	K.13 - Komplexe technische Aufga...	K.14 - Gesellschaftliche und eth...	K.15 - Befähigung zum lebenslang...	K.16 - Kommunikative und interku...	SK.1 - Global Citizenship	SK.2 - Internationalisierung	SK.3 - Interdisziplinarität	SK.4 - Transfer
NDQ	Nachhaltigkeit durch Qualität		●		●		●	●				●		●		●	●	●			●			●	●
NP	Netze und Protokolle	●	●	●	●	●	●	●	●	●	●	●	●	●					●	●	●				
NSA	Netzsicherheit und Automation	●	●	●	●	●	●	●	●	●	●	●	●	●					●	●	●				
PHTB	Philosophische Handlungstheorie Bachelor																					●		●	
PI1	Praktische Informatik 1	●		●	●		●		●		●	●	●	●		●	●		●	●	●				
PI2	Praktische Informatik 2	●		●	●		●	●	●		●	●													
PP	Programmierpraktikum	●		●	●	●	●	●	●		●	●	●	●		●	●		●	●	●				
PPRA	Parallelprogrammierung und Rechnerarchitekturen	●		●	●	●	●	●	●	●	●	●	●	●					●	●	●				
PRA	Praxisphase	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
PUK	Präsentation und Kommunikation												●	●				●		●	●	●		●	
QKC	Quellen- und Kanalcodierung	●		●	●	●	●	●	●	●															
RT	Regelungstechnik	●	●	●	●	●	●	●	●	●	●	●		●											
SE	Software Engineering	●	●	●	●	●	●	●	●	●	●	●													
SIG	Signalverarbeitung	●	●	●	●	●	●	●	●	●	●														
SM	Software Management	●	●	●	●	●	●	●	●	●	●	●	●	●	●			●		●					
SMO	Smart Mobility Components	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●	●	●				●	●
SMP	Signalverarbeitung mit Matlab/Python und µC	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●				
SOP	Systems on Programmable Chips	●		●	●	●		●	●	●	●	●				●	●								

Abbr.	Module Name	HF1 - Systeme zur Verarbeitung,...	HF2 - Anforderungen, Konzepte u...	HF3 - Informationstechnische Sy...	HF4 - Mit Auftraggebern, Anwend...	K.1 - In Systemen denken	K.2 - fachliche Probleme abstra...	K.3 - Konzepte und Methoden der...	K.4 - Systeme analysieren	K.5 - Systeme entwerfen	K.6 - Systeme realisieren	K.7 - Systeme prüfen	K.8 - Informationen beschaffen ...	K.9 - Typische Werkzeuge, Stand...	K.10 - In vorhandene Systeme ein...	K.11 - Projekte organisieren	K.12 - Grundzüge wissenschaftlic...	K.13 - Komplexe technische Aufga...	K.14 - Gesellschaftliche und eth...	K.15 - Befähigung zum lebenslang...	K.16 - Kommunikative und interku...	SK.1 - Global Citizenship	SK.2 - Internationalisierung	SK.3 - Interdisziplinarität	SK.4 - Transfer
SWP	Softwarepraktikum	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●		●	●	●				
SYP	Systementwicklungs-Projekt	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
UT	Übertragungstechnik	●	●	●	●	●	●	●	●	●			●						●	●					
VDS	Verteilte Datenverarbeitungssysteme	●	●	●	●	●	●	●	●	●	●	●	●			●	●			●					
VMA	Programmierung verteilter und mobiler Anwendungen	●	●		●		●				●	●	●	●	●			●	●	●	●				
XIB	Fachübergreifende Kompetenzen und Soft-Skills																								
XIB2	Fachübergreifende Kompetenzen und Soft-Skills																								
XPSS	Praxisorientierte Summer School	●	●		●	●									●						●				

11. Version History

The table below lists the different versions of the course offer. The versions are sorted in reverse chronological order with the currently valid version in the first row. The individual versions can be accessed via the link in the right-hand column on the right.

Version	Date	Changes	Link
3.9	2024-12-06-08-45-55	1. Begutachtete Version für Reakkreditierung 2024 2. Neues Layout für sämtliche Modulhandbücher	Link
3.8	2024-06-11-14-00-00	1. Mitarbeit in Gremien im Bachelor/Master Technische Informatik und Bachelor Elektrotechnik 2. Produktentwicklung für Smart-Cities im Wintersemester (verantwortlich jetzt Prof. Humpert) 3. Überarbeitung Modul und Lehrveranstaltung Elektrische Sicherheit und EMV 4. Überarbeitung Modul und Lehrveranstaltung Hochspannungstechnik 5. Überarbeitung Modul und Lehrveranstaltung Hochspannungsübertragungstechnik 6. Neue(s) Modul und Lehrveranstaltung Themen der Medientechnologie	Link
3.7	2024-05-10-14-30-00	1. Weitere Literaturangabe in Lehrveranstaltung Antennentechnik 2. Medienrecht: Tutorium entfernt, Prüfungstyp zu "andere studienbegleitende Prüfungsform" geändert, weitere Literaturangaben, Modul in Bachelor Medientechnologie angelegt 3. Änderungen der Dozenten und Literatur in Lehrveranstaltung "Informatik 1/2", "Mensch-Maschine-Interaktion" und "Parallelprogrammierung" 4. Lehrveranstaltung Leistungselektronik: Überarbeitung der Prüfungsdetails und Literaturangaben 5. Änderung der Modulverantwortung von Kunz zu Reiter / Grünvogel / Salmen in "Visuelle und auditive Wahrnehmung", "Mathematik 2" und "Signaltheorie und angewandte Mathematik" 6. Anlage Modul "Interdisziplinäres Projekt" im Bachelor Medientechnologie 7. Lehrveranstaltung Energiemanagement in Energieverbundsystemen: Änderungen an Präsenzzeit, Räumen und Prüfungstyp (jetzt Projektarbeit + E-Assessment) 8. Lehrveranstaltung Energiespeicher: Streichung des Tutoriums, Prüfungstyp jetzt Fachgespräch 9. Lehrveranstaltung Energiewirtschaft: Überarbeitung der Fertigungsbeschreibungen und des Begleitmaterials, Streichung des Tutoriums 10. Modul SmartCity: Änderung Modulverantwortung von Stadler zu Humpert 11. Lehrveranstaltung Elektrische Antriebe: Überarbeitung der Prüfungsbeschreibung und Praktikumskompetenzen 12. Lehrveranstaltung Schaltnetzwerke: Überarbeitungen der Prüfungsbeschreibung, -typ, Begleitmaterialien, Präsenzlehre und Fertigkeiten. 13. Leistungselektronische Stellglieder für PV- und Windkraftanlagen: Überarbeitungen der Prüfungsbeschreibung, -typ, Begleitmaterialien, Präsenzlehre und Fertigkeiten.	Link
3.6	2024-02-23-15-00-00	1. Generelle Überarbeitung des Layouts 2. Eingangstexte bei Wahlmodulkatalogen und Schwerpunkten überarbeitet und POs angeglichen 3. Lehrveranstaltung BWR (Kim) sowohl im Sommer- als auch Wintersemester. 4. Modellierung von Energiesystemen der Fakultät 09 als wählbares Modul im allgemeinen Wahlkatalog im Master Technische Informatik	Link
3.5	2023-07-20-15-00-00	1. Neue Lehrveranstaltung "Projektleitung Systementwurfs-Praktikum" im Master Technische Informatik (anerkannt für XIM) 2. Geänderter Studienverlauf mit vermindertem Workload in Bachelor Technische Informatik	Link

Version	Date	Changes	Link
3.4	2023-04-17-15-00-00	1. Modul "Netze und Automation (NSA)" (vormals "Internetworking (IN)") den Schwerpunkten "Internet of Things" und "Information- und Kommunikationstechnik" des Studiengangs "Bachelor Elektrotechnik" hinzugefügt (FR-2023-6) 2. Modulkürzel für "Autonome Systeme" lautet jetzt auch in "Bachelor Technische Informatik" ATS (FR-2023-7)	Link
3.3	2023-03-27-11-00-00	1. Lehrveranstaltung Grundgebiete der Elektrotechnik 2 (Kronberger) aktualisiert; Modul "Entrepreneurship, Gewerblicher Rechtsschutz, Market Knowledge" auch für X1 in Master Technische Informatik wählbar.	Link
3.2	2023-03-08-16-00-00	1. Licht- und Beleuchtungstechnik (LB) Wintersemester-Wahlmodulen zugeordnet; BaTIN-Modul "Web-Architekturen" SGL zugeordnet und vakante Lehrveranstaltung vorerst entfernt.	Link
3.1	2023-02-24-20-00-00	1. Allgemeine Bereinigung von kaputten Links (http 404)	Link
3.0	2022-12-09-12-31-39	1. Umbenennung des Moduls und der Lehrveranstaltung "Internetworking und Netzsicherheit (IN)" in "Netzsicherheit und Automation (NSA) (FR-2022-20)"	Link

Impressum

Datenschutzhinweis

Haftungshinweis

Bei Fehlern, bitte Mitteilung an
die
modulhandbuchredaktion@f07.th-koeln.de